Statistical Signal Processing for Target Tracking with Multipath Radar

Bentarage Sachintha Karunaratne

Submitted in total fulfilment of the requirements of the degree of

Doctor of Philosophy

Department of Electrical and Electronic Engineering
THE UNIVERSITY OF MELBOURNE

Produced on archival quality paper

December, 2014

Abstract

Until recently the applications of radar technology have been limited to environments with large open space. One of the reasons for such confinement is the radar system's dependence on line of sight communication for detection and/or tracking; this is easily accomplished in an open space environment. Additionally, it has been long believed that multipath reflections hinder the functions of a radar system.

Modern research efforts, however, have radically challenged the belief that multipath reflections are nuisances. Various studies suggest that multipath reflections contain information that could be exploited for tracking and/or detecting objects. This paradigm shift has enabled potential new applications in radar technology, particularly pertaining to multipath rich dense urban environments. The main focus of this thesis is the study of radar tracking using multipath in an urban environment. We have particularly emphasised accounting for uncertainty inherent in urban environments. In doing so, we have adapted a Bayesian probabilistic framework for inferential tasks.

After introducing a robust model for a multipath environment, we derive performance bounds for tracking a moving target in such an environment. Recent developments in nonlinear statistical signal processing, as well as the availability of powerful computing resources have enabled us to design statistical filters for challenging tracking problems. Consequently, we propose a novel Markov Chain Monte Carlo based particle filter to address the tracking problem pertaining to our multipath model. We then address the multipath tracking problem where much larger uncertainty exists on the locations of the building in the urban environment; that is, when a map of the environment is not available. We also study a particular generalisation of a multivariate von-Mises distribution, which was encountered while addressing the tracking problem. A comprehensive Bayesian conjugate analysis of this distribution is provided.

Declaration

This is to certify that

1. the thesis comprises only my original work towards the PhD,
2. due acknowledgement has been made in the text to all other material used,
3. the thesis is fewer than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Bentarage Sachintha Karunaratne

Date

"Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise."

— John Tukey (1915-2000)

Acknowledgements

First and foremost, I would like to thank my supervisors, Prof. Bill Moran and Dr. Mark Morelande for guiding me and being sources of light and strength in a challenging journey. On the 24th of February 2009, I was overjoyed with excitement when Prof. Bill Moran responded to me with his willingness to supervise my PhD. It is an honour to be a disciple of Prof. Moran, a brilliant mathematician and a true researcher. Despite his busy schedule, he always allocated time to ensure that my research continued smoothly, and actively steered the direction of my study. His expertise and insights into mathematics proved very valuable in overcoming crucial hurdles in my research work. Not only was he interested in developing my academic maturity, Prof. Moran, shielded me countless times from typical administrative bureaucracies. Dr. Mark Morelande, an extraordinary researcher and a humble human being, empathetically guided and more importantly taught me *how to think*. I learnt and developed a lot by having the opportunity to work with him on an extremely technical level. His ability to come up with a solution to a seemingly unsolvable problem is something that inspires me. This thesis is largely made possible by these two giants, Prof. Moran and Dr. Morelande, who held me steadily on their shoulders.

I am ever in debt to my wonderful wife Imali for the tremendous support she gave to me despite herself working towards a PhD. During this challenging period, her love, kindness, understanding, and encouragement had significant positive impact on my research efforts. I specifically appreciate the efforts she put towards looking after our son, Methindu, who was born during the period I was busy with my PhD studies. I also thank Methindu, my biggest contribution to this world by far, for cheering me up with his little gestures of love, at times I needed it the most.

My loving parents, Dr. Vineetha Karunatatne and Dr. B. Karunaratne had always been a source of great strength, inspiration, and encouragement for me throughout my life. Their help, particularly in proof-reading my thesis and publications, is much appreciated. Additionally the support they gave me by flying to Australia and staying with my family for an extensive period looking after Methindu is invaluable. I am fortunate to be blessed with wonderful in-laws Mrs. Manel Wickramasinghe and Mr. Bernard Dias, who also took turns flying to Australia to help look after Methindu while I focused on research.

I take this opportunity to thank my teachers and lecturers both at D. S. Senanayake College and the University of Moratuwa, Sri Lanka. It is with utmost respect that I thank them for laying a solid foundation in my education, which lead me to pursue higher studies.

Though I am unable to name individually here, I thank all my friends and colleagues at The University of Melbourne for creating a friendly atmosphere to conduct my research. I have met so many wonderful friends during my studies whose friendship will last a lifetime. In particular, I thank Sei Zhen Kong for inspiring me to be a good PhD student and for helping me adapt to the research atmosphere when I commenced my studies at The University of Melbourne. Additionally, I

thank all my friends and relatives in Sri Lanka for all the positive influence they had on me.

I acknowledge The University of Melbourne, one of the leading universities in the world, for admitting me to the PhD programme and waiving-off course fees while providing a stipend scholarship. Without that support, it would not have been possible for me to pursue higher studies at such a prestigious university. I am grateful to the National ICT Australia (NICTA) for funding my international conference travels and granting me a top-up scholarship. Additionally NICTA provided a stipend scholarship in my last six months of the candidature. I mention with great gratitude Mr. Chris Mrakas, Chief Executive Officer of GreenBox-Group Pvt. Limited for employing me part time while writing this thesis. Finally, I like to thank Dr. Malka Halgamuge for recommending me to Chris in a much needed time.

Contents

List of Figures

CHAPTER 1

Introduction

1.1 What is multipath?

A radar system operates by transmitting electromagnetic signals, which upon contact with various objects are reflected back to a radar sensor. The signal that is received is then processed to detect and/or track the state of the objects from which it is reflected. Unless the radar system operates in an open area, the radar sensor receives signals over multiple paths due to the transmitted signal being scattered by other uninteresting objects in the environment. We use the term "direct path" to refer to a path that only comes into contact with the object of interest, whilst a path that arises due to multiple reflections is referred to as a "multipath".

Typically, in conventional radar systems, only the direct path is considered valuable or informative; therefore various techniques are used to mitigate the effects of multipath. The modern treatment of the subject is quite the opposite; since some of the multipaths may have come into contact with the object of interest, recent studies have initiated a paradigm shift where multipath is exploited to enhance the radar detection and filtering process.

This thesis contributes and further extends the research on multipath radar by rigorously investigating some interesting topics related to the subject.

1.2 Summary of contributions

- We introduce a novel measurement model for the multipath radar tracking problem. One of the objectives of the model is to probabilistically formulate some of the uncertainty prevailing in a realistic radar environment, which is not accounted for in existing models. This is done by extending the radar measurement equation by introducing some random parameters. Further, the new model does not impose any geometrical restrictions on the placement of obstacles within the environment.

- We investigate the performance bounds for estimating the target state using

our model of the environment. The Posterior Cramér Rao Bound (PCRB) is derived by exploiting the physical relationships formed in the multipath environment due to specular nature of the reflections.

- We propose a Markov Chain Monte Carlo (MCMC) based particle filter to solve the difficult filtering problem associated with tracking a target under the proposed measurement model.
- We study the target tracking problem when the map of the environment is unavailable. The data association problem appearing in this context is solved using an importance sampling based method.
- We present a particular generalisation of the multivariate von-Mises (VM) distribution, which has a direct relationship to the proposed multipath measurement model. A comprehensive Bayesian analysis of the new distribution is provided.

1.3 List of publications

Journal articles

- Scholarly journal articles titled *"Target tracking in a partially unknown multipath environment using an MCMC based particle filter"* and *"Multipath radar tracking in an unknown environment"* are under preparation.

International conferences

- B. S. Karunaratne, M. Morelande, and B. Moran, *"MCMC particle filter for tracking in a partially known multipath environment"* in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, Vancouver, Canada.
- B. S. Karunaratne, M. Morelande, and B. Moran, *"Target tracking in a multipath environment"* in *IET International Conference on Radar Systems*, Glasgow, UK, 2012.
- B. S. Karunaratne, M. Morelande, and B. Moran, *"Bayesian conjugate analysis for multiple phase Estimation"* in *IEEE International Conference on Information Fusion (FUSION)*, Singapore, 2012.
- B. S. Karunaratne, M. Morelande, B. Moran, and S.Howard, *"Performance bounds for tracking in a multipath environment"* in *IEEE International Conference*

on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 2011, pp. 3652-3655.

1.4 The organisation of the thesis

From a high level perspective, our efforts have focused on modelling and mitigation of the uncertainty present in a realistic multipath radar environment. In most of the existing literature on multipath radar, it is assumed that the environment is known precisely. We believe that by acknowledging the unknown and incorporating the resulting uncertainty through mathematical models, more robust filtering solutions could be developed. We have adapted the Bayesian probabilistic framework throughout our work, since it blends naturally into translating the uncertainty associated with the multipath environment into stochastic mathematical models. The core of this thesis spans over six chapters.

Chapter 2 provides background material required for understanding the rest of the thesis. Here the objectives are twofold. The first being to introduce some concepts relevant to statistical signal processing, mainly from a Bayesian point of view. Introductions to popular tracking algorithms are also provided. The second objective is to introduce briefly some of the main radar system concepts.

In Chapter 3, we study the performance bounds for tracking a target in a multipath environment. First, we propose a model for the multipath environment where the locations of the walls are only approximately known. In other words, we assume that there exists some uncertainty in the knowledge about the location of the walls. We believe that this is a reasonably realistic assumption given the availability of electronic data such as Google Maps. As an example, the uncertainty in the location could be due to the errors in the map used to obtain knowledge about the environment. Further, we relax the assumption that the reflectivity factors of the walls are known exactly. These various sources of uncertainty in the environment are captured in the model by introducing relevant parameters to the radar measurement equation and treating them as unknown random parameters. As an example, the uncertainty in the wall locations is captured by introducing a random phase shift to each of the multipath signals. The PCRB is defined as the lower bound on mean squared error for estimating random parameters. In this chapter, we derive the PCRB for tracking a target under the proposed multipath model.

The random phase shifts that we introduced into the measurement equation make the filter design process challenging. These parameters are circular variables by definition. Thus, while in pursuit of designing a filter, we were led to investi-

gate a particular multivariate generalisation of the VM distribution. In particular, the normalising constant of this multivariate VM distribution has a direct relationship with the multipath radar target tracking problem; the specifics are described in the same chapter. A Bayesian conjugate analysis for this Generalised von-Mises (GVM) is described in detailed in Chapter 4.

The statistical filter developed for the multipath filtering problem introduced in Chapter 3 is described in Chapter 5. The estimation problem proved to be very challenging, particularly because of the environmental uncertainty in the model. The standard nonlinear filtering methods such as variants of the Kalman filter were not even remotely suitable for our problem. We used a sequential Monte Carlo technique known as particle filtering to develop a solution. However, we had to adopt a different approach from the common one of using importance sampling to generate the required particles for many reasons that are also discussed in Chapter 5. As an alternative to importance sampling, we made use of MCMC to generate the required particles.

In Chapter 6, we extend the previous problem by considering a model where extremely large uncertainty of the locations prevails. The brute force method of solving the data association problem appearing in this context was not feasible because the number of association hypotheses grows exponentially with the number of obstacles. We provide an elegant solution to this problem through importance sampling. It is shown in the same chapter that explicit ordering of the elements of the state vector sometimes adversely affect the filter performance. This problem was solved in Chapter 6 using a recently published method known as Set Joint Probabilistic Data Association Filter (SJPDAF). Because of the difficulty posed by the tracking problem, we had to explore beyond the realm of traditional tracking algorithms to propose a robust solution.

CHAPTER 2

Background

Summary

In this chapter, we present the relevant background information to assist in comprehending the rest of the thesis. The chapter is divided into two main parts. In the first part, the statistical signal processing concepts are introduced. Brief introductions to tracking algorithms: Kalman filter, Extended Kalman filter, Unscented Kalman filter, and Particle filter are also discussed in the first part. The second part is reserved to introduce the important principles of a generic radar system.

2.1 Introduction to statistical signal processing for target tracking

The main aim in statistical signal processing is to infer and/or predict unobserved parameters of a system upon observing noisy measurements. Apart from the noise present in the measurements, the unobserved parameters of the system may also introduce randomness to the overall system through non-deterministic dynamic behaviour. The methods used in statistical signal processing can be divided into two broad categories based on whether the unobserved parameters are considered deterministic or random. The former is known as the classical approach to statistical signal processing while the latter is known as the "Bayesian" approach.

We explain the difference between the two approaches with the aid of a simple estimation problem.

Consider a simple scenario where a single unobserved parameter denoted by $\theta \in \mathbb{R}$ is concerned. We observe a sequence of N measurements $y_1, y_2, \ldots, y_N$ given by

$$y_i = \theta + w_i \qquad \text{for } i = 1, 2, \ldots, N, \tag{2.1}$$

where $w_1, w_2, \ldots, w_N$ are Independent and Identically Distributed (IID) random variables with

$$w_i \sim \mathcal{N}(\cdot; 0, \sigma^2) \qquad \text{for } i = 1, 2, \ldots, N. \tag{2.2}$$

Here $\mathcal{N}()$ denotes the normal distribution with the usual parameterisation. Assume that the noise variance $\sigma^2 \in \mathbb{R}$ is known.

2.1.1 Classical inference of the unobserved parameter θ

In the classical approach the parameter θ is treated as a deterministic (but unknown) quantity. An estimator of θ upon observing the measurements $\mathbf{y}_n$ is a function of the measurements, denoted by $\hat{\theta}(\mathbf{y}_n)$. An estimator is said to be unbiased if the expected value of the estimator (averaged over the measurements) is equal to the true parameter θ, for all possible θ; that is, for an unbiased estimator $\hat{\theta}(\mathbf{y}_n)$:

$$\theta = \int_{\mathbf{y}} \hat{\theta}(\mathbf{y}_n) p(y_1, y_2, \ldots, y_N) \mathrm{d}y_1 \mathrm{d}y_2 \cdots \mathrm{d}y_N, \qquad \text{for all possible } \theta. \quad (2.3)$$

Among all the unbiased estimators, the one that has the minimum variance is of particular interest, and is known as the Minimum Variance Unbiased Estimator (MVUE).

For the simple example considered here, the likelihood function $p(\mathbf{y};\theta)$ is given by

$$\begin{aligned} \mathcal{L}(\theta;\mathbf{y}) &= \prod_{i=1}^{N} \mathcal{N}(y_i;\theta,\sigma^2), \\ &= \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i-\theta)^2\right). \end{aligned} \quad (2.4)$$

Though the likelihood function is a probability measure, $p(\mathbf{y};\theta)$, defined on the space of measurements, it is helpful to treat it as a function of θ, with the measurements fixed. This leads to the popular classical estimator known as the Maximum Likelihood Estimator (MLE), which is defined as the value of θ that maximises the likelihood function. We simulated an experiment of the model (2.1) with $\{N, \sigma, \theta\}$ set to $\{5, 0.5, 10\}$ and observed:

$$[y_1, \ldots, y_5]' = [9.956053,\ 10.058880,\ 10.706742,\ 10.354342,\ 10.609167]'.$$

The likelihood function (2.4) for the experiment is shown in Figure 2.1.

MLE can be found by differentiating the likelihood function and solving for the parameters that yield a derivative of 0. Usually, it is mathematically convenient to maximise the natural logarithm of the likelihood function, which is known as the *log-likelihood* function; this is mathematically consistent because the logarithm is a

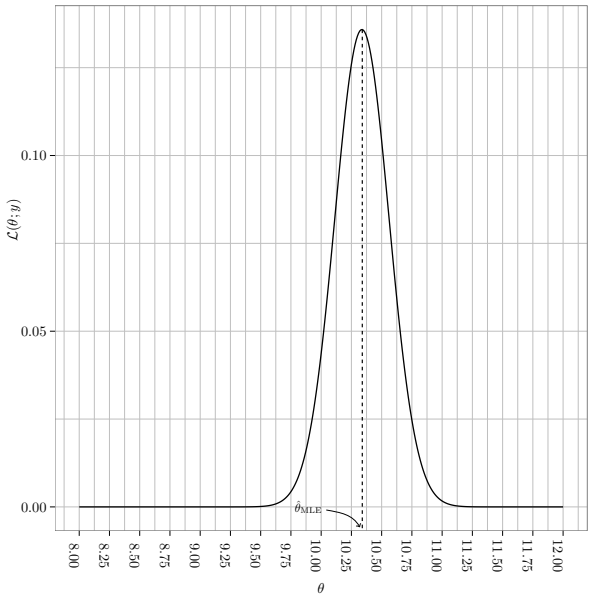


Figure 2.1: The likelihood function for a hypothetical experiment using (2.1) with $\{N, \sigma, \theta\}$ set to $\{5, 0.5, 10\}$.

monotonically increasing function.

For the likelihood function (2.4), the MLE is found as follows:

$$\left[\frac{\partial \log \mathcal{L}(\theta; \mathbf{y})}{\partial \theta}\right]_{\theta=\hat{\theta}_{\text{MLE}}} = 0, \tag{2.5}$$

$$\frac{1}{\sigma^2}\sum_{i=1}^{N}(y_i - \hat{\theta}_{\text{MLE}}) = 0, \tag{2.6}$$

$$\hat{\theta}_{\text{MLE}} = \frac{1}{N}\sum_{i=1}^{N} y_i, \tag{2.7}$$

$$=: \bar{y}. \tag{2.8}$$

It is easily seen that the MLE is an unbiased estimator and the standard deviation of the estimator is $\sigma/\sqrt{N}$.

2.1.2 Bayesian inference of the unobserved parameter θ

In contrast to the classical estimation theory, the Bayesian way of treating unknown parameters is to consider them as random quantities. Any prior knowledge of the unknown parameters is quantified by a probability distribution, known as the *prior*. As an example, suppose we know that a parameter we wish to estimate is restricted to a certain range of values; this could be captured in the prior distri-

bution by assigning 0 probability density for any point outside the said range.

When new information (such as measurements from a sensor) is available, the prior distribution is updated to arrive at the *posterior* distribution. The mathematical framework for obtaining the posterior from the prior (in the light of new measurements) is given by the well known Bayes' theorem, which is named after Rev. Thomas Bayes. However, the current mathematical form of the theorem is due to the French mathematician, Pierre-Simon Laplace. Bayes' theorem states:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}. \tag{2.9}$$

Each of the terms appearing in (2.9) is identified with special terminology:

$p(\theta|y) =$ The posterior distribution,
$p(y|\theta) =$ The likelihood function (same as (2.4), which appears in the context of classical inference theory. This is considered as, a function of θ with y fixed.),
$p(\theta) =$ The prior distribution,
$p(y) =$ The normalising constant or evidence.

Notice that in (2.9), the new information modifies the prior distribution through the likelihood function. The posterior distribution quantifies the knowledge of the parameter θ, upon observing y. In many instances of practical application, the posterior distribution is known only up to the normalising constant; however, that is usually sufficient to obtain posterior estimates such as the posterior mean which minimizes the Mean Squared Error (MSE). Nevertheless, the normalising constant is useful for model comparisons, and predicting future measurements.

Another challenge of practicality in Bayesian inference is that a closed form solution of the posterior density is not available most of the time. However, there are many techniques to obtain approximate solutions to the posterior distribution when an exact solution is not available. With the increasing computing capabilities available today, Monte Carlo methods are becoming a popular option for performing Bayesian inference. In particular, both Sequential Monte Carlo and Markov Chain Monte Carlo methods have evolved as viable options, especially due to the recent developments in parallel computing.

Now, we demonstrate the use of Bayesian inference by applying it to solve the same estimation problem specified in (2.1) and (2.2).

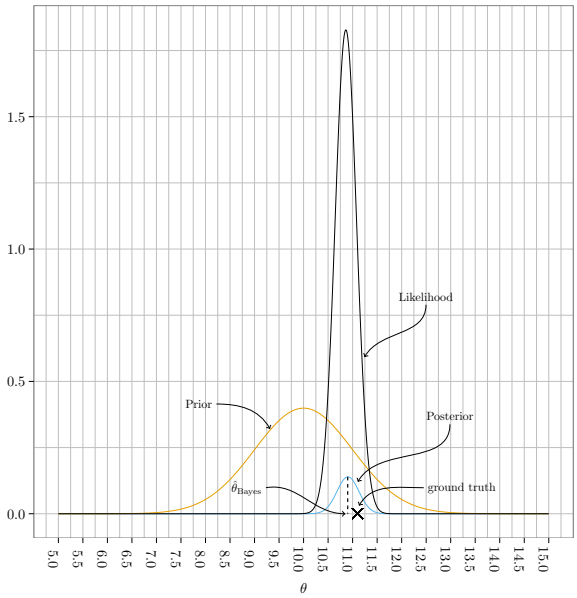


Figure 2.2: An illustration of the likelihood, prior, and posterior distributions for a hypothetical experiment.

Assume that the prior distribution θ is a normal distribution with mean 10 and standard deviation 1; that is

$$p(\theta) = \mathcal{N}(\theta; 10, 1); \tag{2.10}$$

For this example, the posterior distribution $p(\theta|y_1, y_2 \ldots, y_5)$ is readily available through the well known Kalman Filter equations, which we describe later in this chapter:

$$p(\theta|y_1, \ldots, y_5) = \mathcal{N}(\theta; \mathbf{K}(\mathbf{y} - 10\mathbf{H}), 1 - \mathbf{KH}), \tag{2.11}$$

where

$$\mathbf{y} = [y_1\ y_2\ y_3\ y_4\ y_5]', \tag{2.12}$$

$$\mathbf{H} = [1\ 1\ 1\ 1\ 1]', \tag{2.13}$$

$$\mathbf{K} = \mathbf{H}'(\mathbf{H}\mathbf{H}' + \sigma^2\mathbf{I}_5), \tag{2.14}$$

with $\mathbf{I}_5$ denoting the identity matrix of rank 5.

We set $\sigma = 0.5$ as before and run a single realisation of the experiment. The likelihood function, prior, and posterior distribution alongside the true value of θ is shown in Figure 2.2. Notice, how the uncertainty on θ, which is reflected by the spread of the prior distribution, is reduced after observing relatively precise

measurements as evident from the spread of the posterior distribution. The point estimate of θ (chosen as the posterior mean), in this particular example, is approximately the same as that would have been given by the MLE in classical inference. However, Bayesian inference, provides much richer information such as the credible region of an estimate, which can be clearly interpreted probabilistically as a measure of believability based on the prior knowledge and the *observed measurements*.

Particularly motivated by this ability to quantify uncertainty, we follow the Bayesian framework in this thesis.

2.1.3 General recursive Bayesian solution for single target tracking

Consider a single target whose state evolves through time according to a known stochastic differential equation. It is usually convenient to solve the inference problem in discrete time rather than in continuous time. Thus, we focus our efforts on a discrete time target tracking problem (a stochastic differential equation can be discretized by converting it to a stochastic difference equation).

We denote the target state vector at discrete time k as $\mathbf{x}_k$, where $k = 0, 1, 2, 3, \ldots$. The target state may include quantities such as position and velocity that might be of interest for inferential purposes. Following the Bayesian framework, we treat the target state as a random quantity. The target state is directly not observable, but only through the use of some measurements. Denote the measurement vector available at time k by $\mathbf{y}_k$. The trajectory of measurements up to time k is denoted by the vector $\mathbf{y}^k$; that is

$$\mathbf{y}^k = [\mathbf{y}_1' \ \mathbf{y}_2' \ \ldots \ \mathbf{y}_k']'. \tag{2.15}$$

Similarly, the trajectory of the target states up to time k is denoted by $\mathbf{x}^k$; that is

$$\mathbf{x}^k = [\mathbf{x}_1' \ \mathbf{x}_2' \ \ldots \ \mathbf{x}_k']'. \tag{2.16}$$

Assume that the measurement $\mathbf{y}_k$ is related to the target state $\mathbf{x}_k$ through a known deterministic function $\mathbf{h}_k(\mathbf{x}_k)$:

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{w}_k, \tag{2.17}$$

where $\mathbf{w}_k \sim p_w(\cdot)$ is a temporally uncorrelated (that is $\mathbf{w}_k$ is uncorrelated to $\mathbf{w}_l$ for $k \neq l$) white noise process. Further, we assume that the target dynamics follow a first order Markovian process and $\mathbf{x}_k$ is related to $\mathbf{x}_{k-1}$ through a known deterministic function $\mathbf{f}_k(\mathbf{x}_k)$ as

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}) + \mathbf{v}_k, \tag{2.18}$$

where the noise $\mathbf{v}_k$ follows a temporally uncorrelated white random process with $\mathbf{v}_k \sim p_v(\cdot)$.

The state transitions are Markovian in the sense that:

$$p(\mathbf{x}_k|\mathbf{x}^{k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}). \tag{2.19}$$

Upon observing $\mathbf{y}^k$ we are interested in establishing a recursive relationship between the posterior distributions at time k and $k-1$; that is between $p(\mathbf{x}_k|\mathbf{y}^k)$ and $p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})$. From Bayes' theorem, we write $p(\mathbf{x}_k|\mathbf{y}^k)$ as

$$p(\mathbf{x}_k|\mathbf{y}^k) = \frac{p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{y}^{k-1})p(\mathbf{x}_k|\mathbf{y}^{k-1})}{p(\mathbf{y}_k|\mathbf{y}^{k-1})}. \tag{2.20}$$

Assuming that we know the posterior distribution at time $k-1$, that is $p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})$, the posterior at time k is available from (2.20) via a two step procedure: prediction and measurement update.

2.1.3.1 Prediction step

In this step, the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ appearing in (2.20) is obtained. Note that the density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ quantifies the knowledge of the target state at time k after observing measurements *only* up to time $k-1$; hence the name *prediction*. Following the Chapman-Kolmogorov equations, the prediction density is related to $p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})$ by:

$$p(\mathbf{x}_k|\mathbf{y}^{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}^{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}, \tag{2.21}$$

$$= \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}. \tag{2.22}$$

Using a transformation of random variables, the density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ appearing above is obtained from (2.18) as $p_v(\mathbf{x}_k - \mathbf{f}_k(\mathbf{x}_{k-1}))$. Thus, we write the prediction density as

$$p(\mathbf{x}_k|\mathbf{y}^{k-1}) = \int p_v(\mathbf{x}_k - \mathbf{f}_k(\mathbf{x}_{k-1}))p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}. \tag{2.23}$$

2.1.3.2 Measurement update

In this step, the new measurement $\mathbf{y}_k$ is used to adjust the prior belief conveyed through the prediction density. From (2.20), note that the likelihood function $p(\mathbf{y}_k|\mathbf{x}_k)$, which contains the new measurement $\mathbf{y}_k$, when multiplied by the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ returns a quantity proportional to the posterior density

$p(\mathbf{x}_k|\mathbf{y}^k)$; that is

$$p(\mathbf{x}_k|\mathbf{y}^k) \propto p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}^{k-1}), \tag{2.24}$$

$$= p_w(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k))p(\mathbf{x}_k|\mathbf{y}^{k-1}). \tag{2.25}$$

The last line follows from a transformation of random variable principle using (2.17). Note that the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ was calculated in the previous step: prediction. The normalising constant $p(\mathbf{y}_k|\mathbf{y}^{k-1})$, which ensures that the posterior distribution integrates to 1 is given by

$$p(\mathbf{y}_k|\mathbf{y}^{k-1}) = \int p_w(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k))p(\mathbf{x}_k|\mathbf{y}^{k-1})d\mathbf{x}_k. \tag{2.26}$$

Thus, the posterior distribution resulting after the measurement update is

$$p(\mathbf{x}_k|\mathbf{y}^k) = \frac{p_w(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)) \int p_v(\mathbf{x}_k - \mathbf{f}_k(\mathbf{x}_{k-1}))p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}}{\int p_w(\mathbf{y}_k - \mathbf{h}_k(\chi_k)) \left\{ \int p_v(\chi_k - \mathbf{f}_k(\mathbf{x}_{k-1}))p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1} \right\} d\chi_k}. \tag{2.27}$$

This procedure, which we refer to as the general Bayesian filtering recursion, is shown in Figure 2.3.

Note that in the context of tracking a target over time, the posterior distribution at time k is used to derive the prior distribution for time $k+1$. Hence, it is convenient for the prior and the posterior to have the same functional form so that the mathematical manipulations needed to transition from prior to the posterior remain consistent over time. However, in general the posterior and the prior belong to different parametric families of distributions. Under some conditions, it can be guaranteed that both the posterior and the prior are of the same form. This property is known as conjugacy in Bayesian inference. Of particular interest are the conditions that guarantee the prior and the posterior to be Gaussians. The well known Kalman filter provides a closed form solution for the posterior distribution for this case.

The conditions for which the Kalman filter is applicable are:

C1: The state transitions and measurements are linear functions of the state vector; that is

$$\mathbf{x}_k = \mathbf{F}_k\mathbf{x}_{k-1} + \mathbf{v}_k \qquad (\text{note that } \mathbf{f}_k(\mathbf{x}_{k-1}) = \mathbf{F}_k\mathbf{x}_{k-1}),$$

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{w}_k \qquad (\text{note that } \mathbf{h}_k(\mathbf{x}_k) = \mathbf{H}_k\mathbf{x}_k).$$

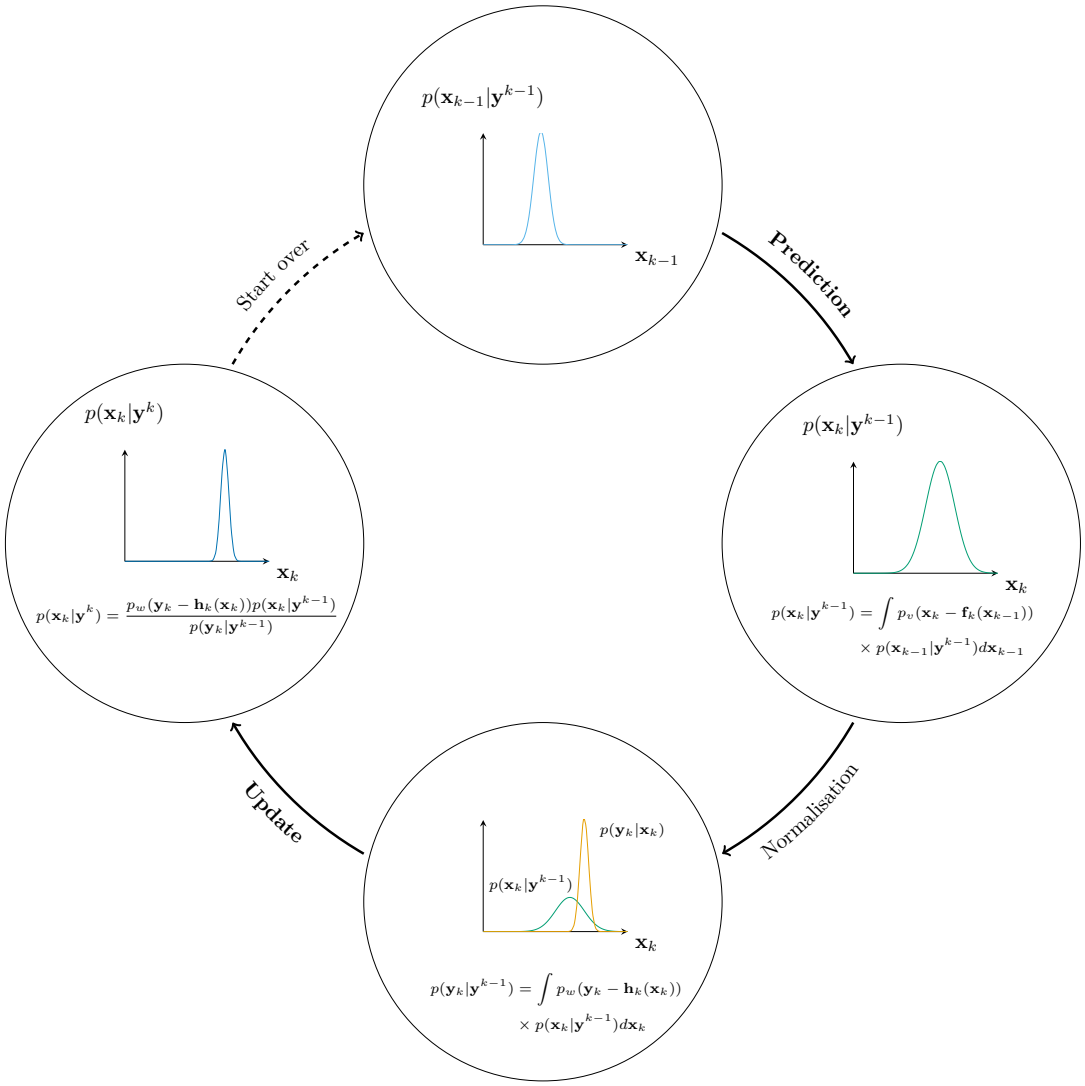


Figure 2.3: An illustration of the general Bayesian recursion for single target tracking.

C2: The noise sequences $\mathbf{v}_k$ and $\mathbf{w}_k$ are from temporally uncorrelated, zero mean, white Gaussian noise processes:

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k),$$
$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k).$$

C3: The prior distribution at time 0, $p(\mathbf{x}_0)$, is Gaussian, with mean $\hat{\mathbf{x}}_{0|0}$ and covariance $\mathbf{P}_{0|0}$.

Next we introduce the Kalman filter and subsequently introduce some of the common suboptimal filters when all or some of the conditions above do not apply.

2.1.3.3 Kalman Filter

The Kalman Filter is named after Rudolf Kálmán, one of the co-inventors of the celebrated signal processing algorithm. The filter is usually expressed as a set of equations, which are known as *Kalman Filter equations*. When all the conditions **C1**, **C2**, and **C3** are valid, the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$, the normalising constant $p(\mathbf{y}_k|\mathbf{y}^{k-1})$, and the posterior $p(\mathbf{x}_k|\mathbf{y}^k)$ are conveniently Gaussian. The Kalman equations provide the exact expressions for the parameters (mean and covariance) of these distributions.

Before deriving the exact expressions, we present the following theorem, which we will use to derive the Kalman filter.

Theorem 2.1. *For* $\mathbf{x}_1, \boldsymbol{\mu}_1 \in \mathbb{R}^{d_1}$, $\mathbf{x}_2 \in \mathbb{R}^{d_2}$, $\mathbf{H} \in \mathbb{R}^{d_2 \times d_1}$, *and covariance matrices* $\mathbf{P}_1$ *and* $\mathbf{P}_2$*:*

$$\mathcal{N}(\mathbf{x}_2; \mathbf{H}\mathbf{x}_1, \mathbf{P}_2)\mathcal{N}(\mathbf{x}_1; \boldsymbol{\mu}_1, \mathbf{P}_1) = \mathcal{N}(\mathbf{x}_2; \mathbf{H}\boldsymbol{\mu}_1, \mathbf{P}_3)\mathcal{N}(\mathbf{x}_1; \boldsymbol{\mu}, \mathbf{P}),$$

where

$$\mathbf{P}_3 = \mathbf{H}\mathbf{P}_1\mathbf{H}' + \mathbf{P}_2,$$
$$\boldsymbol{\mu} = \boldsymbol{\mu}_1 + \mathbf{K}(\mathbf{x}_2 - \mathbf{H}\boldsymbol{\mu}_1),$$
$$\mathbf{P} = \mathbf{P}_1 - \mathbf{K}\mathbf{H}\mathbf{P}_1,$$

with $\mathbf{K} = \mathbf{P}_1\mathbf{H}'\mathbf{P}_3^{-1}$.

Proof. See [Ho 1964]. □

First, we focus on the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$. Note that $p_v(\mathbf{x}_k - \mathbf{f}_k(\mathbf{x}_{k-1})) = \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k\mathbf{x}_{k-1}, \mathbf{Q}_k)$; we apply that on (2.23) and use Theorem 2.1 to obtain an expres-

sion for the prediction density;

$$p(\mathbf{x}_k|\mathbf{y}^{k-1}) = \int p_v(\mathbf{x}_k - \mathbf{f}_k(\mathbf{x}_{k-1}))p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}, \tag{2.28}$$

$$= \int \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k\mathbf{x}_{k-1}, \mathbf{Q}_k)\mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1})d\mathbf{x}_{k-1}, \tag{2.29}$$

$$= \int \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$$
$$\times \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{L}(\mathbf{x}_k - \mathbf{F}_k\hat{\mathbf{x}}_{k-1|k-1}), \tag{2.30}$$
$$\mathbf{P}_{k-1|k-1} - \mathbf{L}_k\mathbf{F}_k\mathbf{P}_{k-1|k-1})d\mathbf{x}_{k-1},$$

$$= \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}), \tag{2.31}$$

where $\mathbf{L}_k = \mathbf{P}_{k-1|k-1}\mathbf{F}_k'(\mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k' + \mathbf{Q}_k)^{-1}$ and

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k\hat{\mathbf{x}}_{k-1|k-1}, \tag{2.32}$$

$$\mathbf{P}_{k|k-1} = \mathbf{Q}_k + \mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k'. \tag{2.33}$$

With the expression for the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ in hand, we move on to obtaining an expression for the normalising constant $p(\mathbf{y}_k|\mathbf{y}^{k-1})$. Note that due to the conditions **C1** and **C2**, we have $p_w(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)) = \mathcal{N}(\mathbf{x}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k)$. Therefore, once again we use Theorem 2.1 as:

$$p(\mathbf{y}^k|\mathbf{y}^{k-1}) = \int p_w(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k))p(\mathbf{x}_k|\mathbf{y}^{k-1})d\mathbf{x}_k, \tag{2.34}$$

$$= \int \mathcal{N}(\mathbf{y}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})d\mathbf{x}_x, \tag{2.35}$$

$$= \mathcal{N}(\mathbf{y}; \hat{\mathbf{y}}_{k|k-1}, \mathbf{S}_k), \tag{2.36}$$

where

$$\hat{\mathbf{y}}_{k|k-1} = \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}, \tag{2.37}$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k' + \mathbf{R}_k. \tag{2.38}$$

We use Theorem 2.1 once more to find the posterior distribution:

$$p(\mathbf{x}_k|\mathbf{y}^k) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}^{k-1})}{p(\mathbf{y}_k|\mathbf{y}^{k-1})}, \tag{2.39}$$

$$= \frac{\mathcal{N}(\mathbf{y}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})}{\mathcal{N}(\mathbf{y}; \hat{\mathbf{y}}_{k|k-1}, \mathbf{S}_k)}, \tag{2.40}$$

$$= \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}), \tag{2.41}$$

where

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}), \tag{2.42}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1}, \tag{2.43}$$

and $\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k'\mathbf{S}_k^{-1}$.

The terms $\hat{\mathbf{y}}_{k|k-1}$, $\mathbf{S}_k$, and $\mathbf{K}_k$ are also identified with the terms: "predicted measurement", "innovation covariance", and "Kalman gain" respectively. The Kalman filter recursion is given by Algorithm 2.1.

Algorithm 2.1: Kalman filter recursion

1 Compute the statistics of the prediction distribution:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k\hat{\mathbf{x}}_{k-1|k-1},$$
$$\mathbf{P}_{k|k-1} = \mathbf{Q}_k + \mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k'.$$

2 Compute the statistics of the predicted measurement density:

$$\hat{\mathbf{y}}_{k|k-1} = \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1},$$
$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k' + \mathbf{R}_k.$$

3 Compute the Kalman gain:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k'\mathbf{S}_k^{-1}.$$

4 Compute the posterior statistics after measurement update:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}),$$
$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1},$$

2.1.3.4 Extended Kalman Filter

The Extended Kalman Filter (EKF) is an extension of the Kalman Filter when condition **C1**, that is when the linearity of the measurement and/or dynamic equations do not hold. In such scenarios, the EKF provides a mechanism for approximating the statistics of the posterior distribution.

The idea is as follows. The EKF relies on the linearisation of the non-linear measurement ($\mathbf{h}_k(\mathbf{x}_k)$) and process ($\mathbf{f}_k(\mathbf{x}_{k-1})$) functions. The linear approximations are obtained by performing Taylor series expansions and ignoring the higher-order

terms. With the linear approximations, the derivation of the EKF equations is the same as that of the Kalman filter.

The linearisation of $\mathbf{f}_k(\mathbf{x}_{k-1})$ by Taylor series expansion around $\hat{\mathbf{x}}_{k-1|k-1}$ results in

$$\mathbf{f}_k(\mathbf{x}_{k-1}) \approx \mathbf{F}_k \mathbf{x}_{k-1} + \boldsymbol{\epsilon}_{\mathbf{f}_k}(\hat{\mathbf{x}}_{k-1|k-1}) + \mathbf{v}_k, \tag{2.44}$$

where

$$\mathbf{F}_k = \nabla_{\mathbf{x}'_{k-1}} \mathbf{f}_k(\mathbf{x}_{k-1}) \Big|_{\mathbf{x}_{k-1}=\hat{\mathbf{x}}_{k-1|k-1}}, \tag{2.45}$$

$$\boldsymbol{\epsilon}_{\mathbf{f}_k}(\mathbf{x}) = \mathbf{f}_k(\mathbf{x}) - \mathbf{F}_k \mathbf{x}, \tag{2.46}$$

with $\nabla_{\mathbf{x}}$ denoting the gradient operation.

Equation (2.44) and assumption **C2** lead to the approximated transition density

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) \approx \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k \mathbf{x}_{k-1} + \boldsymbol{\epsilon}_{\mathbf{f}_k}(\hat{\mathbf{x}}_{k-1|k-1}), \mathbf{Q}_k). \tag{2.47}$$

Theorem 2.1 is used to obtain the prediction density $p(\mathbf{x}_k|\mathbf{y}^{k-1})$:

$$p(\mathbf{x}_k|\mathbf{y}^{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1}) d\mathbf{x}_{k-1}, \tag{2.48}$$

$$\approx \int \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k \mathbf{x}_{k-1} + \boldsymbol{\epsilon}_{\mathbf{f}_k}(\hat{\mathbf{x}}_{k-1|k-1}), \mathbf{Q}_k) \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) d\mathbf{x}_{k-1}, \tag{2.49}$$

$$= \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k \mathbf{x}_{k-1} + \boldsymbol{\epsilon}_{\mathbf{f}_k}(\hat{\mathbf{x}}_{k-1|k-1}), \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}'_k + \mathbf{Q}_k), \tag{2.50}$$

$$= \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}), \tag{2.51}$$

where

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}_k(\hat{\mathbf{x}}_{k-1|k-1}), \tag{2.52}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}'_k + \mathbf{Q}_k. \tag{2.53}$$

The measurement function $\mathbf{h}_k(\mathbf{x}_k)$ is linearised using the Taylor series expansion about $\hat{\mathbf{x}}_{k|k-1}$:

$$\mathbf{h}_k(\mathbf{x}_k) \approx \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\epsilon}_{\mathbf{h}_k}(\hat{\mathbf{x}}_{k|k-1}) + \mathbf{w}_k, \tag{2.54}$$

where

$$\mathbf{H}_k = \nabla_{\mathbf{x}'_k} \mathbf{h}_k(\mathbf{x}_k) \Big|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k-1}}, \tag{2.55}$$

$$\boldsymbol{\epsilon}_{\mathbf{h}_k}(\mathbf{x}) = \mathbf{h}_k(\mathbf{x}) - \mathbf{H}_k \mathbf{x}. \tag{2.56}$$

Algorithm 2.2: Extended Kalman Filter recursion

1 Compute the Jacobian of $\mathbf{f}_k$ at $\hat{\mathbf{x}}_{k-1|k-1}$:

$$\mathbf{F}_k = \nabla_{\mathbf{x}'_{k-1}} \mathbf{f}_k(\mathbf{x}_{k-1}) \Big|_{\mathbf{x}_{k-1}=\hat{\mathbf{x}}_{k-1|k-1}}.$$

2 Compute the statistics of the prediction distribution:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{f}_k(\hat{\mathbf{x}}_{k-1|k-1}), \\ \mathbf{P}_{k|k-1} &= \mathbf{Q}_k + \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}'_k. \end{aligned}$$

3 Compute the Jacobian of $\mathbf{h}_k$ at $\hat{\mathbf{x}}_{k|k-1}$:

$$\mathbf{H}_k = \nabla_{\mathbf{x}'_k} \mathbf{h}_k(\mathbf{x}_k) \Big|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k-1}}.$$

4 Compute the statistics of the predicted measurement density:

$$\begin{aligned} \hat{\mathbf{y}}_{k|k-1} &= \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}), \\ \mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}'_k + \mathbf{R}_k. \end{aligned}$$

5 Compute the posterior statistics after measurement update:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k-1} \mathbf{H}'_k \mathbf{S}_k^{-1} (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}), \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1} \mathbf{H}'_k \mathbf{S}_k^{-1} \mathbf{H}_k \mathbf{P}_{k|k-1}. \end{aligned}$$

The linearised measurement function allows the use of Theorem 2.1 to obtain a closed form (but approximate) solution to the measurement prediction density (normalising constant):

$$\begin{aligned} p(\mathbf{y}_k|\mathbf{y}^{k-1}) &= \int p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{y}^{k-1}) d\mathbf{x}_k, && (2.57) \\ &\approx \int \mathcal{N}(\mathbf{y}_k; \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\epsilon}_{\mathbf{h}_k}(\hat{\mathbf{x}}_{k|k-1}), \mathbf{R}_k) \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}) d\mathbf{x}_k, && (2.58) \\ &= \mathcal{N}(\mathbf{y}_k; \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1} + \boldsymbol{\epsilon}_{\mathbf{h}_k}(\hat{\mathbf{x}}_{k|k-1}), \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}'_k + \mathbf{R}_\mathbf{k}), && (2.59) \\ &= \mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_{k|k-1}, \mathbf{S}_k), && (2.60) \end{aligned}$$

where

$$\begin{aligned} \hat{\mathbf{y}}_{k|k-1} &= \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}), && (2.61) \\ \mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}'_k + \mathbf{R}_\mathbf{k}. && (2.62) \end{aligned}$$

Finally, the statistics of the posterior distribution are approximated as:

$$p(\mathbf{x}_k|\mathbf{y}^k) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}^{k-1})}{p(\mathbf{y}_k|\mathbf{y}^{k-1})}, \tag{2.63}$$

$$\approx \frac{\mathcal{N}(\mathbf{y}_k; \mathbf{H}_k\mathbf{x}_k + \boldsymbol{\epsilon}_{\mathbf{h}_k}(\hat{\mathbf{x}}_{k|k-1}), \mathbf{R}_k)\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})}{\mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_{k|k-1}, \mathbf{S}_k)}, \tag{2.64}$$

$$= \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}), \tag{2.65}$$

where

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k-1}\mathbf{H}_k'\mathbf{S}_k^{-1}(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}), \tag{2.66}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1}\mathbf{H}_k'\mathbf{S}_k^{-1}\mathbf{H}_k\mathbf{P}_{k|k-1}. \tag{2.67}$$

The EKF recursion is summarised in Algorithm 2.2.

2.1.3.5 Unscented Kalman Filter

A drawback of the EKF is the need to compute the Jacobian matrix, which is more often than not computationally demanding. Additionally the accuracy and generality are some what limited in the EKF. The Unscented Kalman Filter (UKF) avoids the need to evaluate the Jacobians; instead the UKF uses a mathematical tool known as the Unscented Transform (UT) to approximate the required statistics appearing in the general Bayesian recursion for target tracking. Generally, the approximations obtained via UT are more accurate on average than the linearised approximations used in the EKF.

The UT is similar in spirit to the method of importance sampling [Robert 2004] where both use a weighted average of a set of discrete points to evaluate moments of a function of a random variable. However, the difference between the two methods stems from the mechanism used to select the discrete points used for weighing. In UT, the points and the associated weights are chosen deterministically. Suppose that we are interested in evaluating the expectation of some function $\mathbf{g}(\boldsymbol{\psi})$ of a random variable $\boldsymbol{\psi}$ with a probability density function $p(\boldsymbol{\psi})$:

$$\hat{\mathbf{g}} = \int \mathbf{g}(\boldsymbol{\psi})p(\boldsymbol{\psi})d\boldsymbol{\psi}. \tag{2.68}$$

The UT approximates $\hat{\mathbf{g}}$ by evaluating the function $\mathbf{g}(\boldsymbol{\psi})$ at specific points known as *sigma points* and assigning weights known as *sigma weights* followed by taking

the weighted average of the function $\mathbf{g}$ evaluated at those sigma points; that is,

$$\hat{\mathbf{g}} \approx \sum_{i=1}^{s} \sigma^i \mathbf{g}(\boldsymbol{\chi}^i), \tag{2.69}$$

where s, σ^i, and $\boldsymbol{\chi}^i$ are the number of sigma points, the i^{th} sigma weight and the i^{th} sigma point respectively.

Let $\boldsymbol{\mu}$ and $\mathbf{C}$ be the mean and covariance of $\boldsymbol{\psi}$; that is,

$$\boldsymbol{\mu} = \int \boldsymbol{\psi} p(\boldsymbol{\psi}) d\boldsymbol{\psi}, \tag{2.70}$$

$$\mathbf{C} = \int (\boldsymbol{\psi} - \boldsymbol{\mu})(\boldsymbol{\psi} - \boldsymbol{\mu})' p(\boldsymbol{\psi}) d\boldsymbol{\psi}, \tag{2.71}$$

$$\tag{2.72}$$

The sigma points and sigma weights are chosen to match the first two moments of $p(\boldsymbol{\psi})$: $\boldsymbol{\mu}$ and $\mathbf{C}$; that is, sigma points and sigma weights should satisfy

$$\boldsymbol{\mu} = \sum_{i=1}^{s} \sigma^i \boldsymbol{\chi}^i, \tag{2.73}$$

$$\mathbf{C} = \sum_{i=1}^{s} \sigma^i (\boldsymbol{\chi}^i - \boldsymbol{\mu})(\boldsymbol{\chi}^i - \boldsymbol{\mu})'. \tag{2.74}$$

There are many methods to select sigma points and sigma weights that satisfy (2.73) and (2.74) [Julier 2000, Lerner 2002].

We now show how the UT is used to approximate the moments of the posterior distribution $p(\mathbf{x}_k|\mathbf{y}^k)$. It can be shown [Anderson 1979, Example 3.2] that the first two moments of the posterior distribution $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$ are given by

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \boldsymbol{\psi}_k \mathbf{S}_k^{-1} (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}), \tag{2.75}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \boldsymbol{\psi}_k \mathbf{S}_k^{-1} \boldsymbol{\psi}_k', \tag{2.76}$$

where

$$\hat{\mathbf{x}}_{k|k-1} = \mathrm{E}(\mathbf{x}_k|\mathbf{y}^{k-1}), \tag{2.77}$$

$$\hat{\mathbf{y}}_{k|k-1} = \mathrm{E}(\mathbf{y}_k|\mathbf{y}^{k-1}), \tag{2.78}$$

$$\boldsymbol{\psi}_k = \mathrm{cov}(\mathbf{x}_k, \mathbf{y}_k|\mathbf{y}^{k-1}), \tag{2.79}$$

$$\mathbf{S}_k = \mathrm{cov}(\mathbf{y}_k|\mathbf{y}^{k-1}), \tag{2.80}$$

with the operators E and cov indicating expectation and covariance respectively. The quantities $\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}, \hat{\mathbf{y}}_{k|k-1}, \boldsymbol{\psi}_k$, and $\mathbf{S}_k$ are generally not known in closed

form when the functions $\mathbf{f}_k$ and $\mathbf{h}_k$ are non-linear. The UT provides approximations for these quantities which are required for calculating the posterior statistics.

Algorithm 2.3: Unscented Kalman Filter recursion

1 Compute the sigma points $\boldsymbol{\chi}^i$ and sigma weights σ_i, for $i = 1, 2, \ldots, s$ to match the prior distribution moments $\hat{\mathbf{x}}_{k-1|k-1}$ and $\mathbf{P}_{k-1|k-1}$.

2 Compute the statistics of the Prediction distribution:

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=1}^{s} \sigma^i \mathbf{f}_k(\boldsymbol{\chi}^i),$$
$$\mathbf{P}_{k|k-1} = \sum_{i=1}^{s} \sigma^i (\mathbf{f}_k(\boldsymbol{\chi}^i) - \hat{\mathbf{x}}_{k|k-1})(\mathbf{f}_k(\boldsymbol{\chi}^i) - \hat{\mathbf{x}}_{k|k-1})' + \mathbf{Q}_k.$$

3 Compute the sigma points $\boldsymbol{\mathcal{Y}}^i$ and sigma weights ϵ_i, for $i = 1, 2, \ldots, s$ to match the predictive distribution moments $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$.

4 Compute the statistics of the predicted measurement density:

$$\hat{\mathbf{y}}_{k|k-1} = \sum_{i=1}^{s} \epsilon^i \mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i),$$
$$\mathbf{S}_k = \mathbf{R}_k + \sum_{i=1}^{s} \epsilon^i (\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})(\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})'.$$

5 Compute the $\boldsymbol{\psi}_k$ matrix:

$$\boldsymbol{\psi}_k = \sum_{i=1}^{s} \epsilon^i (\boldsymbol{\mathcal{Y}}^i - \hat{\mathbf{x}}_{k|k-1})(\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})'.$$

6 Compute the posterior statistics after measurement update:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \boldsymbol{\psi}_k \mathbf{S}_k^{-1} (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}),$$
$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1} \boldsymbol{\psi}_k \mathbf{S}_k^{-1} \boldsymbol{\psi}_k'.$$

First, consider $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$:

$$\hat{\mathbf{x}}_{k|k-1} = \mathrm{E}(\mathbf{f}_k(\mathbf{x}_{k-1}) + \mathbf{v}_k | \mathbf{y}^{k-1}), \tag{2.81}$$
$$= \mathrm{E}(\mathbf{f}_k(\mathbf{x}_{k-1}) | \mathbf{y}^{k-1}), \tag{2.82}$$
$$\approx \sum_{i=1}^{s} \sigma^i \mathbf{f}_k(\boldsymbol{\chi}^i), \tag{2.83}$$
$$\mathbf{P}_{k|k-1} = \mathrm{cov}(\mathbf{f}_k(\mathbf{x}_{k-1}) + \mathbf{v}_k | \mathbf{y}^{k-1}), \tag{2.84}$$
$$= \mathrm{cov}(\mathbf{f}_k(\mathbf{x}_{k-1}) | \mathbf{y}^{k-1}) + \mathbf{Q}_k, \tag{2.85}$$

$$\approx \sum_{i=1}^{s} \sigma^i (\mathbf{f}_k(\boldsymbol{\chi}^i) - \hat{\mathbf{x}}_{k|k-1})(\mathbf{f}_k(\boldsymbol{\chi}^i) - \hat{\mathbf{x}}_{k|k-1})' + \mathbf{Q}_k, \tag{2.86}$$

where $(\boldsymbol{\chi}^i, \sigma^i)$ is the i^{th} sigma point and weight selected to match the first two moments of $p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})$. We again use UT to approximate the remaining quantities needed for approximating the posterior moments. Let $\boldsymbol{\mathcal{Y}}^i$ denote the i^{th} sigma point with weight ϵ^i chosen to match the previously approximated moments of $p(\mathbf{x}_k|\mathbf{y}^{k-1})$ denoted by $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$. Then,

$$\hat{\mathbf{y}}_{k|k-1} = \mathrm{E}(\mathbf{x}_k|\mathbf{y}^{k-1}), \tag{2.87}$$

$$\approx \sum_{i=1}^{s} \epsilon^i \mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i), \tag{2.88}$$

$$\boldsymbol{\psi}_k = \mathrm{cov}(\mathbf{x}_k, \mathbf{y}_k|\mathbf{y}^{k-1}), \tag{2.89}$$

$$\approx \sum_{i=1}^{s} \epsilon^i (\boldsymbol{\mathcal{Y}}^i - \hat{\mathbf{x}}_{k|k-1})(\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})', \tag{2.90}$$

$$\mathbf{S}_k = \mathrm{cov}(\mathbf{y}_k|\mathbf{y}^{k-1}), \tag{2.91}$$

$$\approx \mathbf{R}_k + \sum_{i=1}^{s} \epsilon^i (\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})(\mathbf{h}_k(\boldsymbol{\mathcal{Y}}^i) - \hat{\mathbf{y}}_{k|k-1})'. \tag{2.92}$$

The UKF in algorithmic form is given in Algorithm 2.3.

2.1.4 Particle filtering

Particle filtering is a popular technique to solve challenging non-linear filtering problems. Since we are using particle filtering extensively in this thesis, the topic is discussed in more detail than the tracking methods already described in this section. A particle filter recursively approximates the posterior density using a set of weighted random samples. The introduction presented in this section follows the auxiliary variable implementation of [Pitt 1999]. Note that most popular techniques including the Bootstrap Filter (BF) [Gordon 1993] and the Optimum Importance Density (OID) particle filter [Arulampalam 2002] are covered by the auxiliary variable framework.

Assume that at time $k-1$, the posterior is approximated using Z samples $\mathbf{x}_{k-1}^{(1)}, \ldots, \mathbf{x}_{k-1}^{(Z)}$ and corresponding weights $w_{k-1}^1, \ldots, w_{k-1}^Z$ as follows:

$$p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1}) \approx \sum_{z=1}^{Z} w_{k-1}^z \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(z)}), \tag{2.93}$$

where $\delta(\cdot)$ denotes the Kronecker delta.

Bayes' rule leads to a posterior approximation at time k as shown below:

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{y}^k) &\propto p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}^{k-1}), \\ &= p(\mathbf{y}_k|\mathbf{x}_k)\int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}^{k-1})d\mathbf{x}_{k-1}, \\ &\propto\!\!\!\!\sim \sum_{z=1}^{Z} w_{k-1}^z p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(z)}). \end{aligned} \tag{2.94}$$

Note that (2.94) is only an approximation to the posterior distribution. Let $\hat{p}(\mathbf{x}_k|\mathbf{y}^k)$ denote this approximation and we refer to it as the empirical posterior distribution; that is

$$\hat{p}(\mathbf{x}_k|\mathbf{y}^k) \propto \sum_{z=1}^{Z} w_{k-1}^z p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(z)}). \tag{2.95}$$

Following [Pitt 1999], we reverse the marginalisation in (2.95) and introduce the particle index i as an auxiliary variable. This gives

$$\hat{p}(\mathbf{x}_k, i|\mathbf{y}^k) \propto w_{k-1}^i p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}). \tag{2.96}$$

An approximation to the posterior at time k can be obtained by sampling from (2.96). The auxiliary variable, which can be discarded after sampling is intended to assist in drawing samples of the state vector. Often, it is difficult to sample from $\hat{p}(\mathbf{x}_k, i|\mathbf{y}^k)$, and instead a suitable candidate distribution known as *an importance density* $q(\mathbf{x}_k, i|\mathbf{y}^k)$ is used to obtain samples, which are then appropriately weighted. This procedure of drawing samples from an alternative distribution is known as *importance sampling*. The samples drawn from the importance density is used as follows to obtain the posterior density approximation:

$$\hat{p}(\mathbf{x}_k|\mathbf{y}^k) \propto \sum_{i=1}^{Z} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}), \tag{2.97}$$

where $(\mathbf{x}_k^{(j)}, t_j) \sim q(\mathbf{x}_k, i|\mathbf{y}^k)$ for $j = 1, 2, \ldots, Z$ and

$$w_k^j \propto \frac{p(\mathbf{x}_k^{(j)}, t_j|\mathbf{y}^k)}{q(\mathbf{x}_k^{(j)}, t_j|\mathbf{y}^k)}. \tag{2.98}$$

The particle filtering recursion in algorithmic form is presented in Algorithm 2.4, assuming that at time $k-1$ the posterior approximation $\hat{p}(\mathbf{x}_k|\mathbf{y}^k)$ is available as in (2.93).

The particular choice of the importance density is crucial for a good particle

Algorithm 2.4: Basic particle filter recursion

1 **for** $i = 1$ **to** Z **do**

`/* Generate a new particle and assign a weight */`

2 Draw $(\mathbf{x}_k^{(i)}, t_i) \sim q(\mathbf{x}_k, i|\mathbf{y}^k)$

3 Calculate

$$\tilde{w}_k^i = \frac{w_{k-1}^{t_i} p(\mathbf{y}_k|\mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(t_i)})}{q(\mathbf{x}_k^{(i)}, t_i|\mathbf{y}^k)}$$

4 Normalise the weights:

$$w_k^i = \tilde{w}_k^i \Big/ \sum_{j=1}^{Z} \tilde{w}_k^j \qquad \text{for } i = 1, 2, \ldots, Z.$$

5 Estimate the posterior distribution at time k:

$$\hat{p}(\mathbf{x}_k|\mathbf{y}^k) = \sum_{i=1}^{Z} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}),$$

filter design. The BF is obtained by choosing

$$q_{bs}(\mathbf{x}_k, i|\mathbf{y}^k) \propto w_{k-1}^i p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}). \tag{2.99}$$

A sample $(\tilde{\mathbf{x}}, t)$ can be obtained from $q_{bs}(\mathbf{x}_k, i|\mathbf{y}^k)$ following a two step procedure. First draw an index t with Probability$(t = z) \propto w_{k-1}^z$. Then the target sample $\tilde{\mathbf{x}}$ is drawn from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(t)})$. The main drawback of $q_{bs}(\cdot)$ is that the current measurement $\mathbf{y}_k$ is not used to draw samples. Consequently, the BF is very inefficient in all but very basic practical applications; a relatively large number Z of samples is required to obtain comparable performance with better importance density choices.

A much desired importance density is the OID, which involves drawing samples directly from (2.96). Let $\phi_i = \int p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}) d\mathbf{x}_k$. Then the OID can be derived as follows:

$$\begin{aligned} q_{oid}(\mathbf{x}_k, i|\mathbf{y}^k) &\propto w_{k-1}^i p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}), \\ &= w_{k-1}^i \phi_i \left\{ \frac{p(\mathbf{y}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)})}{\phi_i} \right\}, \\ &= \gamma_i p_i(\mathbf{x}_k|\mathbf{y}_k), \end{aligned} \tag{2.100}$$

where $\gamma_i = w_{k-1}^i \phi_i$, and

$$p_i(\mathbf{x}_k|\mathbf{y}_k) = p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)})/\phi_i.$$

Drawing a sample $(\tilde{\mathbf{x}}, t)$ from q_{oid} is similar to that of q_{bs} except, we first draw t_i with Probability$(t = j) \propto \gamma_j$ followed by drawing $\tilde{\mathbf{x}}$ from $p_t(\mathbf{x}_k|\mathbf{y}^k)$.

It can be easily verified by taking the ratio between $p(\mathbf{x}_k, i|\mathbf{y}^k)$ and $q_{oid}(\mathbf{x}_k, i|\mathbf{y}^k)$, that the importance weights of the samples obtained from the OID are constant irrespective of the value of the sample. It is this property that led to the use of the term OID for the density (2.100). Unlike the BF, the OID has the desirable property of using the current measurement to influence the selection of the particle index t and the state vector. However, often it is difficult to find the OID in closed form.

Two common problems of particle filters are particle degeneracy and impoverishment. Particle degeneracy occurs when after several iterations of the filter, relatively few particles contain significant weight, while the others contribute a negligible mass. A common technique used to counter the degeneracy problem is to introduce a resampling step [Arulampalam 2002]. However, resampling introduces the second problem of particle impoverishment [Arulampalam 2002]; that is, particles losing diversity and ultimately all collapsing to a single point. Two common solutions to the impoverishment problem are the resample-move algorithm [Gilks 2001] and kernel regularisation [Musso 2001]. We only discuss kernel regularisation since it is the method that we have used in this thesis.

2.1.4.1 Kernel regularisation

Kernel regularisation, in the context of particle filtering, means approximating the density of an underlying set of particles using a kernel [Silverman 1986].

First, we look at the concept of kernel estimation of a density. Suppose $\boldsymbol{\nu} \in \mathbb{R}^d$ is a random variable with the probability distribution $p(\boldsymbol{\nu})$. Let $\boldsymbol{\nu}^{(z)} \sim p(\boldsymbol{\nu})$ for $z = 1, 2, \ldots, Z$ be Z samples from $p(\boldsymbol{\nu})$. A question one might face here would be: how to estimate the underlying probability distribution when presented with a set of samples. A crude approximation to $p(\boldsymbol{\nu})$ can be made by a sum of delta functions centered around the sample points; that is,

$$p(\boldsymbol{\nu}) \approx 1/Z \sum_{z=1}^{Z} \delta(\boldsymbol{\nu} - \boldsymbol{\nu}^{(z)}). \tag{2.101}$$

Note that the draws from (2.101) always produce samples from a finite set of points. This can lead to particle impoverishment. The kernel method of approximating the original distribution from samples involves convolving the right hand

side of (2.101) with a kernel function $K_h(\boldsymbol{\nu})$. Here, $K_h(\boldsymbol{\nu})$ is a non-negative valued radially symmetric function, which integrates to unity over the domain of $\boldsymbol{\nu}$. Two common kernel functions are multivariate normal and Epanechnikov functions. For a bandwidth parameter $h > 0$, the kernel approximation to $p(\boldsymbol{\nu})$ is

$$\hat{p}_h(\boldsymbol{\nu}) = \frac{1}{Zh^d} \sum_{z=1}^{Z} K_h \left\{ \frac{1}{h} (\boldsymbol{\nu} - \boldsymbol{\nu}^{(z)}) \right\}. \tag{2.102}$$

Note that unlike the previous approximation (2.102) has continuous support. The bandwidth parameter h effects the approximation performance in two ways, namely: bias and variance. A larger h value will result in a lower variance but at the expense of bias. A trade-off between bias and variance is effected by the choice of h. The common practice, as suggested by Silverman [Silverman 1986], is to choose h that minimises the Mean Integrated Squared Error (MISE)

$$\text{MISE} = E\left[\int (p(\boldsymbol{\nu}) - \hat{p}_h(\boldsymbol{\nu}))^2 d\boldsymbol{\nu} \right]. \tag{2.103}$$

Unfortunately, it can be shown [Silverman 1986] that the optimum choice of h depends on the distribution $p(\boldsymbol{\nu})$ which needs to be estimated. Assume the unknown distribution is a standard multivariate normal distribution; approximate formulae for the optimum bandwidth in the MISE sense for various kernel functions are given in [Silverman 1986, p. 87].

Kernel regularisation in the context of particle filtering simply refers to replacing the discrete distribution (2.97) with its kernel density.

The effectiveness of using a kernel to solve particle impoverishment is best illustrated by a simple example.

Example 2.1.1. *Assume ν_0, ν_1, and $y \in \mathbb{R}$ are three random variables such that*

$$\nu_0 \sim \mathcal{N}(\cdot; \mu, p^2), \tag{2.104}$$

$$\nu_1 = 2\nu_0 + v, \tag{2.105}$$

$$y = \nu_1 + w, \tag{2.106}$$

where noise variables v and w are mutually independent, zero mean, and Gaussian with variances r^2 and q^2 respectively. It is obvious that the estimation problem of finding the posterior mean $p(\nu_1|y)$ can be optimally solved (in MMSE sense) using the Kalman filter, but for demonstration purposes, we use the OID particle filter described earlier.

Let $v_0^{(z)}$ for $z = 1, 2, \ldots, Z$ be Z particles; such that,

$$p(v_0) \approx \frac{1}{Z} \sum_{z=1}^{Z} \delta(v_0 - v_0^{(z)}). \tag{2.107}$$

Use of a Gaussian kernel with a bandwidth parameter h on the particle set results in the following continuous approximation to $p(v_0)$:

$$p(v_0) \approx \frac{1}{Z} \sum_{j=1}^{Z} \mathcal{N}(v_0; v_0^{(i)}, h\sigma^2), \tag{2.108}$$

where σ^2 is the sample covariance. Note that, when h tends to 0, the smoothed density (2.108) tends to the original unsmoothed approximation (2.107).

It can be shown that the OID (2.100) is given by

$$q_{oid}(v_1, i|y) \propto \mathcal{N}(y; 2v_0^{(i)}, \gamma^2)\mathcal{N}(v_1; 2v_0^{(i)} + K\lambda_i, \phi^2), \tag{2.109}$$

where i is the auxiliary variable used to assist in sampling, and

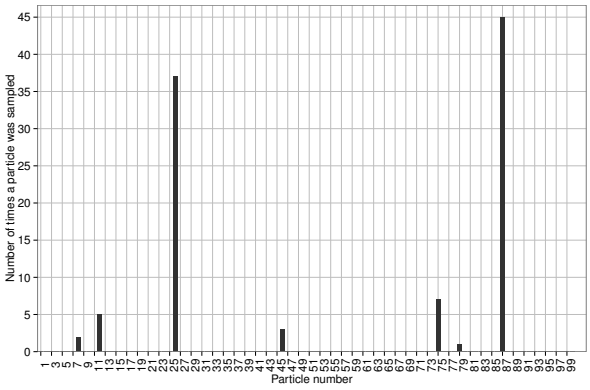
$$\gamma^2 = r^2 + q^2 + 4h^2, \tag{2.110}$$
$$K = (r^2 + 4h^2\sigma^2)\gamma^{-2}, \tag{2.111}$$
$$\lambda_i = y - 2v_0^{(i)}, \tag{2.112}$$
$$\phi^2 = (r^2 + 4h^2\sigma^2)(1 - K). \tag{2.113}$$

As explained earlier, obtaining a sample from $q_{oid}(v_1, i|y)$ involves, first, drawing a particle index i. This is analogous to a resampling step in the Sequential Importance Sampling (SIS) particle filter. If the filter is subjected to severe particle impoverishment; then only a few samples will be selected many times. Figure 2.4 illustrates the effect of bandwidth on particle impoverishment. Simulation parameters were set as follows:

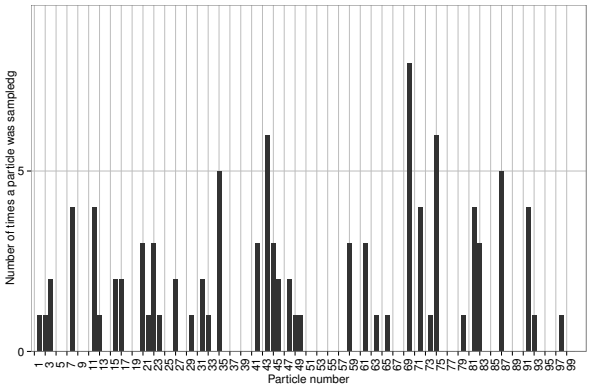
$$\{Z, \mu, p^2, q^2, r^2\} = \{100, 0, 3.5, 0.01, 0.01\}.$$

Figure 2.4(a) is a plot of the number of times each particle index was chosen when sampling 100 times from $q_{oid}(v_1, i|y)$ with $h = 0$ (that is, without any kernel smoothing). But repeating the same experiment with the bandwidth h chosen according to [Silverman 1986, p. 87] results in Figure 2.4(b).

Out of 100 particle indices, only 7 were chosen at least once, when resampled 100 times without using a kernel. This result indicates severe particle impoverishment. On the other hand, 39 indices were chosen at least once, when a smoothing kernel was used, suggesting



(a)



(b)

Figure 2.4: Simulation results: (a) Particle index selection without using a kernel. (b) Particle index selection with a kernel.

the effectiveness of kernel regularisation as a solution to particle impoverishment.

2.2 Radar Fundamentals

The acronym RADAR stands for RAdio Detection And Ranging. Radar technology exploits the properties of Electro Magnetic (EM) waves to interrogate objects located further away. Application areas of radar technology span industries such as aerospace, automotive navigation, defence applications, and meteorology. One of the reasons for such diverse applications is the ability of Radar to operate under all weather conditions.

At a fundamental level, a radar system operates by transmitting an EM signal and subsequently collecting the reflected signals. The received signals are then processed to extract information about the objects that caused the reflections.

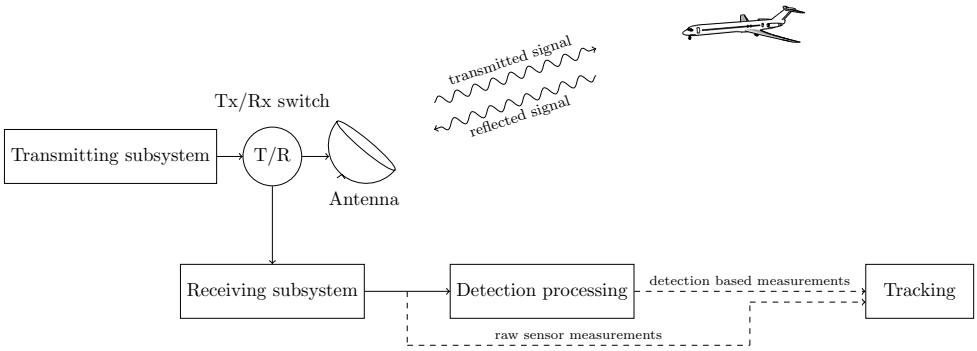


Figure 2.5: A block diagram of a typical monostratic radar system.

The frequency of operation for radar varies depending on the application from 3 MHz (HF) to 110 GHz (W-Band) up to millimetre wave frequencies (100-300 GHz) [Richards 2010].

A radar system consists of three main components: transmitter, receiver, and signal processor. If the transmitter and receiver are co-located then the radar system is said to be Monostatic; the term Bistatic refers to a configuration where the transmitter and receiver are not co-located. Figure 2.5 illustrates how the subsystems are inter-related for a monostatic radar system. Giving comprehensive details about the operation of a radar is beyond the scope of this thesis. Next we focus on developing the necessary background knowledge to complement the main theoretical contributions laid out in the upcoming chapters.

The transmitter subsystem is responsible for preparing and transmitting the EM signal to interrogate the objects of interest in the environment. The transmitted waveform is a baseband signal modulated by a sinusoidal carrier signal that can be represented by the following complex form:

$$g(t) = s(t)\exp(j(\omega_c t + \phi(t))), \tag{2.114}$$

where $s(t)\exp(j\phi(t))$ is the baseband signal and ω_c is the carrier frequency. A rule of thumb is that if the bandwidth of the baseband signal is less than one tenth of the carrier frequency, then the transmitted signal is considered to be narrowband [Levanon 2004]. Throughout the rest of the thesis, we make the narrowband assumption for the transmitted waveform. The signal received back at the receiver is delayed by the time taken for the transmitted signal to traverse the signal path. This delay provides crucial information for the radar system to detect the range of

the object that caused the reflection. The delay is related to the range by

$$\tau = \frac{2R}{C}, \tag{2.115}$$

where we have assumed that the signal travels at the speed of light C. For a narrowband transmitted signal, any motion (relative to the radar system) of the reflecting object induces an apparent frequency shift in the returned signal. This phenomenon is known as the Doppler effect [Levanon 2004] and the frequency change is known as the doppler shift. Doppler shift is given by

$$\nu = \frac{2\pi\dot{R}}{\lambda_c}, \tag{2.116}$$

where $\dot{R}$ denotes the time derivative of range (range rate) and λ_c is the wavelength of the carrier. In modern radar systems the doppler shift is used to distinguish moving objects from the stationary clutter as well as to infer the velocity of the objects. Apart from time delay and doppler effects, the returned signal is also subjected to attenuation. This attenuation is caused by many factors such as atmospheric absorption, distance of travel, reflection losses, etc. The nature of atmospheric absorption is dependent on the frequency of operation of the radar. Typically the radar carrier frequencies are chosen to minimize the absorption losses. The attenuation due to the travelled distance follows the inverse square law. A reflectivity factor of an object is a scalar quantity by which the amplitude of the signal is reduced after reflecting off the object. Ignoring the atmospheric effects, the losses due to the distance travelled and reflection can be written as (for a monostatic system)

$$\alpha = \frac{\epsilon}{R^2}, \tag{2.117}$$

where $\epsilon \in [0, 1]$ is the reflectivity factor of the object.

The returned signal (modulated) which is subjected to time delay, doppler and attenuation is

$$h(t) = \alpha g(t - \tau) \exp(j\nu t). \tag{2.118}$$

Upon receipt of the reflected signal by the receiver subsystem, it is subjected to demodulation, where the baseband signal is extracted from the modulated signal. The extracted baseband signal is then passed through an Analog-to-Digital conversion process to obtain the discrete samples:

$$y[n] = \alpha s(nT - \tau) \exp(\nu nT) + w[n] \text{ for } n = 1, 2, 3, \ldots, \tag{2.119}$$

where T is the sampling period and $w[n]$ is a zero mean IID noise process known as *measurement noise*. Measurement noise arises because of the imperfections in the radar sensors and the Analog-to-Digital conversion process.

The sampled measurements are then passed onto the detection processing (matched filtering) subsystem for generating candidate range and doppler pairs (detections) and/or tracking (filtering). A tracking subsystem typically uses one of the algorithms discussed in Section 2.1.3 to track the state of the objects of interest.

2.2.1 Matched filtering

Matched filtering refers to the process used by radar systems to produce candidate detections. It can be shown [Levanon 2004] that for a signal $s(t)$ superimposed with an Additive White Gaussian Noise (AWGN) process $w(t)$, the impulse response of a filter that maximises the SNR at a given time t_0 is,

$$f(t; t_0) = Ks^*(t_0 - t), \tag{2.120}$$

where K is an arbitrary constant. Without loss of generality we set $K = 1$ and $t_0 = 0$. Note that the impulse response is proportional to the time reversed and delayed original signal $s(t)$; hence the rationale for the use of the term *matched filter*. The time domain response of the filter (2.120) at lag τ for a doppler shifted signal $s(t)\exp(j2\pi\nu_0 t)$ is

$$s_o(\tau, \nu) = \int s(t)\exp(j\nu t)s^*(t - \tau)dt. \tag{2.121}$$

When viewed as a function of two parameters τ and ν, the output of the matched filter will be maximized when $\tau = 0$ and $\nu = 0$. This is not surprising because when $\tau = 0$ and $\nu = 0$ the input signal is exactly matched to the underlying filter. If the input signal to the matched filter is delayed by τ_0, the peak in the output function $s_o(\tau, \nu)$ will be shifted by τ_0 in the delay dimension. In an actual use case, the delay and doppler shift of the received signal are not known; hence the received signal is passed through a bank of matched filters, where each filter is designed (matched) to maximise the SNR at a distinct delay (t_0) and doppler (ν) pair. In practice the detection is performed on a discretized grid on the delay and doppler dimensions, with the grid spacings chosen according to the requirements of the radar application. A target is declared at a particular delay doppler pair if the matched filter output at that point is greater than a pre-selected threshold. This threshold is chosen to satisfy the required probability of detection and/or

false alarms [Kay 1998].

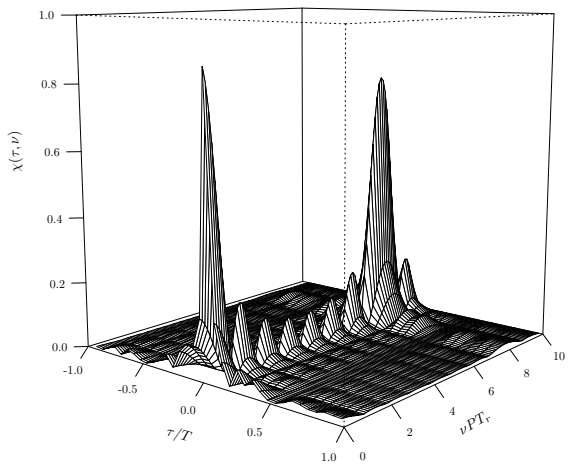


Figure 2.6: Ambiguity function for $P = 8$ Linear Frequency Modulated pulses where the duty cycle $T/T_r = 1/9$ with $T_r = 1/1.25$. The signal parameters used in this example are borrowed from [Levanon 2004].

Note that the peak of (2.121) does not depend on the shape of the waveform but rather on the energy of the transmitted signal $s(t)$ and the measurement noise power. However, the shape of the waveform influences the response at points off the peak. It is desired from a matched filter to have a sharp (peak) response at the corresponding delay-doppler of a received signal and a very low response at off-peak points. One of the main challenges in radar waveform design is to invent waveforms that result in matched filter responses which fall-off quickly from the peak response; this helps in separating out targets that are close to each other (in either range or doppler). The ability of a waveform to separate a target in range and doppler is measured by the range and doppler resolution of the waveform respectively. A common way of increasing the range resolution is to use pulse compression by linear frequency modulation, which is also known as *chirping* [Levanon 2004]. The doppler resolution is typically increased by processing a stream of pulses instead of a single pulse. An interesting recent contribution in a waveform design with a high doppler resolution is found in [Pezeshki 2007].

The tool used to analyse the range and doppler resolution of a waveform is

known as the *ambiguity function* and is given by

$$\chi(\tau, \nu) = \left| \int s(t) \exp(j\nu t) s^*(t - \tau) dt \right|. \tag{2.122}$$

Properties of the ambiguity function are discussed in [Levanon 2004]. The ambiguity function for a train of Linear Frequency Modulated (LFM) pulses is shown in Figure 2.6.

2.2.2 Raw sensor measurements vs. detection based measurements for tracking

As shown by dashed lines in Figure 2.5 the input to the tracking subsystem can either be supplied from the output of the matched filter or directly from the output of the receiving subsystem. In the former case the tracking algorithms use the candidate range/doppler points as measurements. We refer to this type of measurements as *detection based measurements*. The second type of measurements, which is the output of the receiving subsystem, is referred to as *raw sensor measurements*. It has been shown in [Morelande 2007] that an improved Posterior Cramér-Rao Bound [Van Trees 1968, Tichavsky 1998] can be obtained by using raw sensor measurements instead of detection based measurements. This is intuitively understandable through a well known result in information theory, which states that the entropy of a random variable could only decrease by passing through a deterministic transformation.

CHAPTER 3

Posterior Cramér-Rao bounds for multipath radar tracking

Summary

Tracking in a multipath environment poses many challenges. It is important to quantify the achievable performance bounds in such an environment. Finding the performance bounds can be challenging because it requires calculation of derivatives of parameters that are functions of reflection points with respect to target related quantities. We propose a method to calculate a lower bound for the Mean Squared Error (MSE) in target state estimation in a multipath environment. A novel measurement model is introduced involving a general abstraction of a realistic multipath environment. In this model, we have incorporated some of the inherent uncertainty in the environment typically not accounted for in conventional problem formulations such as the uncertainty because of imprecision in maps of the surveillance area and in the reflectivity factors of the reflective surfaces.

3.1 Introduction

In this chapter, we introduce a novel model for tracking in a partially known environment and obtain the Posterior Cramér Rao lower Bound (PCRB) for the estimation problem. The novelty of the model is the acknowledgement of the imperfect knowledge of the multipath environment and the accounting for the resulting uncertainty by the introduction of appropriate random components to the measurement equation.

The Cramér-Rao Lower Bound (CRB) provides a lower bound on the variance of unbiased estimators of deterministic parameters. The CRB is explicitly given by the inverse of the Fisher Information Matrix (FIM) $\mathcal{I}(\cdot)$ [Kay 1998], which for a deterministic parameter $\mathbf{x}$ and measurement vector $\mathbf{y}$ is defined as

$$\mathcal{I}(\mathbf{x}) = \int \frac{\partial \log p(\mathbf{y};\mathbf{x})}{\partial \mathbf{x}} p(\mathbf{y};\mathbf{x}) d\mathbf{y}. \tag{3.1}$$

The Bayesian counterpart of the lower bound is known as the PCRB or Bayesian CRB. More specifically, the PCRB is the lower bound on the MSE of estimators for random parameters. Note that the notion of "unbiasedness" is not present in the description as compared to that of the CRB [Gill 1995]. PCRB is also specified by the inverse of the FIM, which in a Bayesian setting is defined by

$$J = \int \frac{\partial \log p(\mathbf{y}, \mathbf{x})}{\partial \mathbf{x}} p(\mathbf{y}, \mathbf{x}) d\mathbf{y} d\mathbf{x}. \tag{3.2}$$

Note that the expectation is over both the measurement $\mathbf{y}$ and the parameter $\mathbf{x}$ in (3.2). In a Bayesian filtering problem, the calculation (3.2) becomes progressively more difficult over time because of the unbounded growth in the dimension of the measurement trajectory vector. However, Tichavsky *et al* [Tichavsky 1998] have proposed an elegant method for efficiently calculating the PCRB, by employing a recursive procedure. We use this method to find the PCRB for our multipath filtering problem.

The performance bounds in the context of multipath radar appear in numerous research findings [Rendas 1991, Hamilton 1992, Ianniello 1986, Hayvaci 2012b]. In [Hamilton 1992, Hayvaci 2012b, Ianniello 1986] the analysis is restricted to two multipaths. Thus, obtaining the derivatives for the CRB is not mathematically complicated, particularly when the number of allowed reflection points in a multipath is one. In [Rendas 1991] an arbitrary number of multipaths is allowed in the measurement model, but no method is proposed to calculate the required derivatives for evaluating the CRB. The existing work could also be categorised based on the nature of sensing: active [Hayvaci 2012b] or passive [Hamilton 1992, Ianniello 1986, Rendas 1991]. A common feature in [Rendas 1991, Hamilton 1992, Ianniello 1986, Hayvaci 2012b] is the treatment of unknown parameters as deterministic quantities, which prevents the assignment of probabilistic measures to reflect the prior knowledge. Additionally, these research efforts do not consider the tracking problem where localizing a target over time is of interest.

Most of the multipath tracking solutions that we review in Chapters 5 and 6 use detection based measurements for the tracking algorithm. However, as mentioned in the Chapter 2, it has been shown [Morelande 2007] that direct supply of raw sensor data as measurements to a filter results in improved performance bounds. Further, the data association problem, present when using detections based measurements, is no longer applicable when the sensor data are directly used. However, the drawback of using raw measurements for filtering is the need for computationally intensive calculations, particularly because of the large size of the measurement. With the enormous processing capabilities of modern technolo-

gies such as Graphical Processing Units (GPU), the computational demand for raw sensor data processing is no longer a problem. In our work on multipath radar appearing in this chapter, we use raw sensor measurements as the measurements for the tracking algorithm.

In this chapter, we present a general model for a multipath environment containing a dynamic target and concentrate on deriving a performance bound (PCRB) for the tracking problem. We do not impose any restriction on the number of multipaths or the number of walls in the environment. In Chapter 5, we design a stochastic filter for the model. One of the main challenges in deriving the PCRB for our model is the need to calculate the derivatives of the likelihood function with respect to the reflection points. We achieve this by exploiting recursive relationships between reflection points induced by the specular nature of multipath reflections. In developing the multipath model we have emphasised the inherent uncertainty present in a realistic multipath environment, which is not typically accounted for in conventional multipath models. In many multipath radar applications, the knowledge of the location of the reflecting objects (such as walls) is assumed to be available through a map of the terrain [Chakraborty 2010, Chakraborty 2011, Barbosa 2008]. Usually, this knowledge of the locations is accepted without questioning the accuracy of the map. However, even when the reflecting environment is ostensibly known, we believe that it is important to account for the small errors (of the scale of few wavelengths) in a realistic model. Since this error in location translates to a phase shift in the received signal, we capture this uncertainty by introducing a uniformly distributed random phase shift to each of the multipaths. Another source of uncertainty that we have accounted for originates from the reflectivity factors of the multipath causing obstacles. A reflectivity factor is the fraction of the amplitude of the incident signal which is retained in the reflected signal. In conventional multipath radar problem formulations, this fraction is either assumed to be known or taken as 1. We model the reflectivity factors of the walls as random parameters following a Gaussian distribution.

The rest of this chapter is organised as follows. Section 3.2 introduces the dynamic model as well as the measurement model for the partially known multipath environment [1]. Section 3.3 presents the procedure for calculating the PCRB for the filtering problem along with the details on how to calculate the various derivatives required to compute the PCRB. Section 3.4 is dedicated to demonstrating the PCRB bound through simulation examples along with a supplementary discussion. Finally, concluding remarks for this chapter are contained in Section 3.5.

[1] The term "partially known" is used because, we assume that a map of the area is available but it is accurate only up to few wavelengths.

Some of the notations and acronyms we have adopted are given in Table 3.1. Any notation/acronym, which does not appear in Table 3.1 is defined when first introduced.

Table 3.1: Summary of common notations and acronyms.

Notation/Acronym	Description
δ	Kronecker delta.
$\mathbf{I}$	Identity matrix. The dimensions would be implied by the context unless explicitly stated through a subscript.
$\mathbf{0}$	Zero matrix. The dimensions would be implied by the context unless explicitly stated through a subscript.
$\otimes$	Kronecker product
$\mathbf{y}^k$	$[\mathbf{y}_1' \ \mathbf{y}_2' \ \ldots, \mathbf{y}_k']'$
$\mathbf{\Theta}^k$	$[\mathbf{\Theta}_0', \mathbf{\Theta}_1' \ \ldots, \mathbf{\Theta}_k']'$
$\mathcal{N}(\cdot; \mathbf{\Delta}, \mathbf{\Lambda})$	Multivariate Gaussian distribution with parameters $\mathbf{\Delta}$ (mean) and $\mathbf{\Lambda}$ (covariance)
$\mathcal{CN}(\cdot; \mathbf{\Delta}, \mathbf{\Lambda})$	Circular symmetric complex Gaussian distribution with parameters $\mathbf{\Delta}$ (mean) and $\mathbf{\Lambda}$ (covariance)
$\mathcal{U}_{(\alpha,\beta)}$	Uniform distribution in (α, β).
AOA	Angle of Arrival
CRB	Cramér Rao lower Bound
MSE	Mean Squared Error
PCRB	Posterior CRB
RMSE	Root MSE

3.2 Modelling and notation

Consider a target travelling in an urban terrain. Multiple radar transmitters are placed at suitable locations to illuminate the radar environment. The radar sensors receive a superposition of multiple signals due to scattering of the transmitted signal. We introduced the term "multipath" in Chapter 1. Recall that a signal which has been in contact with the target as well as some reflective surfaces in the environment is referred to as a "multipath". The path which only hits the target along the way is called a "direct path". Figure 3.1 illustrates that two possible paths reflecting off the target. The first is a direct path, whereas the second is an example of a multipath.

The target state (in a 2-dimensional world) at time t_k is denoted by $\mathbf{x}_k = [x_k \ \dot{x}_k \ y_k \ \dot{y}_k]'$, where (x_k, y_k) are the target coordinates (position) in the Cartesian plane and

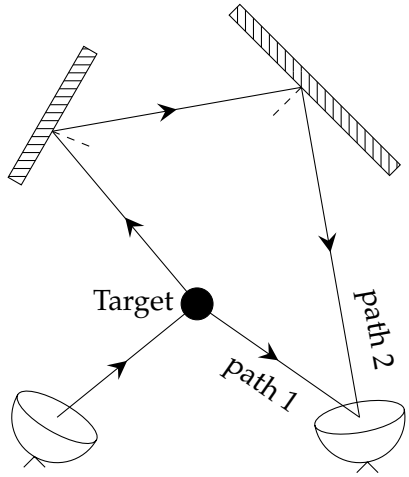


Figure 3.1: An example of multipath reflections.

$(\dot{x}_k, \dot{y}_k)$ are the respective velocities. The target state is assumed to transition from time t_{k-1} to t_k according to the following dynamic equation:

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{w}_k, \qquad k = 1, 2, \ldots, \tag{3.3}$$

where

$$\mathbf{F}_k = \mathbf{I}_2 \otimes \begin{bmatrix} 1 & \tilde{T} \\ 0 & 1 \end{bmatrix}, \tag{3.4}$$

with $\mathrm{cov}(\mathbf{w}_{k_1}, \mathbf{w}_{k_2}) = \delta_{k_1 - k_2} \mathbf{Q}_k$ and

$$\mathbf{Q}_k = \mathbf{I}_2 \otimes \kappa \begin{bmatrix} \tilde{T}^3/3 & \tilde{T}^2/2 \\ \tilde{T}^2/2 & \tilde{T} \end{bmatrix}. \tag{3.5}$$

Here, $\tilde{T} = t_k - t_{k-1}$ is the state sampling period and κ is a noise intensity parameter. The prior distribution for $\mathbf{x}_0$ is assumed to be

$$\mathbf{x}_0 \sim \mathcal{N}(\cdot; \gamma_0, \mathbf{P}_0). \tag{3.6}$$

Consider the following setup:

- N transmitters are placed at suitable locations.
- Return signals are received by M uniform linear arrays with the m^{th} array composed of L_m elements.

- During the sampling period each transmitter transmits a sequence of P pulses of duration D and period T_1, with the first pulse being transmitted at t_k, for $k = 1, 2, 3, \ldots$
- Let $s_{k,n}$ denote the signal transmitted by the n^{th} transmitter at time t_k.
- Let the total number of paths between the n^{th} transmitter and the m^{th} receiver be denoted by $P_{n,m}$.
- Incoming data are sampled every T_2 seconds.
- The reflections off the walls are specular (that is, mirror reflections).
- The transmitters and receivers are coherent.
- The target is a point scatterer.
- There are B buildings in the terrain, which are numbered from 1 to B.
- When a path hits the b^{th} wall, the signal attenuates by a random reflectivity factor, which is distributed according to $\mathcal{N}(\cdot; \mu_b, \iota_b)$, where the prior parameters μ_b and ι_b are known.
- When a multipath hits the target, the signal attenuates by a random target reflectivity factor, which has the distribution $\mathcal{N}(\cdot; \mu_0, \iota_0)$, where the prior parameters μ_0 and ι_0 are known.
- The wall and target reflectivity factors are contained in a vector $\mathbf{z}_k$, and are assumed to be temporally uncorrelated. The diagonal covariance matrix of $\mathbf{z}_k$ is denoted by $\mathbf{Q}_{k,\mathbf{z}}$.
- Because of the uncertainty of the wall locations of the order of a wavelength, the radar signal corresponding to the p^{th} path between the n^{th} transmitter and the m^{th} receiver is phase shifted by $\psi^p_{n,m}$, where

$$\psi^p_{n,m} \sim \mathcal{U}_{[0,2\pi)}.$$

  The phase variables at time k for each transmitter-receiver pair are contained in the vector $\boldsymbol{\psi}_k$.

- Let $\boldsymbol{\Theta}_k$ denote the vector consisting of target kinematics $\mathbf{x}_k$, phase variables $\boldsymbol{\psi}_k$, and reflectivity factors $\mathbf{z}_k$; that is

$$\boldsymbol{\Theta}_k = [\mathbf{x}'_k \ \boldsymbol{\psi}'_k \ \mathbf{z}'_k]'.$$

The signal vector received by all the sensors at time $t_k + uT_2$ is

$$\mathbf{y}_k(u) = \mathbf{h}(\boldsymbol{\Theta}_k; u) + \mathbf{e}(u), \text{ for } u = 0, \ldots, U-1, \tag{3.7}$$

where $\mathbf{e}(u)$ is a circularly symmetric complex white Gaussian process with covariance matrix $2\sigma^2\mathbf{I}$, and

$$\mathbf{h}(\boldsymbol{\Theta}_k; u) = \sum_{n=1}^{N} \begin{bmatrix} \boldsymbol{\mu}_{k,n,1}(\boldsymbol{\Theta}_k; uT_2) \\ \vdots \\ \boldsymbol{\mu}_{k,n,M}(\boldsymbol{\Theta}_k; uT_2) \end{bmatrix}, \tag{3.8}$$

with the measurement function $\boldsymbol{\mu}_{k,n,m}(\cdot)$ for $n = 1, \ldots, N$ and $m = 1, \ldots, M$, given by

$$\boldsymbol{\mu}_{k,n,m}(\boldsymbol{\Theta}; t) = \sum_{p=1}^{P_{n,m}} \mathbf{g}^p_{n,m}(\mathbf{x}, \mathbf{z}; t) \exp(j\psi^p_{n,m}), \tag{3.9}$$

and

$$\mathbf{g}^p_{n,m}(\mathbf{x}, \mathbf{z}; t) = \alpha^p_{n,m}(\mathbf{x}, \mathbf{z}) s_{k,n}(t - \tau^p_{n,m}(\mathbf{x})) e^{jv^p_{n,m}(\mathbf{x})t} \mathbf{a}_m(t; \theta^p_{n,m}(\mathbf{x}), \dot{\theta}^p_{n.m}(\mathbf{x})), \tag{3.10}$$

with

$$\mathbf{a}_m(t; \theta, \dot{\theta}) = \begin{bmatrix} 1 & e^{-j\bar{d}_m r_t} & \cdots & e^{-j(L_m-1)\bar{d}_m r_t} \end{bmatrix}', \tag{3.11}$$

where

$$r_t = \cos\theta - \dot{\theta}t \sin\theta, \tag{3.12}$$

$$\bar{d}_m = 2\pi d_m / \lambda, \tag{3.13}$$

$$d_m = \text{the separation between the elements of the } m^{th} \text{ sensor array}, \tag{3.14}$$

$$\lambda = \text{the wavelength of the carrier signal.} \tag{3.15}$$

Note that we have made the approximation $\cos(\theta + \dot{\theta}t) \approx \cos\theta - (\sin\theta)\dot{\theta}t$. This is valid if $\dot{\theta}t$ is small throughout the surveillance duration [Vincent 2000].

For the p^{th} path between the n^{th} transmitter and the m^{th} sensor array:

- $\alpha^p_{n,m}(\mathbf{x}, \mathbf{z})$ is the intensity of the return signal. This includes transmitted signal strength as well as path attenuation.
- $\theta^p_{n,m}(\mathbf{x})$ is AOA.
- $\dot{\theta}^p_{n,m}(\mathbf{x})$ is the rate of change of the AOA.
- $\tau^p_{n,m}(\mathbf{x})$ is the delay of the signal.

- $\nu_{n,m}^{p}(\mathbf{x})$ is the Doppler shift.

The mathematical expressions for the above five quantities are given in Section 3.3 (A much simpler measurement equation, without any phased array elements and a random phase shift, is explained in Chapter 2.)

The entire measurement vector at time t_k is

$$\mathbf{y}_k = \mathbf{h}(\mathbf{\Theta}_k) + \mathbf{e}, \tag{3.16}$$

where

$$\begin{aligned} \mathbf{y}_k &= [\mathbf{y}_k(0) \dots \mathbf{y}_k(U-1)]', \\ \mathbf{h}(\mathbf{\Theta}_k) &= [\mathbf{h}(\mathbf{\Theta}_k; 0) \dots \mathbf{h}(\mathbf{\Theta}_k; U-1)]', \\ \mathbf{e} &= [\mathbf{e}(0) \dots \mathbf{e}(U-1)]'. \end{aligned}$$

Let $\mathbf{y}^k = [\mathbf{y}_1' \ \mathbf{y}_2' \ \dots, \mathbf{y}_k']'$ denote the vector of all measurements up to and including time k. The target tracking problem is to estimate the target state $\mathbf{x}_k$ after observing $\mathbf{y}^k$.

3.3 Theory/Methodology

3.3.1 Recursive calculation of the Information Matrix

Let the joint distribution of $\mathbf{\Theta}^k$ and $\mathbf{y}^k$ be denoted by p_k; then, from recursive application of Bayes' rule,

$$p_k(\mathbf{\Theta}^k, \mathbf{y}^k) = p(\mathbf{\Theta}_0) \left\{ \prod_{i=1}^{k} p(\mathbf{y}_i | \mathbf{\Theta}_i) \right\} \left\{ \prod_{i=1}^{k} p(\mathbf{\Theta}_i | \mathbf{\Theta}_{i-1}) \right\}. \tag{3.17}$$

We decompose $\mathbf{\Theta}^k$ into two components: $\mathbf{\Theta}_k$ and $\mathbf{\Theta}^{k-1}$:

$$\mathbf{\Theta}^k = [(\mathbf{\Theta}^{k-1})' \ \mathbf{\Theta}_k']'. \tag{3.18}$$

The information matrix corresponding to $\mathbf{\Theta}^k$ and $\mathbf{y}^k$ is given by

$$\mathbf{J}(\mathbf{\Theta}^k) = \begin{bmatrix} E\left\{ -\Delta_{\mathbf{\Theta}^{k-1}}^{\mathbf{\Theta}^{k-1}} p_k \right\} & E\left\{ -\Delta_{\mathbf{\Theta}^{k-1}}^{\mathbf{\Theta}_k} p_k \right\} \\ E\left\{ -\Delta_{\mathbf{\Theta}_k}^{\mathbf{\Theta}^{k-1}} p_k \right\} & E\left\{ -\Delta_{\mathbf{\Theta}_k}^{\mathbf{\Theta}_k} p_k \right\} \end{bmatrix}, \tag{3.19}$$

$$\triangleq \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k \\ \mathbf{B}_k' & \mathbf{C}_k \end{bmatrix}. \tag{3.20}$$

It can be shown that, under some regularity conditions [Tichavsky 1998], the MSE of the estimator of $\mathbf{\Theta}_k$ is lower bounded by the bottom-right matrix block of $\left\{\mathbf{J}(\mathbf{\Theta}^k)\right\}^{-1}$; that is,

$$E\left\{(\mathbf{\Theta}_k - \hat{\mathbf{\Theta}}_k)(\mathbf{\Theta}_k - \hat{\mathbf{\Theta}}_k)'\right\} \geq (\mathbf{C}_k - \mathbf{B}_k'\mathbf{A}_k^{-1}\mathbf{B}_k)^{-1}, \tag{3.21}$$

where $\hat{\mathbf{\Theta}}_k$ is an estimator of $\mathbf{\Theta}_k$. Let the matrix $(\mathbf{C}_k - \mathbf{B}_k'\mathbf{A}_k^{-1}\mathbf{B}_k)$ be called the "information submatrix" for $\mathbf{\Theta}_k$ and denoted by $\mathbf{J}_k$. That is,

$$\mathbf{J}_k = \mathbf{C}_k - \mathbf{B}_k'\mathbf{A}_k^{-1}\mathbf{B}_k. \tag{3.22}$$

Note that, to compute $\mathbf{J}_k$, one would have to either compute the inverse of $\mathbf{J}(\mathbf{\Theta}^k)$, extract the bottom-right sub-matrix followed by a matrix inversion of the extracted sub-matrix or alternatively exploit (3.22) where the size of the $\mathbf{A}_k$ matrix that needs to be inverted is smaller than the full matrix $\mathbf{J}(\mathbf{\Theta}^k)$. However, both approaches are not feasible since, as time k tends to infinity, the size of the matrices which need to be inverted increases. Tichavsky *et al* [Tichavsky 1998] proposed an elegant recursive solution to this problem which we now state as a theorem.

Theorem 3.1. *The sequence* $\{\mathbf{J}_k\}$ *of posterior information submatrices for estimating* $\mathbf{\Theta}_k$ *follows the recursion:*

$$\mathbf{J}_k = \mathbf{D}_k^{22} - \mathbf{D}_k^{21}(\mathbf{J}_{k-1} + \mathbf{D}_k^{11})^{-1}\mathbf{D}_k^{12}, \tag{3.23}$$

where

$$\begin{aligned}
\mathbf{D}_k^{11} &= E\{-\Delta_{\mathbf{\Theta}_{k-1}}^{\mathbf{\Theta}_{k-1}} \log p(\mathbf{\Theta}_k|\mathbf{\Theta}_{k-1})\},\\
\mathbf{D}_k^{12} &= E\{-\Delta_{\mathbf{\Theta}_{k-1}}^{\mathbf{\Theta}_k} \log p(\mathbf{\Theta}_k|\mathbf{\Theta}_{k-1})\},\\
\mathbf{D}_k^{21} &= E\{-\Delta_{\mathbf{\Theta}_k}^{\mathbf{\Theta}_{k-1}} \log p(\mathbf{\Theta}_k|\mathbf{\Theta}_{k-1})\} = \{\mathbf{D}_k^{12}\}',\\
\mathbf{D}_k^{22} &= E\{-\Delta_{\mathbf{\Theta}_k}^{\mathbf{\Theta}_k} \log p(\mathbf{\Theta}_k|\mathbf{\Theta}_{k-1})\} + E\{-\Delta_{\mathbf{\Theta}_k}^{\mathbf{\Theta}_k} \log p(\mathbf{y}_k|\mathbf{\Theta}_k)\},
\end{aligned}$$

provided that the derivatives, expectations, and matrix inversions appearing above exist.

Proof. See [Tichavsky 1998]. □

For the model specified in Section 3.2 the matrices $\mathbf{D}_k^{11}, \mathbf{D}_k^{12}$, and $\mathbf{D}_k^{21}$ can be obtained easily using matrix derivatives. We found the framework presented in [Macrae 1974] to be convenient to perform the required matrix differentiations.

The derivatives are

$$\mathbf{D}_k^{11} = \text{blkdiag}(\mathbf{F}_k' \mathbf{Q}_k^{-1} \mathbf{F}_k, \mathbf{0}), \tag{3.24}$$

$$\mathbf{D}_k^{12} = \text{blkdiag}(-\mathbf{F}_k \mathbf{Q}_k^{-1}, \mathbf{0}), \tag{3.25}$$

$$\mathbf{D}_k^{21} = \{\mathbf{D}_k^{12}\}', \tag{3.26}$$

where the operator blkdiag denotes concatenating the matrices given as arguments diagonally.

The first term of $\mathbf{D}_k^{22}$, which we will denote by $\mathbf{D}_{k,1}^{22}$, can be similarly shown to be

$$\mathbf{D}_{k,1}^{22} = \text{E}\{-\Delta_{\boldsymbol{\Theta}_k}^{\boldsymbol{\Theta}_k} \log p(\boldsymbol{\Theta}_k | \boldsymbol{\Theta}_{k-1})\}, \tag{3.27}$$

$$= \text{blkdiag}(-\mathbf{F}_k \mathbf{Q}_k^{-1}, \mathbf{0}, \mathbf{Q}_{k,\mathbf{z}}^{-1}). \tag{3.28}$$

The second term of $\mathbf{D}_k^{22}$, which we denote by $\mathbf{D}_{k,2}^{22}$, can be shown to be

$$\mathbf{D}_{k,2}^{22} = \text{E}\{-\Delta_{\boldsymbol{\Theta}_k}^{\boldsymbol{\Theta}_k} \log p(\mathbf{y}_k | \boldsymbol{\Theta}_k), \tag{3.29}$$

$$= \frac{1}{\sigma^2} \text{Re}\{\text{E}[\Delta_{\boldsymbol{\Theta}_k} \mathbf{h}(\boldsymbol{\Theta}_k)^* (\Delta_{\boldsymbol{\Theta}_k} \mathbf{h}(\boldsymbol{\Theta}_k)')']\}. \tag{3.30}$$

We expand the right hand side of (3.30) using equations (3.8)-(3.10):

$$\mathbf{D}_{k,2}^{22} = \frac{1}{\sigma^2} \sum_{u=0}^{U-1} \text{Re}\{\text{E}[\Delta_{\boldsymbol{\Theta}_k} \mathbf{h}(\boldsymbol{\Theta}_k; uT_2)^* (\Delta_{\boldsymbol{\Theta}_k} \mathbf{h}(\boldsymbol{\Theta}_k; uT_2)')']\}, \tag{3.31}$$

$$= \frac{1}{\sigma^2} \sum_{u=0}^{U-1} \sum_{n=1}^{N} \sum_{m=1}^{M} \text{Re}\{\text{E}[\Delta_{\boldsymbol{\Theta}_k} \boldsymbol{\mu}_{k,n,m}(\boldsymbol{\Theta}_k; uT_2)^* (\Delta_{\boldsymbol{\Theta}_k} \boldsymbol{\mu}_{k,n,m}(\boldsymbol{\Theta}_k; uT_2)')']\}, \tag{3.32}$$

$$= \frac{1}{\sigma^2} \sum_{u=0}^{U-1} \sum_{n_1=1}^{N} \sum_{n_2=1}^{N} \sum_{m=1}^{M} \sum_{p_1=1}^{P_{n_1,m}} \sum_{p_2=1}^{P_{n_2,m}} \text{Re}\Bigg(\text{E}\Bigg[\Delta_{\boldsymbol{\Theta}_k} \{\mathbf{g}_{n_1,m}^{p_1}(\mathbf{x}_k, \mathbf{z}_k; uT_2) \exp(j\psi_{n_1,m}^{p_1})\}^* \Bigg(\Delta_{\boldsymbol{\Theta}_k} \{\mathbf{g}_{n_2,m}^{p_2}(\mathbf{x}_k, \mathbf{z}_k; uT_2) \exp(j\psi_{n_2,m}^{p_2})\}' \Bigg)' \Bigg] \Bigg). \tag{3.33}$$

Since $\text{E}\{\exp(j\psi_{n_2,m}^{p_2} - j\psi_{n_1,m}^{p_1})\} = \delta_{n_1-n_2}\delta_{p_1-p_2}$, (3.33) reduces to

$$\mathbf{D}_{k,2}^{22} = \frac{1}{\sigma^2} \sum_{u=0}^{U-1} \sum_{n=1}^{N} \sum_{m=1}^{M} \sum_{p=1}^{P_{n,m}} \text{Re}\Bigg(\text{E}\Bigg[\Delta_{\boldsymbol{\Theta}_k} \{\mathbf{g}_{n,m}^{p}(\mathbf{x}_k, \mathbf{z}_k; uT_2) \exp(j\psi_{n,m}^{p})\}^* \Bigg(\Delta_{\boldsymbol{\Theta}_k} \{\mathbf{g}_{n,m}^{p}(\mathbf{x}_k, \mathbf{z}_k; uT_2) \exp(j\psi_{n,m}^{p})\}' \Bigg)' \Bigg] \Bigg). \tag{3.34}$$

With the objective of breaking down (3.34) further, we use the (multivariate) chain rule for differentiation to expand $\Delta_{\boldsymbol{\Theta}}\{\mathbf{g}^p_{n,m}(\mathbf{x},\mathbf{z};t)\exp(j\psi^p_{n,m})\}'$ as

$$\begin{aligned}
&\Delta_{\boldsymbol{\Theta}}\{\mathbf{g}^p_{n,m}(\mathbf{x},\mathbf{z};t)\exp(j\psi^p_{n,m})\}' = \\
&\alpha^p_{n,m}\exp\{j(\psi^p_{n,m}+t\tau^p_{n,m})\}\Bigg(\Bigg[j(\Delta_{\boldsymbol{\theta}}\psi^p_{n,m}+t\Delta_{\boldsymbol{\theta}}\nu^p_{n,m})+\frac{\Delta_{\boldsymbol{\theta}}\alpha^p_{n,m}}{\alpha^p_{n,m}}\Bigg] \\
&\times s(t-\tau^p_{n,m})\mathbf{a}_m(t;\theta^p_{n,m},\dot{\theta}^p_{n,m})' \\
&-\frac{\partial s(t-\tau^p_{n,m})}{\partial\tau^p_{n,m}}\Delta_{\boldsymbol{\theta}}\tau^p_{n,m}\mathbf{a}_m(t;\theta^p_{n,m},\dot{\theta}^p_{n,m})' \\
&+j\bar{d}_m\big[(\sin\theta^p_{n,m}+\dot{\theta}^p_{n,m}t\cos\theta^p_{n,m})\Delta_{\boldsymbol{\theta}}\theta^p_{n,m}+\sin\theta^p_{n,m}t\Delta_{\boldsymbol{\theta}}\dot{\theta}^p_{n,m}\big]s(t-\tau^p_{n,m}) \\
&\times\mathbf{b}_m(t;\theta^p_{n,m},\dot{\theta}^p_{n,m})'\Bigg).
\end{aligned}\tag{3.35}$$

Define $\mathbf{J}(\beta,\gamma)\triangleq\Delta_{\boldsymbol{\theta}}\beta(\Delta_{\boldsymbol{\theta}}\gamma)'$ and $\mathbf{H}(\beta,\gamma)\triangleq\mathbf{J}(\beta,\gamma)+\mathbf{J}(\gamma,\beta)$. We now apply (3.35) in (3.34) to obtain

$$\begin{aligned}
&\mathbf{D}^{22}_{k,2}=\frac{1}{\sigma^2}\sum_{u=0}^{U-1}\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{p=1}^{P_{n,m}}\mathrm{E}\Bigg\{(\alpha^p_{n,m})^2\Bigg(L_m\|s(uT_2-\tau^p_{n,m})\|^2\Big[\mathbf{J}(\psi^p_{n,m},\psi^p_{n,m}) \\
&+(uT_2)^2\mathbf{J}(\nu^p_{n,m},\nu^p_{n,m})+\frac{1}{(\alpha^p_{n,m})^2}\mathbf{J}(\alpha^p_{n,m},\alpha^p_{n,m})\Big] \\
&-L_m\mathrm{Im}\Big\{s(uT_2-\tau^p_{n,m})^*\frac{\partial s(uT_2-\tau^p_{n,m})}{\partial\tau_{n,m}}\Big\} \\
&\times\Big[\mathbf{H}(\psi^p_{n,m},\tau^p_{n,m})+(uT_2)\mathbf{H}(\nu^p_{n,m},\tau^p_{n,m})\Big]-L_m\mathrm{Re}\Big\{s(uT_2-\tau^p_{n,m})^*\frac{\partial s(uT_2-\tau^p_{n,m})}{\partial\tau_{n,m}}\Big\} \\
&\times\Big[\frac{1}{\alpha^p_{n,m}}\mathbf{H}(\alpha^p_{n,m},\tau^p_{n,m})\Big]+L_m\left\|\frac{\partial s(uT_2-\tau^p_{n,m})}{\partial\tau_{n,m}}\right\|^2\mathbf{J}(\tau^p_{n,m},\tau^p_{n,m}) \\
&+\bar{d}_m\frac{L_m(L_m-1)}{2}\|s(uT_2-\tau^p_{n,m})\|^2\Big[\{\sin\theta^p_{n,m}+\dot{\theta}^p_{n,m}(uT_2)\cos\theta^p_{n,m}\} \\
&\times\{\mathbf{H}(\theta^p_{n,m},\psi^p_{n,m})+(uT_2)\mathbf{H}(\nu^p_{n,m},\theta^p_{n,m})\}+(uT_2)\sin\theta^p_{n,m}\{\mathbf{H}(\dot{\theta}^p_{n,m},\psi^p_{n,m}) \\
&+(uT_2)\mathbf{H}(\dot{\theta}^p_{n,m},\nu^p_{n,m})\}\Big]-\bar{d}_m\frac{L_m(L_m-1)}{2}\mathrm{Im}\Big\{s(uT_2-\tau^p_{n,m})^*\frac{\partial s(uT_2-\tau^p_{n,m})}{\partial\tau_{n,m}}\Big\} \\
&\times\Big[\{\sin\theta^p_{n,m}+\dot{\theta}^p_{n,m}(uT_2)\cos\theta^p_{n,m}\}\mathbf{H}(\tau^p_{n,m},\theta^p_{n,m})+\sin\theta^p_{n,m}(uT_2)\mathbf{H}(\tau^p_{n,m},\dot{\theta}^p_{n,m})\Big] \\
&+(\bar{d}_m)^2\frac{(L_m-1)L_m(2L_m-1)}{6}\|s(uT_2-\tau^p_{n,m})\|^2\Big[\{\sin\theta^p_{n,m}+\dot{\theta}^p_{n,m}(uT_2)\cos\theta^p_{n,m}\}^2 \\
&\times\mathbf{J}(\theta^p_{n,m},\theta^p_{n,m})+(\sin\theta^p_{n,m})^2(uT_2)^2\mathbf{J}(\dot{\theta}^p_{n,m},\dot{\theta}^p_{n,m})+\sin\theta^p_{n,m}(uT_2) \\
&\times\{\sin\theta^p_{n,m}+\dot{\theta}^p_{n,m}(uT_2)\cos\theta^p_{n,m}\}\mathbf{H}(\dot{\theta}^p_{n,m},\dot{\theta}^p_{n,m})\Big]\Bigg)\Bigg\}.
\end{aligned}\tag{3.36}$$

The expectation appearing in (3.36) cannot be found in closed form, but we approximate using Monte Carlo. The main challenge of evaluating (3.36) is the calculation of some of the derivatives (appearing within $\mathbf{J}(\cdot)$ and $\mathbf{H}(\cdot)$ terms) with respect to the state variable. Next, we show how this is solved for the functions that are not straightforward.

3.3.2 Evaluating the measurement function gradients

Consider a single path from a transmitter located at $(a_{\mathrm{Tx}}, b_{\mathrm{Tx}})$ to a sensor located at $(a_{\mathrm{Rx}}, b_{\mathrm{Rx}})$. Let the number of reflection points between the transmitter and the target located at (x, y) be S with the s^{th} reflection point denoted by (ω_s, ρ_s). The point corresponding to $s = 1$ is the reflection point immediately after the transmitter. Similarly, let the number of reflection points between the target and the sensor be L with the l^{th} reflection point located at (ξ_l, ζ_l). The reflection point for $l = 1$ corresponds to the reflection immediately before the sensor.

It is convenient to assign:

$$(\omega_0, \rho_0) \triangleq (a_{\mathrm{Tx}}, b_{\mathrm{Tx}}), \tag{3.37}$$

$$(\xi_0, \zeta_0) \triangleq (a_{\mathrm{Rx}}, b_{\mathrm{Rx}}), \tag{3.38}$$

$$(\omega_{S+1}, \rho_{S+1}) \triangleq (x, y), \tag{3.39}$$

$$(\xi_{L+1}, \zeta_{L+1}) \triangleq (x, y). \tag{3.40}$$

Let the bottom most point of the wall on which the l^{th} reflection point lies (in the path from the target to the receiver) be denoted by (γ_l, χ_l). In the case of the wall being parallel to the horizontal axis, choose the left-most point on the wall as (γ_l, χ_l). We use the point (γ_l, χ_l) as a reference point. The distance between the points (ξ_l, ζ_l) and (γ_l, χ_l) is denoted by d_l. We use β_l to denote the angle this wall makes with the horizontal axis. The notation is illustrated in Figure 3.2, which shows a ray hitting the l^{th} reflection point between the target and the sensor.

Similar notation is adopted for the path segments between the transmitter and the target. In particular, we use (ϖ_s, φ_s) to denote the reference point for the wall on which the s^{th} reflection point between the target and the transmitter lies. The distance between (ϖ_s, φ_s) and (ω_s, ρ_s) is denoted by ς_s.

In the following discussion, we have dropped the indices identifying the path in favour of brevity. We now show the procedure only for calculating the derivatives with respect to x and $\dot{x}$ when necessary. The same procedure can be used to find the derivatives with respect to y and $\dot{y}$.

First, we show how $\partial\theta/\partial x$ is calculated. The quantity $\partial\theta/\partial x$ is a term appear-

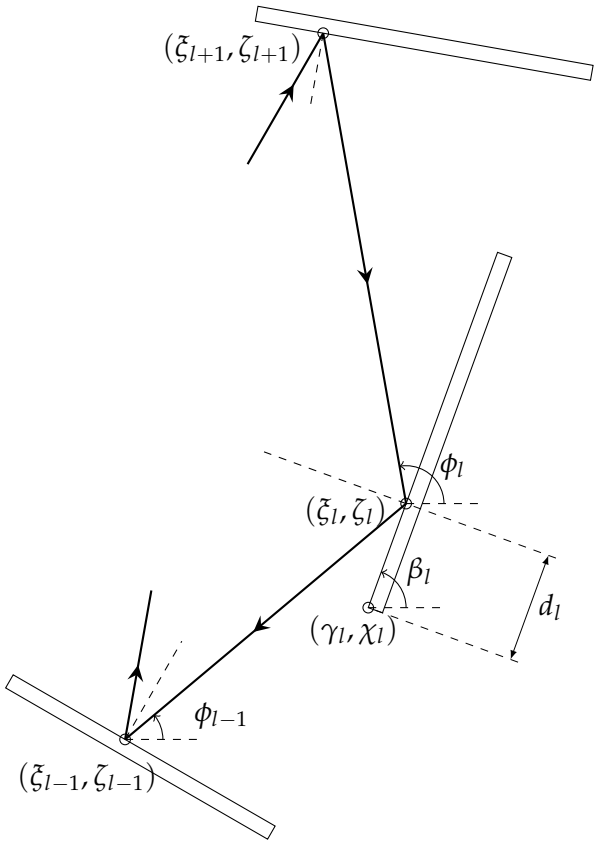


Figure 3.2: Reflection points between the target and sensor.

ing in $\Delta_{\boldsymbol{\Theta}}\theta$ needed to evaluate the PCRB.

The reflection points on the transmitter-target path can be parametrised as follows:

$$\xi_l = \gamma_l + d_l \cos\beta_l, \tag{3.41}$$

$$\zeta_l = \chi_l + d_l \sin\beta_l. \tag{3.42}$$

Let ϕ_l be the AOA at the l^{th} reflecting point. Then it can be shown that, because of the specular nature of the reflections,

$$\phi_l = 2\sum_{i=1}^{l} \beta_i (-1)^{i+l} + \theta(-1)^l. \tag{3.43}$$

Consider the AOA θ at the sensor:

$$\theta =: f_0(d_1) = \arctan\left[\frac{\chi_1 + d_1 \sin\beta_1 - b_{\text{Rx}}}{\gamma_1 + d_1 \cos\beta_1 - a_{\text{Rx}}}\right]. \tag{3.44}$$

Reflection points are related by, for $l = 1, \ldots, L-1$,

$$\tan\phi_{l-1} = \frac{\chi_{l+1} + d_{l+1}\sin\beta_{l+1} - \chi_l - d_l \sin\beta_l}{\gamma_{l+1} + d_{l+1}\cos\beta_{l+1} - \gamma_l - d_l\cos\beta_l}. \tag{3.45}$$

Now by solving (3.45) for d_l we have, for $l = 1, \ldots, L-1$,

$$\begin{aligned} d_l &=: f_l(d_{l+1}, \theta), \\ &= \frac{\chi_{l+1} - \chi_l + d_{l+1}(\sin\beta_{l+1} - \tan\phi_l \cos\beta_{l+1})}{\sin\beta_l - \tan\phi_l\cos\beta_l}, \\ &\quad - \frac{\tan\phi_l(\gamma_{l+1} - \gamma_l)}{\sin\beta_l - \tan\phi_l\cos\beta_l}. \end{aligned} \tag{3.46}$$

Similarly,

$$d_L =: f_L(x, y, \theta) = \frac{y - \chi_L - \tan\phi_L(x - \gamma_L)}{\sin\beta_L - \tan\phi_L\cos\beta_L}. \tag{3.47}$$

Taking derivatives of (3.44),(3.46), and (3.47) with respect to x we have the following equations:

$$\frac{\partial\theta}{\partial x} = \frac{\partial f_0}{\partial d_1}\frac{\partial d_1}{\partial x}, \tag{3.48}$$

$$\frac{\partial d_l}{\partial x} = \frac{\partial f_l}{\partial d_{l+1}}\frac{\partial d_{l+1}}{\partial x} + \frac{\partial f_l}{\partial\theta}\frac{\partial\theta}{\partial x} \text{ for } l = 1, \ldots, L-1, \tag{3.49}$$

$$\frac{\partial d_L}{\partial x} = \frac{\partial f_L}{\partial x} + \frac{\partial f_L}{\partial\theta}\frac{\partial\theta}{\partial x}. \tag{3.50}$$

We rewrite (3.49) as, for $l = 1, \ldots, L-1$,

$$\frac{\partial d_l}{\partial x} = \Lambda_l + \eta_l\frac{\partial\theta}{\partial x}, \tag{3.51}$$

where Λ_l and η_l follow the recursive relationships:

$$\Lambda_l = \Lambda_{l+1}\frac{\partial f_l}{\partial d_{l+1}}, \tag{3.52}$$

$$\eta_l = \eta_{l+1}\frac{\partial f_l}{\partial d_{l+1}} + \frac{\partial f_l}{\partial\theta}, \tag{3.53}$$

with the recursions terminated by using (3.50) to define the quantities:

$$\Lambda_L = \frac{\partial f_L}{\partial x}, \tag{3.54}$$

$$\eta_L = \frac{\partial f_L}{\partial \theta}. \tag{3.55}$$

Finally, by substituting the expression for $\partial d_1/\partial x$ obtained from (3.51) into (3.48), we have

$$\frac{\partial \theta}{\partial x} = \Lambda_0 + \eta_0 \frac{\partial \theta}{\partial x}, \tag{3.56}$$

where

$$\Lambda_0 = \Lambda_1 \frac{\partial f_0}{\partial d_1}, \tag{3.57}$$

$$\eta_0 = \eta_1 \frac{\partial f_0}{\partial d_1}. \tag{3.58}$$

By rearranging (3.56), we obtain

$$\frac{\partial \theta}{\partial x} = \frac{\Lambda_0}{1 - \eta_0}. \tag{3.59}$$

Thus, the derivative $\partial\theta/\partial x$ is found using Λ_0 and η_0, which can be calculated recursively starting from the values of (3.54) and (3.55), and using the recursions (3.52) and (3.53). Once $\partial\theta/\partial x$ is obtained, (3.49) and (3.50) are used to obtain the partial derivatives $\partial d_l/\partial x$, which are needed for some of the calculations that we explain shortly. Figure 3.3 contains an illustration to help understand the sequence of calculations needed.

We now focus on evaluating the derivatives of $\dot{\theta}$, as needed to obtain the PCRB. By differentiating (3.44) with respect to time and using (3.48), we have

$$\dot{\theta} = \frac{\partial f_0}{\partial d_1}\left(\frac{\partial d_1}{\partial x}\dot{x} + \frac{\partial d_1}{\partial y}\dot{y}\right) = \frac{\partial \theta}{\partial x}\dot{x} + \frac{\partial \theta}{\partial y}\dot{y}. \tag{3.60}$$

By once again differentiating the above equation with respect to x, we obtain

$$\frac{\partial \dot{\theta}}{\partial x} = \frac{\partial^2 \theta}{\partial x^2}\dot{x} + \frac{\partial^2 \theta}{\partial x \partial y}\dot{y}. \tag{3.61}$$

The second derivatives appearing in (3.61) can be found following a similar procedure to the one used to obtain the first derivative $\partial\theta/\partial x$.

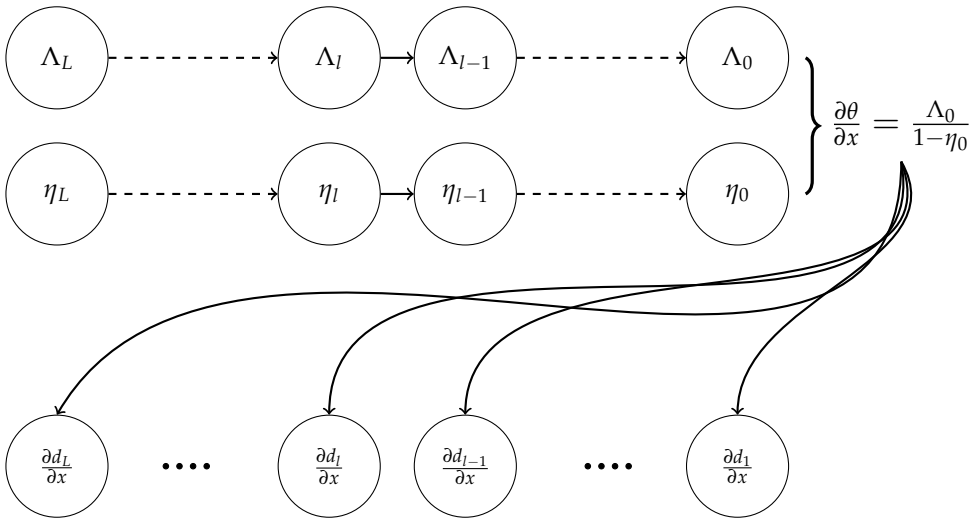


Figure 3.3: An illustration of the process involved in calculating $\partial\theta/\partial x$ and $\partial d_l/\partial x$. The directions of the arrows indicate the sequence of evaluating the quantities.

The derivative of $\dot{\theta}$ with respect to $\dot{x}$ is simply

$$\frac{\partial\dot{\theta}}{\partial\dot{x}} = \frac{\partial\theta}{\partial x}. \tag{3.62}$$

Let the s^{th} and l^{th} path segments of the forward and return path be denoted by R_s and r_l, respectively; that is,

$$R_s = \sqrt{(\omega_s - \omega_{s-1})^2 + (\rho_s - \rho_{s-1})^2}, \tag{3.63}$$

$$r_l = \sqrt{(\xi_l - \xi_{l-1})^2 + (\zeta_l - \zeta_{l-1})^2}. \tag{3.64}$$

Then, the received signal amplitude is given by

$$\alpha = \sqrt{E}\frac{\prod_{s=1}^{S+1}\varrho_s\prod_{l=1}^{L}\epsilon_l}{\sum_{s=1}^{S+1}R_s\sum_{l=1}^{L+1}r_l}, \tag{3.65}$$

where E is the transmitted signal energy and ϱ_s is a random reflectivity factor of the wall which the s^{th} reflection point (between the transmitter and the target) lies, for the path in context. Similarly, ϵ_l is the random reflectivity of the wall on which the l^{th} reflection point lies between the target and the transmitter, for a particular path.

By using (3.63)-(3.65) and noting that R_s and r_l are functions of $(\varsigma_s, \varsigma_{s-1})$ and

(d_l, d_{l-1}) respectively, the derivative of α with respect to x is given by

$$\frac{\partial \alpha}{\partial x} = \sum_{l=1}^{L+1} \frac{\partial \alpha}{\partial r_l}\frac{\partial r_l}{\partial x} + \sum_{s=1}^{S+1} \frac{\partial \alpha}{\partial R_s}\frac{\partial R_s}{\partial x}, \tag{3.66}$$

$$\begin{aligned} &= \left\{ \frac{\partial \alpha}{\partial r_1}\frac{\partial r_1}{\partial d_1}\frac{\partial d_1}{\partial x} + \sum_{l=2}^{L} \frac{\partial \alpha}{\partial r_l}\left(\frac{\partial r_l}{\partial d_l}\frac{\partial d_l}{\partial x} + \frac{\partial r_l}{\partial d_{l-1}}\frac{\partial d_{l-1}}{\partial x} \right) + \frac{\partial \alpha}{\partial r_{L+1}}\frac{\partial r_{L+1}}{\partial d_L}\frac{\partial d_L}{\partial x} \right. \\ &\quad \left. + \frac{\partial \alpha}{\partial r_{L+1}}\frac{\partial r_{L+1}}{\partial x} \right\} \\ &\quad + \left\{ \frac{\partial \alpha}{\partial R_1}\frac{\partial R_1}{\partial \varsigma_1}\frac{\partial \varsigma_1}{\partial x} + \sum_{s=2}^{S} \frac{\partial \alpha}{\partial R_s}\left(\frac{\partial R_s}{\partial \varsigma_s}\frac{\partial \varsigma_s}{\partial x} + \frac{\partial R_s}{\partial \varsigma_{s-1}}\frac{\partial \varsigma_{s-1}}{\partial x} \right) + \frac{\partial \alpha}{\partial R_{S+1}}\frac{\partial R_{S+1}}{\partial \varsigma_S}\frac{\partial \varsigma_S}{\partial x} \right. \\ &\quad \left. + \frac{\partial \alpha}{\partial R_{S+1}}\frac{\partial R_{S+1}}{\partial x} \right\}. \end{aligned} \tag{3.67}$$

We have already shown how $\partial d_l/\partial x$ appearing in (3.67) is calculated. The procedure for calculating $\partial \varsigma_s/\partial x$ terms is exactly the same as that for calculating $\partial d_l/\partial x$ (note that d_l is a quantity appearing in the context of the forward signal path from the transmitter to the target; the counterpart in the reverse segment is ς_s. Thus the same procedure used to obtain $\partial d_l/\partial x$ can be used to evaluate $\partial \varsigma_s/\partial x$). The partial derivatives $\partial r_l/\partial d_l, \partial r_l/\partial d_{l-1}$, and $\partial r_{L+1}/\partial x$ appearing in (3.67) are:

$$\frac{\partial r_1}{\partial d_1} = \frac{(\cos\beta_1)(\gamma_1 + d_1\cos\beta_1 - a_{\mathrm{Tx}}) + (\sin\beta_1)(\chi_1 + d_1\sin\beta_1 - b_{\mathrm{Tx}})}{r_1}, \tag{3.68}$$

$$\begin{aligned} \frac{\partial r_l}{\partial d_l} &= \frac{(\cos\beta_l)\left(d_l\cos\beta_l - d_{l-1}\cos\beta_{l-1} + \gamma_l - \gamma_{l-1}\right)}{r_l} \\ &\quad + \frac{(\sin\beta_l)\left(d_l\sin\beta_l - d_{l-1}\sin\beta_{l-1} + \chi_l - \chi_{l-1}\right)}{r_l} \text{ for } l = 2, 3, \ldots, L, \end{aligned} \tag{3.69}$$

$$\begin{aligned} \frac{\partial r_l}{\partial d_{l-1}} &= \frac{(\cos\beta_{l-1})\left(d_{l-1}\cos\beta_{l-1} - d_l\cos\beta_l + \gamma_{l-1} - \gamma_l\right)}{r_l} \\ &\quad + \frac{(\sin\beta_{l-1})\left(d_{l-1}\sin\beta_{l-1} - d_l\sin\beta_l + \chi_{l-1} - \chi_l\right)}{r_l} \text{ for } l = 2, 3, \ldots, L, \end{aligned} \tag{3.70}$$

$$\frac{\partial r_{L+1}}{\partial d_L} = \frac{(\cos\beta_L)\left(d_L\cos\beta_L + \gamma_L - x\right) + (\sin\beta_L)\left(d_L\sin\beta_L + \chi_L - y\right)}{r_{L+1}}, \tag{3.71}$$

$$\frac{\partial r_{L+1}}{\partial x} = \frac{x - d_L\cos\beta_L - \gamma_L}{r_{L+1}}. \tag{3.72}$$

The derivatives of R_s for $s = 1, 2, \ldots, S+1$ follow similarly and these can then be used in (3.67).

The delay τ is given by

$$\tau = 1/c \sum_{s=1}^{S+1} R_s + 1/c \sum_{r=1}^{L+1} r_l, \tag{3.73}$$

Thus, the derivative of τ with respect to x can be obtained by

$$\frac{\partial \tau}{\partial x} = 1/c \sum_{s=1}^{S+1} \frac{\partial R_s}{\partial \varsigma_s} \frac{\partial \varsigma_s}{\partial x} + 1/c \sum_{r=1}^{L+1} \frac{\partial r_l}{\partial d_l} \frac{\partial d_l}{\partial x}. \tag{3.74}$$

where the quantities on the right hand side of (3.74) are already discussed.

The doppler shift is given by $\nu = 2\pi(V + U)/\lambda$, where

$$\begin{aligned} U &= \dot{R}_{S+1}, \\ &= \frac{\partial R_{S+1}}{\partial x}\dot{x} + \frac{\partial R_{S+1}}{\partial y}\dot{y} + \frac{\partial R_{S+1}}{\partial \varsigma_S}\left\{\frac{\partial \varsigma_S}{\partial x}\dot{x} + \frac{\partial \varsigma_S}{\partial y}\dot{y}\right\}, \end{aligned} \tag{3.75}$$

$$\begin{aligned} V &= \dot{r}_{L+1}, \\ &= \frac{\partial r_{L+1}}{\partial x}\dot{x} + \frac{\partial r_{L+1}}{\partial y}\dot{y} + \frac{\partial r_{L+1}}{\partial d_L}\left\{\frac{\partial d_L}{\partial x}\dot{x} + \frac{\partial d_L}{\partial y}\dot{y}\right\}. \end{aligned} \tag{3.76}$$

To evaluate ν with respect to x, the second derivatives of the d_l and ς_s are required. Those can be found by following exactly the same recursive procedure used to find their first derivatives.

We have omitted other derivatives that are needed to evaluate PCRB since they are straightforward.

3.4 Results and discussion

Consider a transmitted signal of unit energy given by

$$s(t) = \frac{1}{\sqrt{P}} \sum_{-(P-1)/2}^{(P-1)/2} \frac{\exp\left[jv - 1/(2\kappa^2)\right](t - pT_1)^2}{(\pi\kappa^2)^{1/4}}. \tag{3.77}$$

In our example, we have set the number P of transmitted pulses to three. The width parameter κ is chosen to give an effective duration of 250 ns and the chirp rate v is such that the effective bandwidth of the signal is 40 MHz. The pulse repetition interval is chosen as $T_1 = 100$ μs. We use a single transmitter and a single sensor with $L = 3$ array elements. Antenna elements are separated by 4λ; the wavelength of the carrier frequency λ is chosen as 0.1 m. The state sampling period is $\tilde{T} = 1$ s and the receiver samples data with a period of $T_2 = 10$ ns.

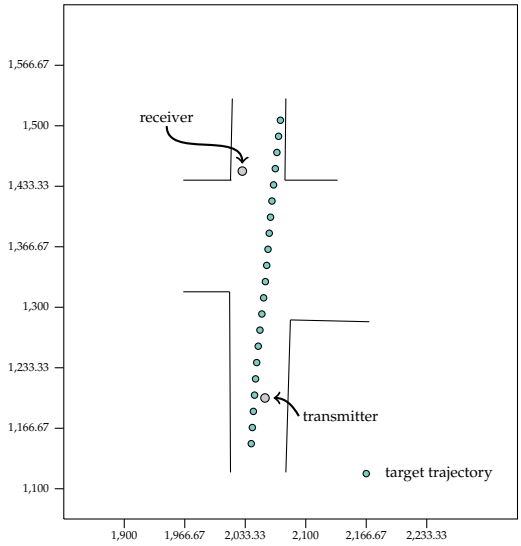


Figure 3.4: The multipath environment and the nominal target trajectory.

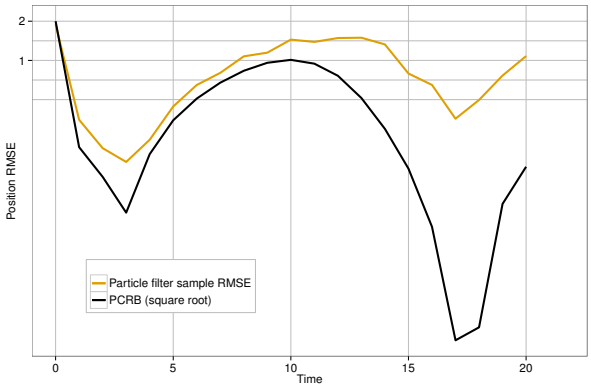
The radar environment and the nominal target trajectory are shown in Figure 3.4. The initial prior distribution for the target kinematic state is Gaussian with mean and covariance matrix given by

$$\gamma_0 = \begin{bmatrix} 2040 \\ 1.55 \\ 1150 \\ 17.5 \end{bmatrix}, \tag{3.78}$$

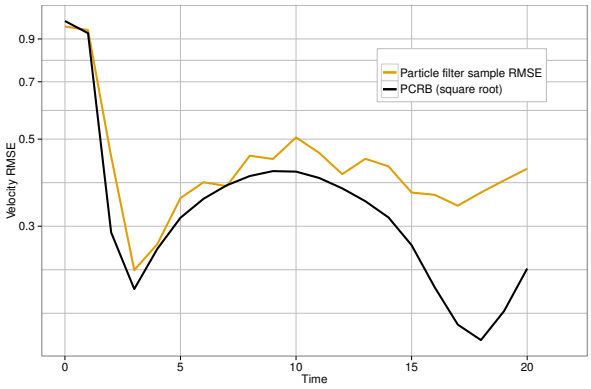
$$\mathbf{P}_0 = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix}. \tag{3.79}$$

The process noise covariance matrix $\mathbf{Q}_k$ is set by $\kappa = 0.04$. The reflecting surfaces are made up of eight walls. All the random reflectivity variables associated with walls are assumed to be distributed according to $\mathcal{N}(\cdot; 0.6, 0.05^2)$; that is, $(\mu_b, \iota_b) \triangleq (0.6, 0.05^2)$, for $b = 1, 2, \ldots, 8$. The random reflectivity variables associated with the target are assumed to be distributed according to $\mathcal{N}(\cdot; 0.7, 0.05^2)$; that is, $(\mu_0, \iota_0) \triangleq (0.7, 0.05^2)$. The transmitted signal energy E is chosen such that the signal-to-noise ratio of the weakest multipath signals at the output of the matched filter, averaged over the entire trajectory, is 20dB.

Figure 3.5 shows the PCRB for position and velocity for the setup considered. The $D_{k,2}^{22}$ component of the bound (3.36), which is the only component without a



(a)



(b)

Figure 3.5: Root mean square error of a particle filter against PCRB for: (a) position in meters (b) velocity in meters per second.

closed form solution, was found using 500 Monte Carlo realisations. The RMSE of a particle filter based on Markov Chain Monte Carlo is overlaid on top of the PCRB curves. The design of this particle filter is covered in Chapter 5. The particle filter was run over 100 realisations to obtain the RMSE. The Markov Chain length is set to 150 for this particle filter. Note that the RMSE curves follow the general shape of the PCRB curves, and in some instances become very close to the lower bound. However, during the latter stage of the trajectory, the RMSE deviates significantly from the PCRB curves. It should be noted that the RMSE being lower at time 0 in the velocity PCRB plot is a consequence of Monte Carlo sampling error.

Note that the overall shape of the PCRB conforms with the general intuition; that is, the PCRB is relatively low while the target is approaching either the transmitter or the sensor because of strong multipath reflections. The PCRB reaches a local peak halfway through the trajectory (around the time $k = 10$), which again is

due to relatively weak multipath signals available around that time of the trajectory.

Next, consider a scenario involving a single wall as shown in Figure 3.6.

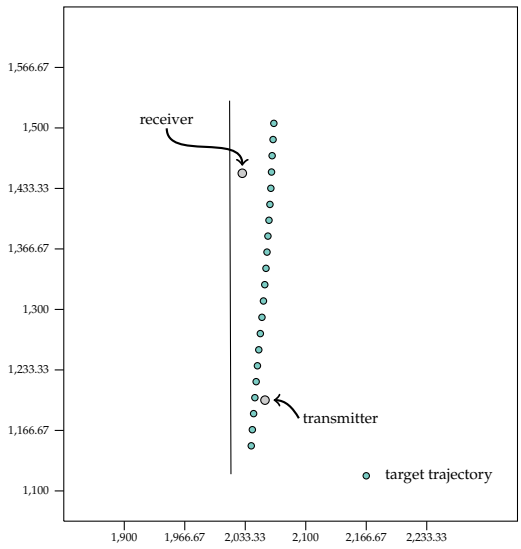
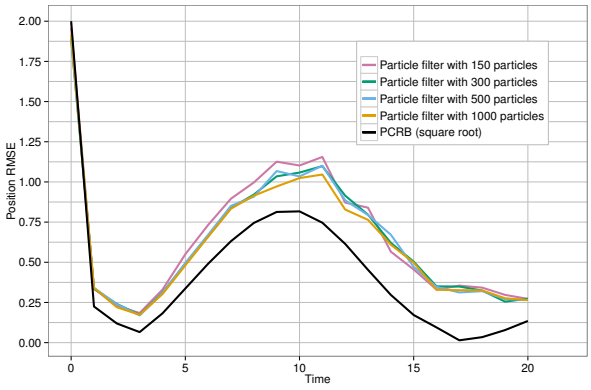


Figure 3.6: The multipath environment and the nominal target trajectory involving a single wall.

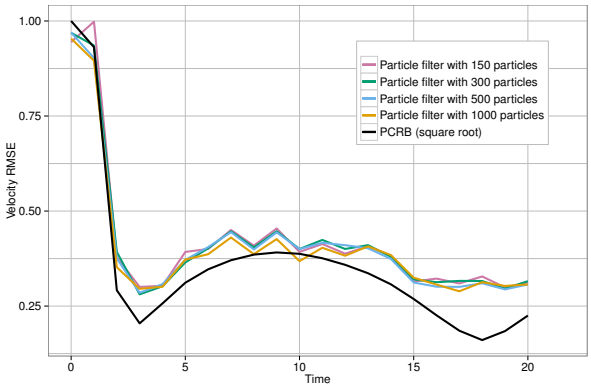
We ran the the particle filter developed in Chapter 5 with varying number of particles. The results are shown in Figure 3.7. As expected, increasing the number of particles helps to reduce the RMSE. However, since the PCRB is not a tight bound for this problem, it is not expected to be reached by asymptotically increasing the number of particles (note that particularly in non-linear signal processing problems, the PCRB not being possible to achieve is the norm rather than the exception).

3.5 Conclusion

In this chapter we have formulated a general tracking problem in a multipath environment. We have incorporated into our model some of the uncertainty prevailing in a realistic tracking situation, not usually taken into account in conventional multipath models. Obtaining performance bounds is an important part of any estimation problem, and this chapter has been dedicated to deriving the Posterior Cramér Rao Lower Bound for the novel model. The obtained bound can be used as a benchmark in designing statistical filters. The main challenge of deriving the bound resides in the evaluation of various derivatives with respect to the target kinematics. We have solved this problem by exploiting geometrical relation-



(a)



(b)

Figure 3.7: Root mean square error of particle filters against PCRB for: (a) position in meters (b) velocity in meters per second.

ships induced in the environment through the specular nature of reflections. The recursive method used to calculate the various derivatives can be easily used or extended to find the PCRB for other multipath models.

CHAPTER 4

Conjugate analysis of a multivariate von-Mises distribution

Summary

In this chapter, we propose a Bayesian conjugate framework for inferring multiple phases. The framework requires a generalisation of the von-Mises (VM) distribution for multiple variables. This work was performed in our pursuit of designing a target tracking filter for the multipath model introduced in Chapter 3. The principal difficulty we faced with generalising the VM distribution is the computation of the first order moment and the normalising constant, which are important for Bayesian inference. We propose two approaches to solve the problem: one based on a Bessel function expansion and the other based on a Markov Chain Monte Carlo (MCMC) technique using the Gibbs sampler. We then assess the performance of these two methods against variations in parameters of the Generalised von-Mises (GVM) distribution.

4.1 Introduction

In the last chapter we introduced a novel model for a multipath environment, where the uncertainty in the wall locations is accounted for by introducing a random phase shift to the radar measurement equation. Since the measurement recorded at the sensor is a superposition of such randomly phase shifted radar signals, the likelihood function depends on multiple phase variables. Our initial attempts at designing a tracker for the multipath model directed us to derive a Bayesian conjugate framework to conveniently estimate the phase variables. This chapter presents the work done towards that end.

In particular, we consider the problem of estimating the phases of multiple superimposed signals embedded in additive Gaussian noise. This problem is of

interest in a number of areas such as radar, communications, and biology. For instance, in communications we may receive a signal, which is the sum of several carriers, each subject to an unknown phase shift. For the case of a single phase, it was shown in [Quinn 2011] that a Bayesian conjugate analysis is possible with the VM distribution as the prior. In other words, when a VM distributed prior is updated with a likelihood function proportional to a VM distribution, then the resulting posterior also follows a VM distribution. It will be shown in this chapter that, for multiple phases, the conjugate prior is a particular multivariate generalisation of the VM distribution. Note that we are considering "phase" variables, which are by definition circular quantities. Some fundamental differences between circular and linear statistics are given in Appendix 4.A.

Several multivariate generalisations of the VM distribution have been proposed in the statistics literature. The key difficulty with the use of multivariate VM distributions is the computation of the integrals such as the normalising constant required to compute posterior statistics. In [Rivest, L. P. 1988], [Singh 2002], and [Mardia 2010] the normalising constant for various bivariate VM distributions were derived. The results derived in [Rivest, L. P. 1988] and [Mardia 2010] are of particular interest as they can be used in the conjugate analysis of two phases. However, there appears to be no results that can be used for conjugate Bayesian analysis of a general number of phases. We present two methods of calculating the posterior statistics for a proposed GVM distribution. The first method is based on the direct evaluation of the integral using summations while the second method is based on MCMC methods [Robert 2004]. The scheme proposed in [Chib 1995] is used to calculate the normalising constant using MCMC, which uses multiple Gibbs chains to approximate the normalising constant. The samples generated in the process of calculating the normalising constant can be conveniently used to calculate posterior statistics such as circular mean and variance. It will be shown later that the direct method requires an infinite summation that does not have a closed form solution. In practice, we have to choose a finite truncation point for the infinite summation and discard the insignificant terms. Thus, the two methods presented in this chapter are not exact methods but approximations. In this chapter, we assess the performance of these two approximations as a function of the parameters of the proposed GVM distribution.

The rest of the chapter is organised as follows. First, we show the relationship of the GVM distribution with the multipath model explained in Chapter 3 in Section 4.2. In Section 4.3, we introduce the general measurement model. The core of the technical content is embedded in Section 4.4. There, we present the conjugate analysis for the GVM distribution and present two methods for calculating

the moments. Various simulation results used to assess the methods are presented in Section 4.5. Finally, the conclusion for the work in this chapter is presented in Section 4.6.

Some of the notations and acronyms we have adapted are given in Table 4.1. Any notation/acronym, which does not appear in Table 4.1 is defined when it first appears.

Table 4.1: Summary of common notations and acronyms.

Notation/Acronym	Description
$\mathbb{C}$	Field of complex numbers
$\mathbb{R}$	Field of real numbers
$\mathcal{VM}(\psi;\gamma,\delta)$	von-Mises distribution with concentration parameter γ and location parameter δ
$\mathcal{GVM}(\boldsymbol{\psi};\boldsymbol{\mu}_1)$	Generalised von-Mises distribution with parameters defined by $\boldsymbol{\mu}_1$
$\mathcal{N}(\cdot;\boldsymbol{\Delta},\boldsymbol{\Lambda})$	Multivariate Gaussian distribution with parameters $\boldsymbol{\Delta}$ (mean) and $\boldsymbol{\Lambda}$ (covariance)
$\mathcal{CN}(\cdot;\boldsymbol{\Delta},\boldsymbol{\Lambda})$	Circular symmetric complex Gaussian distribution with parameters $\boldsymbol{\Delta}$ (mean) and $\boldsymbol{\Lambda}$ (covariance)
$\mathcal{R}(\cdot)$	Real component of the argument
GVM	Generalised von-Mises distribution
VM	von-Mises distribution
MCMC	Markov Chain Monte Carlo
MSE	Mean Squared Error

4.2 The relationship of the GVM distribution to the multipath model

For brevity, we present a simplified version of the multipath model presented in Chapter 3. Consider the target tracking problem in an urban terrain where the received signal at the sensors consists of multiple reflecting paths. Assume that a map of the terrain is available. In a realistic setting, the information obtained about the building locations using the map may not be very accurate. As discussed in Chapter 3 we account for this uncertainty in reflector locations by introducing a random phase shift for each multipath in the measurement equation (4.1); that is,

$$\mathbf{y} = \sum_{i=1}^{q} \exp(j\psi_i)\mathbf{g}_i(\mathbf{x}) + \mathbf{w}, \tag{4.1}$$

where

$\mathbf{x}$ = Target state,
$\mathbf{y}$ = Measurement vector,
q = Number of multipaths,
$\mathbf{g}_i(\mathbf{x})$ = Measurement function for the i^{th} multipath. This function accounts for quantities such as delay, doppler and attenuation of the transmitted signal,
$e^{j\psi_i}$ = Random phase shift for the i^{th} path.

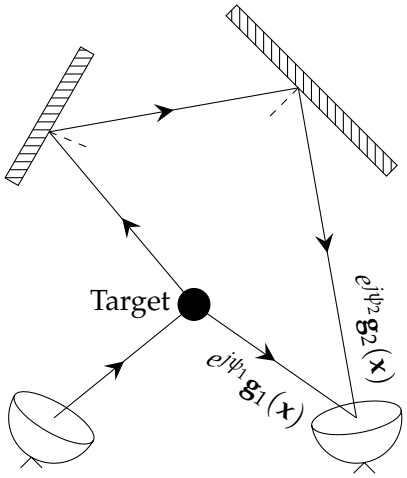


Figure 4.1: An illustration of circular variables in the multipath model.

A hypothetical radar scene involving two multipaths is shown in Figure 4.1.

We are interested in obtaining an estimate for the target state upon observing the measurements. Suppose we design a marginalised particle filter [Arulampalam 2002] to approximate $p(\mathbf{x}, \boldsymbol{\psi}|\mathbf{y})$ as follows:

$$
\begin{aligned}
p(\mathbf{x}, \boldsymbol{\psi}|\mathbf{y}) &= p(\boldsymbol{\psi}|\mathbf{x}, \mathbf{y})p(\mathbf{x}|\mathbf{y}) && (4.2)\\
&\propto p(\boldsymbol{\psi}|\mathbf{x}, \mathbf{y})p(\mathbf{y}|\mathbf{x})p(\mathbf{x}), && (4.3)\\
&= p(\boldsymbol{\psi}|\mathbf{x}, \mathbf{y}) \left[\int p(\mathbf{y}|\mathbf{x}, \boldsymbol{\psi})p(\boldsymbol{\psi}|\mathbf{x})d\boldsymbol{\psi} \right] p(\mathbf{x}), && (4.4)\\
&\approx \sum_{i=1}^{N} v_i p(\boldsymbol{\psi}|\mathbf{x}^{(i)}, \mathbf{y})\delta(\mathbf{x} - \mathbf{x}^{(i)}), && (4.5)
\end{aligned}
$$

where for $i = 1, 2, \ldots, N$,

$$x^{(i)} \sim q(\mathbf{x}|\mathbf{y}) \text{ for some importance distribution } q(\mathbf{x}|\mathbf{y}) \tag{4.6}$$

$$v^i \propto \frac{p(\mathbf{x}^{(i)}) \int p(\mathbf{y}|\mathbf{x}^{(i)}, \boldsymbol{\psi}) p(\boldsymbol{\psi}|\mathbf{x}^{(i)}) d\boldsymbol{\psi}}{q(\mathbf{x}^{(i)}|\mathbf{y})}. \tag{4.7}$$

Note that the marginalisation is over the phase variables $\boldsymbol{\psi}$ and only $\mathbf{x}$ samples are drawn to obtain the posterior approximation (4.5). From (4.5) point estimates for $\mathbf{x}$ could be obtained by using a weighted average of the samples. Point estimates for $\boldsymbol{\psi}$ could be obtained by taking the weighted average of the means of the conditional posterior $p(\boldsymbol{\psi}|\mathbf{x}, \mathbf{y})$. Now focus on the term $\int p(\mathbf{y}|\mathbf{x}, \boldsymbol{\psi}) p(\boldsymbol{\psi}) d\boldsymbol{\psi}$ appearing in the weight calculation (4.7). This is exactly the normalising constant of $p(\boldsymbol{\psi}|\mathbf{y}, \mathbf{x})$.

Thus the distribution $p(\boldsymbol{\psi}|\mathbf{y}, \mathbf{x})$ is being used for calculating the particle weights v_i as well as for obtaining a point estimate for $\boldsymbol{\psi}$. It turns out that the GVM distribution studied in this chapter has the same functional form of $p(\boldsymbol{\psi}|\mathbf{x}, \mathbf{y})$; hence our motivation to invent a procedure to calculate the normalising constant and a conjugate framework for GVM distribution.

4.3 Modelling and notation

We drop the dependency on target state $\mathbf{x}$ from the example used in Section 4.2 and re-pose the problem so that the setup is not specific to the multipath filtering problem introduced in Chapter 3.

We observe q superimposed signals in additive noise to obtain an observation vector $\mathbf{y}$ consisting of k complex-valued measurements. In particular, we consider a measurement equation of the form

$$\mathbf{y} = \sum_{i=1}^{q} \exp(j\psi_i)\mathbf{g}_i + \mathbf{w}, \tag{4.8}$$

where $\mathbf{g}_i \in \mathbb{C}^k$, $i = 1, \ldots, q$ are known signal vectors and the noise vector $\mathbf{w}$ is a circular symmetric complex multivariate Gaussian with covariance matrix $\sigma^2\mathbf{I}_k$, where $\mathbf{I}_k$ is the $k \times k$ identity matrix. It is desired to estimate the unknown phase shifts: $\psi_i \in [0, 2\pi)$, $i = 1, \ldots, q$. Equation (4.8) can be considered as a multivariate circular regression model. In addition to the multipath filtering problem, real world examples of this model can be found in communications, radar, bioinformatics, microbiology and molecular physics.

4.4 Theory/Methodology

4.4.1 Conjugate analysis

A conjugate prior has the desirable property that the posterior distribution retains the same form as the prior, which facilitates convenient recursive Bayesian inference schemes. Thus, we wish to find a conjugate prior for our problem of multiple phase estimation.

The measurement equation (4.8) results in a likelihood of the form:

$$\mathcal{L}(\boldsymbol{\psi};\mathbf{y}) = \mathcal{CN}\left(\mathbf{y};\sum_{i=1}^{q}\exp(j\psi_i)\mathbf{g}_i;\sigma^2\mathbf{I}_k\right), \tag{4.9}$$

$$= \frac{b(\mathbf{y})}{\pi^k\left|\sigma^2\mathbf{I}_k\right|}\exp\left[\mathcal{R}\{\mathbf{h}(\mathbf{y})'\mathbf{u}(\boldsymbol{\psi})\}\right], \tag{4.10}$$

where

$$\mathbf{h}(\mathbf{y}) = \frac{2}{\sigma^2}[\mathbf{y}^*\mathbf{g_1} \quad \dots \quad \mathbf{y}^*\mathbf{g_q} \quad \dots -\mathbf{g_2}^*\mathbf{g_1} \quad -\mathbf{g_3}^*\mathbf{g_1}\dots \quad -\mathbf{g_q}^*\mathbf{g_{q-1}}]',$$

$$b(\mathbf{y}) = \exp\left[-\frac{1}{\sigma^2}(||\mathbf{y}||_2^2 + \sum_{i=1}^{q}||\mathbf{g}_i||_2^2)\right],$$

$$\mathbf{u}(\boldsymbol{\psi}) = [\mathbf{a}(\boldsymbol{\psi})' \quad \tilde{\mathbf{a}}(\boldsymbol{\psi})']',$$

and

$$\mathbf{a}(\boldsymbol{\psi}) = [e^{j\psi_1} \quad e^{j\psi_2} \quad \dots \quad e^{j\psi_q}]',$$

$$\tilde{\mathbf{a}}(\boldsymbol{\psi}) = [e^{j(\psi_1-\psi_2)} \quad e^{j(\psi_1-\psi_3)} \quad \dots \quad e^{j(\psi_{q-1}-\psi_q)}]'.$$

The conjugate prior for the likelihood (4.10) is given by

$$f(\boldsymbol{\psi}|\boldsymbol{\mu}_0) = \frac{1}{N_0(\boldsymbol{\mu}_0)}\exp\left[\mathcal{R}\{\boldsymbol{\mu}_0'\mathbf{u}(\boldsymbol{\psi})\}\right], \tag{4.11}$$

where $\boldsymbol{\mu}_0 \in \mathbb{C}^{q(q+1)/2}$ is the parameter vector of the prior and $N_0(\boldsymbol{\mu}_0)$ is the normalising constant.

To see that (4.11) is indeed the conjugate prior, we use Bayes' rule to find the posterior density by:

$$f(\boldsymbol{\psi}|\mathbf{y},\boldsymbol{\mu}_0) \propto \mathcal{L}(\boldsymbol{\psi};\mathbf{y})f(\boldsymbol{\psi}|\boldsymbol{\mu}_0),$$

$$
\begin{aligned}
&= \mathcal{CN}(\mathbf{y}; \sum_{i=1}^{q} e^{j\psi_i}\mathbf{g}_i; \sigma^2\mathbf{I}_k)\mathcal{GVM}(\boldsymbol{\psi}; \boldsymbol{\mu}_0), \\
&\propto \exp\left[\mathcal{R}\{\mathbf{h}(\mathbf{y})'\mathbf{u}(\boldsymbol{\psi})\}\right] \exp\left[\mathcal{R}\{\boldsymbol{\mu}_0'\mathbf{u}(\boldsymbol{\psi})\}\right], \\
&= \exp\left[\mathcal{R}\{(\mathbf{h}(\mathbf{y}) + \boldsymbol{\mu}_0)'\mathbf{u}(\boldsymbol{\psi})\}\right], \\
&\propto \mathcal{GVM}(\boldsymbol{\psi}; \boldsymbol{\mu}_1),
\end{aligned} \tag{4.12}
$$

where we use the notation $\mathcal{GVM}(\boldsymbol{\psi}; \boldsymbol{\mu}_0)$ to refer to the distribution (4.11) and

$$
\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + \mathbf{h}(\mathbf{y}). \tag{4.13}
$$

Thus, a posterior of the same form as the prior is obtained with parameters updated using the simple rule (4.13), hence the conjugacy.

Note that the distribution (4.11) is a multivariate generalisation of the VM distribution. This can be easily seen by re-writing (4.11) in a re-parameterised form as follows:

$$
\begin{aligned}
f(\boldsymbol{\psi}|\boldsymbol{\mu}_0) &=: \tilde{f}(\boldsymbol{\psi}|\boldsymbol{\nu}_0), \\
&= \frac{1}{\tilde{N}_0(\boldsymbol{\nu}_0)} \exp\left[\sum_{i=0}^{q} \kappa_{i,i}\cos(\psi_i - \theta_{i,i}) + \sum_{i=1}^{q-1}\sum_{j=i+1}^{q} \kappa_{i,j}\cos(\psi_i - \psi_j + \theta_{i,j})\right],
\end{aligned} \tag{4.14}
$$

where $\boldsymbol{\nu}_0$ is the vector of parameters $\kappa_{i,j}$ and $\theta_{i,j}$ for $1 \leq i \leq q$ and $i \leq j \leq q$. Transformation of the parameters from $\boldsymbol{\mu}_0$ to $\boldsymbol{\nu}_0$ and vice versa are straightforward.

Note that the univariate VM distribution is a special case of the GVM distribution for $q = 1$. Non-zero $\kappa_{i,j}$ terms for $i \neq j$ give rise to correlation between the phase variables. Thus $\kappa_{i,j} = 0$ for all $i \neq j$ corresponds to phases being independent of each other and hence for this situation GVM becomes the product of q VM distributions with circular mean for i^{th} phase being equal to $\theta_{i,i}$. When correlation between the phases is present the circular mean for i^{th} phase does not equal $\theta_{i,i}$, but rather is influenced by other phases.

In point estimation, we are very much interested in two particular posterior statistics, namely the first order moment and the variance. In distributions involving circular variables, it is customary to take the expectation of $\exp(j\psi_k)$ instead of ψ_k as the first moment, as explained in Appendix 4.A.

The first order moment Λ of the distribution $\mathcal{GVM}(\boldsymbol{\psi}; \boldsymbol{\mu}_1)$ is given by

$$
\Lambda = E\{[\exp(j\psi_1) \quad \ldots \quad \exp(j\psi_q)]\}'. \tag{4.15}
$$

Point estimates of the phases can be found as

$$\hat{\boldsymbol{\psi}} = \angle\Lambda. \tag{4.16}$$

Evaluation of the posterior statistic in (4.15) requires integration with respect to $\mathcal{GVM}(\boldsymbol{\psi};\boldsymbol{\mu}_1)$. This is not straightforward for an arbitrary number of q phases. The $q = 1$ case is trivial since this leads to the well known VM distribution. Evaluation of similar integrals for $q = 2$ can be found in [Singh 2002] and [Mardia 2010]. The integrals for $q > 2$ have not been addressed in the literature but are essential for Bayesian inference for multiple phases using our model.

Next, we present two methods to evaluate the posterior statistics.

4.4.2 Evaluating Posterior statistics using the direct method

Consider the problem of finding the normalising constant and the first order moment of $\mathcal{GVM}(\boldsymbol{\psi};\boldsymbol{\mu})$. It is easier to solve this by using the re-parameterised form (4.14). Let $\boldsymbol{\nu}$ be the new parameter obtained by transforming $\boldsymbol{\mu}$, and define $\tilde{N}_{i,k}(\boldsymbol{\nu})$ with $1 \leq i \leq q$ and $k \in \{0,1\}$ for the parameter vector $\boldsymbol{\nu}$ as

$$\begin{aligned}\tilde{N}_{i,k}(\boldsymbol{\nu}) = \int_{\boldsymbol{\psi}\in[0,2\pi]^q} \exp(j\psi_i k) \exp\Bigg[\sum_{j=0}^{q} \kappa_{j,j}\cos(\psi_j - \theta_{j,j}) \\ + \sum_{t=1}^{q-1}\sum_{j=t+1}^{q} \kappa_{t,j}\cos(\psi_t - \psi_j + \theta_{t,j})\Bigg] \mathrm{d}\boldsymbol{\psi},\end{aligned} \tag{4.17}$$

Note that $\tilde{N}_{i,0}(\boldsymbol{\nu})$ gives the normalising constant of $\mathcal{GVM}(\boldsymbol{\psi};\boldsymbol{\mu})$ and is independent of the index i. Thus we will simply refer to $\tilde{N}_{i,0}(\boldsymbol{\nu})$ as $\tilde{N}_0(\boldsymbol{\nu})$.

Now the first order moment vector (4.15) of $\mathcal{GVM}(\boldsymbol{\psi};\boldsymbol{\mu})$ is expressed as

$$\Lambda = \frac{1}{\tilde{N}_0(\boldsymbol{\nu})}[\tilde{N}_{1,1}(\boldsymbol{\nu}) \quad \tilde{N}_{2,1}(\boldsymbol{\nu}) \quad \dots \quad \tilde{N}_{q,1}(\boldsymbol{\nu})]'. \tag{4.18}$$

The key formula for evaluating integral (4.17) is the following result from [Abramowitz 1964, 9.6.34]:

$$\exp[z\cos(\beta)] = \sum_{k=-\infty}^{\infty} I_k(z)\cos(k\beta), \tag{4.19}$$

where $z \in \mathbb{R}$ and I_k denotes the modified Bessel function of the first kind of order k.

Theorem 4.1 presents the direct expression for $\tilde{N}_{i,k}(\boldsymbol{\nu})$.

Theorem 4.1. *$\tilde{N}_{i,k}(\boldsymbol{\nu})$ is given by*

$$\tilde{N}_{i,k}(\boldsymbol{\nu}) = (2\pi)^q \sum_{n_1,n_2,\ldots,n_{q(q-1)/2}} \left\{ \prod_{m=1}^{q} I_{\hat{n}_m}(\kappa_{m,m}) \right\} \left\{ \prod_{m=1}^{q-1} \prod_{l=m+1}^{q} I_{n_{c_m+l-2m}}(\kappa_{l,m}) \right\} u_k(\alpha_q). \tag{4.20}$$

where, for $m = 1, \ldots, q$:

$$\begin{aligned}
\hat{n}_m &= \delta_{i-m} + \sum_{j=1}^{m-1} n_{c_j+m-2j} - \sum_{j=m+1}^{q} n_{c_m+j-2m}, \\
c_m &= (m-1)q - (m-1)(m-2)/2 + 1, \\
\alpha_q &= \sum_{l=1}^{q-1} \sum_{j=l+1}^{q} n_{c_l+j-2l}\theta_{l,j} - \sum_{j=1}^{q} \hat{n}_j \theta_{j,j}, \\
u_k(\alpha_q) &= \begin{cases} \cos(\alpha_q), & \text{if } k = 0, \\ \exp(j\alpha_q), & \text{otherwise}, \end{cases}
\end{aligned}$$

with δ_i denoting Kronecker delta.

Proof. See Appendix 4.B. □

In general for q phases, (4.20) implies the need of $q(q-1)/2$ nested summations. Thus, this method becomes computationally very expensive as the number of phases involved increases. Equation (4.20) requires each indexing variable of the summation to range from $-\infty$ to ∞. A finite summation suffices to obtain a close approximation to the converged value of the sum because of the decaying characteristics of I_k as $k \to \infty$. The number of terms that is needed to get a good approximation depends on the parameters of the integral (4.20). Larger values of the parameters $\kappa_{i,j}$ in (4.20) require more terms in the summation.

The main practical constraint for applying this method is the computational infeasibility of Bessel functions for large arguments. Bayesian update according to (4.13) results in the Bessel arguments required in the summation monotonically increasing at each iteration. After several iterations the computation of the summation becomes numerically infeasible and hence results in unstable estimates. Performing calculations on a logarithmic scale and normalising all the summation terms with the product of zero order Bessel functions as shown in equation (4.21) can help to some extent, but even that does not completely solve the problem of

numerical precision;

$$\log\left(\tilde{N}_{i,k}(\boldsymbol{\nu})\right) = q\log(2\pi) + \sum_{m=1}^{q-1}\sum_{l=m}^{q}\log\left(I_0(\kappa_{l,m})\right) + \log\left(\check{N}_{i,k}(\boldsymbol{\nu})\right), \tag{4.21}$$

where

$$\check{N}_{i,k}(\boldsymbol{\nu}) = \sum_{n_1,n_2,\ldots,n_{q(q-1)/2}} \left\{\prod_{m=1}^{q}\frac{I_{\hat{n}_m}(\kappa_{m,m})}{I_0(\kappa_{m,m})}\right\} \left\{\prod_{m=1}^{q-1}\prod_{l=m+1}^{q}\frac{I_{n_{c_m+l-2m}}(\kappa_{l,m})}{I_0(\kappa_{l,m})}\right\} u_k(\alpha_q).$$

A robust method for obtaining the logarithm of a modified Bessel function of the first kind is given in [Tanabe 2007].

Next, we describe an alternative method of calculating the same posterior statistics using an MCMC scheme.

4.4.3 Evaluating Posterior statistics using MCMC

The idea of MCMC is to design a Markov Chain such that its stationary distribution is same as the distribution from which the samples are required. The two most popular methods of achieving this goal are Metropolis-Hastings and Gibbs sampling [Robert 2004]. Gibbs sampling is a particularly attractive solution when samples from the conditional distributions are easy to obtain, but direct sampling from the joint distributions is not straightforward. The Gibbs method has the added advantage that its acceptance rate is 100%.

Interestingly, the posterior distribution $f(\boldsymbol{\psi}|\mathbf{y},\boldsymbol{\mu}_0)$ has all the full conditionals (the univariate distribution obtained by conditioning on all the variables except one from a multivariate distribution) as univariate VM; that is, for all j ($1 \leq j \leq q$), we have

$$f(\psi_j|\psi_1,\ldots,\psi_{j-1},\psi_{j+1}\ldots,\psi_q,\mathbf{y},\boldsymbol{\mu}_0) = \mathcal{VM}(\psi_j;\gamma_j;\delta_j)$$

where γ_j and δ_j are functions of $\psi_1,\ldots,\psi_{j-1},\psi_{j+1}\ldots,\psi_q,\mathbf{y},\boldsymbol{\mu}_0$.

Efficient methods exist to obtain samples from the VM distribution[Best 1979]. Therefore, Gibbs sampling is a natural choice for obtaining samples from the posterior. These samples can be used to calculate the point estimates as an alternative to the direct method.

In addition to the approximation of the point estimates, the normalising constant can also be approximated using an MCMC scheme by using the method pro-

posed by Chib [Chib 1995]. Chib's method uses the output from multiple Gibbs chains to evaluate the normalising constant.

Next, we describe how the normalising constant can be calculated using Chib's method when Gibbs sampling with two blocks is considered.

4.4.3.1 Chib's method illustrated for Gibbs sampling involving two blocks

Consider a posterior distribution $p(x_1, x_2|y)$. We are interested in approximating the normalising constant $p(y)$, which we express as

$$p(y) = \frac{p(y|x_1, x_2)p(x_1, x_2)}{p(x_1, x_2|y)}. \tag{4.22}$$

Note that the normalising constant does not depend on a particular value of x_1 or x_2; in other words, if we can obtain (or approximate) each of the three terms appearing on the right hand side of (4.22) for some particular value of $x_1 = x_1^*$ and $x_2 = x_2^*$, then the normalising constant can be found (or approximated) using (4.22). Usually, out of the three terms, the likelihood $p(y|x_1^*, x_2^*)$ and the prior $p(x_1^*, x_2^*)$ are known exactly. However, the third term $p(x_1^*, x_2^*|y)$ is known only up to a normalising constant (ironically, the normalising constant is exactly the quantity that we are interested in evaluating).

We write $p(x_1^*, x_2^*|y)$ as:

$$p(x_1^*, x_2^*|y) = p(x_1^*|x_2^*, y)p(x_2^*|y), \tag{4.23}$$

$$= p(x_1^*|x_2^*, y) \int p(x_2^*|x_1, y)p(x_1|y)dx_1. \tag{4.24}$$

Gibbs sampling proceeds by sampling, in turn, from the conditional marginal posterior distributions for each sampling block. For the two sampling block example considered here, the conditional marginal distributions are $p(x_1|x_2, y)$ and $p(x_2|x_1, y)$. Suppose, we have obtained G samples, $\{x_1^{(g)}, x_2^{(g)}\}_{g=1}^G$, after sufficient burn-in from the Markov Chain, then these samples can be used to make approximate inferences for the statistics of the posterior distribution $p(x_1, x_2|y)$. Thus, if we fix x_2 at x_2^*, then use the samples from the Gibbs chain, the integral appearing in (4.24) is approximated by

$$\int p(x_2^*|x_1, y)p(x_1|y)dx_1 \approx \frac{1}{G}\sum_{g=1}^{G} p(x_2^*|x_1^{(g)}, y). \tag{4.25}$$

We assume that the conditional marginal distribution $p(x_1|x_2, y)$ can be calculated (or approximated) for any given value of x_1 and x_2. Therefore $p(x_1^*, x_2^*|y)$ is

approximated by using (4.25) in (4.24). Finally, (4.24) is used in (4.22) to arrive at the desired quantity, $p(y)$.

Though x_1^* and x_2^* can be chosen arbitrarily, Chib recommends setting them at a mode such as the posterior mean, which is easily approximated by the Gibbs samples.

The general case involving more than two sampling blocks is discussed in detail in [Chib 1995]; this generalisation, which is required for approximating the normalising constant of GVM is presented in algorithmic form in Algorithm 4.1.

Algorithm 4.1: Normalising constant using MCMC

1 Draw $\psi_i^{(0)} \sim \mathcal{U}_{[0,2\pi)}(\cdot)$ for $i = 2, \ldots, q$;

2 **for** $i = 1$ **to** q **do**

3 Draw G samples (after a burn-in period) for each phase ψ_j for $i \leq j \leq q$ using the following $\mathcal{VM}$ distribution.

$$\psi_j^{(g)} \sim p(\psi_j^{(g)} | \psi_{i-1}^*, \ldots, \psi_1^*, \psi_{i+1}^{(g)}, \ldots, \psi_{j-1}^{(g)}, \psi_{j+1}^{(g-1)} \ldots, \psi_q^{(g-1)}, \mathbf{y}, \boldsymbol{\mu}_0)$$

4 Set $\psi_i^* = \frac{1}{G} \sum_{g=1}^{G} \psi_i^{(g)}$

5 Calculate

$$C_i = \frac{1}{G} \sum_{g=1}^{G} \pi(\psi_i^* | \psi_1^*, \ldots, \psi_{i-1}^*, \psi_{i+1}^{(g)}, \ldots, \psi_q^{(g)})$$

6 **return**

$$\frac{p(\boldsymbol{\psi}^* | \mathbf{y}, \boldsymbol{\mu}_0)}{\prod_{i=1}^{q} C_i}$$

The samples generated when $i = 1$ in Algorithm 4.1 are from a Markov Chain with the stationary distribution being the posterior $p(\boldsymbol{\psi}|\mathbf{y}, \boldsymbol{\mu}_0)$ and hence can be used to calculate the point estimates. This is a particular advantage of Chib's method; that is, some of the samples generated to calculate the normalising constant can also be used to obtain point estimates. Another important advantage of using the MCMC method compared to the direct method discussed in Section 4.4.2 is that the Bessel functions need not be evaluated, as is required in the direct method.

4.5 Results and discussion

In this section we assess the two methods in terms of performance and demonstrate the effect of:

- Number of variates in the GVM,
- Size of concentration parameters $\kappa_{i,j}$ for $1 \leq i \leq q$ and $i \leq j \leq q$,
- Amount of correlation between the phase variables.

We have chosen a simulation setup that easily enables us to control the parameters indicated above. Consider a signal of q complex exponentials with known frequency and amplitude (but unknown phase) embedded in complex white Gaussian noise. The n^{th} element ($1 \leq n \leq k$) of the measurement vector is given by

$$y_n = \sum_{i=1}^{q} e^{j\psi_i} a_i \exp(j\omega_i n) + w_n, \tag{4.26}$$

where $w_n \sim \mathcal{CN}(0, \sigma^2)$ and $\sigma^2 = E(w_n w_n^*)$.

The unknown phase variables ψ_i for $1 \leq i \leq q$ are treated as random variables. Note that the size of concentration parameters $\kappa_{i,j}$ can be altered by varying the complex noise variance σ^2. The correlation between the phases is easily controlled by varying the frequency separation between carrier frequencies.

For the tests that follow, the base settings are as follows:

$$\begin{aligned}
[\psi_1 \quad \psi_2 \quad \psi_3 \quad \psi_4] &= [\pi/6 \quad -\pi/4 \quad \pi/3 \quad \pi/5], \\
[a_1 \quad a_2 \quad a_3 \quad a_4] &= [0.2 \quad 0.135 \quad 0.185 \quad 0.23], \\
[\omega_1 \quad \omega_2 \quad \omega_3 \quad \omega_4] &= [0.5 \quad 0.75 \quad 1.0 \quad 1.25], \\
\sigma &= 0.2, \\
k &= 5.
\end{aligned}$$

Note that not all parameters may be used for some experiments which follow.

For convenience we define a test scenario using the triplet:

$$T = \{q, \sigma, \Delta\},$$

where,

$$\begin{aligned}
q &= \text{number of unknown variates included in the model,} \\
\sigma &= \text{complex noise variance in the measured signal,}
\end{aligned}$$

Δ = frequency separation between carriers.

The algorithms are assessed as a function of computational load. The computational complexity of evaluating (4.20) is $\mathcal{O}(M^{q(q-1)/2})$ where M is the truncation point in the infinite summation. Therefore, $M^{q(q-1)/2}$ is used in the performance analysis as the computational complexity of the direct method. For the MCMC method, complexity is measured by the number of samples drawn after burn-in for each Gibbs iteration.

In particular, we wish to find out the relative expense of each method when using it to:

- find point estimates,
- find the normalising constant.

Point estimation is performed as follows. First, we obtain $k = 5$ measurements under the test scenario at hand. Next, we use an uninformative (flat) prior and arrive at the posterior parameter vector ($\boldsymbol{\mu}_5$) by using the 5 measurements obtained. We use a GVM distribution with parameter vector $\boldsymbol{\mu}_5$ as the prior for processing a further 5 measurements to obtain point estimates under each method; that is, we use a GVM distribution parametrised by $\boldsymbol{\mu}_5$ as a prior to obtain approximations to $E\{\exp(j\boldsymbol{\psi})|\boldsymbol{\mu}_{10}\}$ using each method.

We define the error between the true posterior angle $\hat{\psi} = \arg(E\{\exp(j\psi)|\boldsymbol{\mu}_{10}\})$ and the approximate value $\hat{\hat{\psi}}$ to be $||\exp(j\hat{\psi}) - \exp(j\hat{\hat{\psi}})||$. The true posterior angle is approximated by using the Monte Carlo method with a very large sample size (10^5). As a consequence of this approximation, the MSE is bounded by the variance of the MCMC method's estimator with 100000 samples.

A similar procedure is followed to assess the normalising constant evaluation. As before, we obtain a suitable parameter vector $\boldsymbol{\mu}_5$ for the prior using $k = 5$ measurements. The predicted density for the next measurement $p(y_6|\boldsymbol{\mu}_5)$ is given by

$$\begin{aligned} p(y_6|\boldsymbol{\mu}_5) &= \int p(y_6|\boldsymbol{\psi}, \boldsymbol{\mu}_5) p(\boldsymbol{\psi}|\boldsymbol{\mu}_5) d\boldsymbol{\psi}, \\ &= \frac{b(y_6)}{\pi\sigma^2 N_0(\boldsymbol{\mu}_5)} \int \exp\left[\mathcal{R}\{\boldsymbol{\mu}_6(\mathbf{y}_{1:6})' \mathbf{u}(\boldsymbol{\psi})\}\right] d\boldsymbol{\psi}, \\ &= \frac{b(y_6) N_0\left[\boldsymbol{\mu}_6(\mathbf{y}_{1:6})\right]}{\pi\sigma^2 N_0(\boldsymbol{\mu}_5)}, \end{aligned} \tag{4.27}$$

where $\mathbf{y}_{1:6} = [y_1 \dots y_6]'$ and $\boldsymbol{\mu}_6(\cdot)$ is the updated parameter vector which follows from $\boldsymbol{\mu}_5$ according to (4.13).

We can obtain two approximations for the predicted measurement density by using each method to evaluate the normalising constants $N_0(\cdot)$ appearing in (4.27). Let $\hat{p}_1$ and $\hat{p}_2$ be the two approximations to the predicted measurement density obtained by using the direct method and the MCMC method respectively. A close approximation to the true predicted measurement density $p(y|\boldsymbol{\mu}_5)$ is an indication that the corresponding method performs well in calculating the normalising constant.

We use the Kullback-Leibler divergence (KLD) as a measure of closeness between two distributions. Denote the KLD between two distributions p and $\tilde{p}$ as $K(p, \tilde{p})$ where

$$K(p, \tilde{p}) = \int p(y) \log\left(\frac{p(y)}{\tilde{p}(y)}\right) dy. \tag{4.28}$$

Let $L(\hat{p}_1, \hat{p}_2)$ be defined as

$$L(\hat{p}_1, \hat{p}_2) = K(p(y_6|\boldsymbol{\mu}_5), \hat{p}_1(y)) - K(p(y_6|\boldsymbol{\mu}_5), \hat{p}_2(\mathbf{y})). \tag{4.29}$$

Thus $L(\hat{p}_1, \hat{p}_2)$ is the change in KLDs between the approximate distributions induced by the two methods with respect to the true predicted measurement density. Intuitively $L(\hat{p}_1, \hat{p}_2)$ can be thought of as the difference between "distances" of $\hat{p}_1$ and $\hat{p}_2$ from $p(y_6|\boldsymbol{\mu}_5)$. Note that KLD is not a true distance (metric) since in general $K(p, \hat{p}) \neq K(\hat{p}, p)$.

A positive value of L indicates that the MCMC method yields a better approximation than the direct method while a negative value of L implies the opposite. The Monte Carlo method to approximate L is given in Appendix 4.C.

In the next three subsections, we use the above procedures to assess the performance of the direct method and the MCMC method in calculating point estimates and normalising constants of the GVM.

4.5.1 Effect of number of variates

In order to infer the effect of number of variates in the GVM, we estimated ψ_1 under the following test scenarios and calculated the MSE:

- T_1={2,σ,0.25}
- T_2={3,σ,0.25}
- T_3={4,σ,0.25}

The effect of the number of variates on the MSE in obtaining point estimates for ψ_1 under each method is presented in Figure 4.2. From these results, it is clear

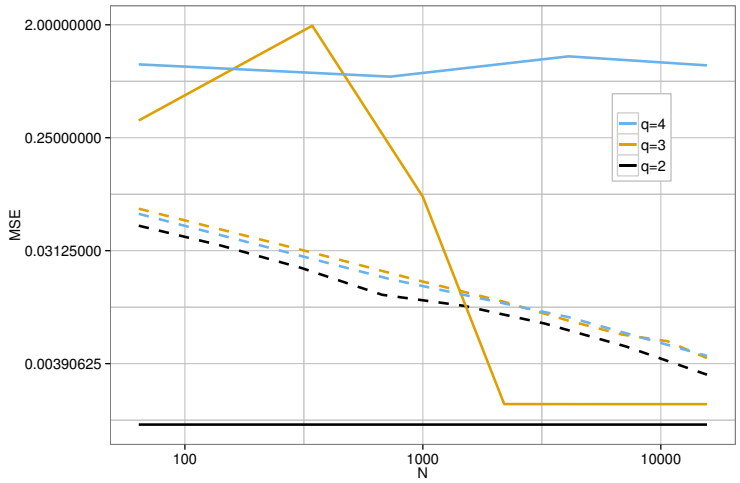


Figure 4.2: The effect of number of variates on estimating ψ_1 plotted against expense N. Solid lines correspond to the direct method while the dashed lines correspond to MCMC method.

that for both methods MSE decreases as the complexity N increases. The MCMC method is less sensitive to a change in the number of variables in terms of MSE compared to the direct method. The direct method fails in particular for relatively small numbers of truncation points, which is attributed to ignoring significant terms in the infinite summation (4.20).

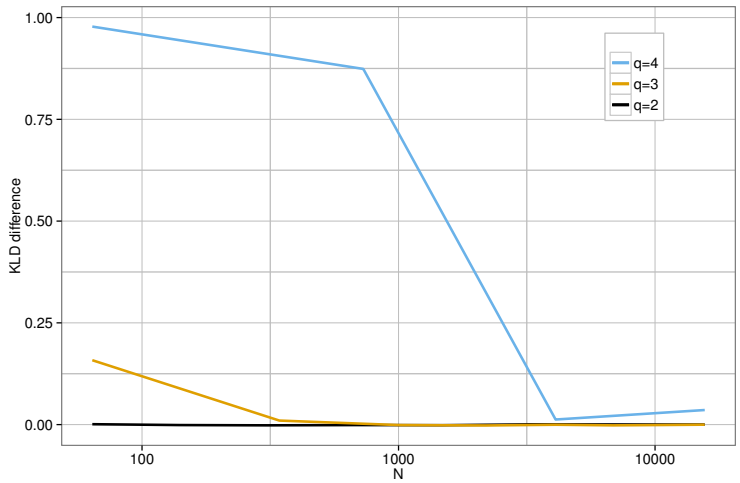


Figure 4.3: The effect of number of variates on the KLD difference $L(\hat{p}_1, \hat{p}_2)$ plotted against expense N.

Figure 4.3 plots the KLD difference $L(\hat{p}_1, \hat{p}_2)$ for setups T_1, T_2 and T_3. The results suggest that the performance of the direct method worsens when q increases.

This confirms the expected behaviour since truncating the summations in (4.20) ignores more significant terms when q is increased. However, when enough expense is allowed both methods respond to a change in q the same way as indicated by L reaching 0 asymptotically.

4.5.2 Effect of size of concentration parameters $\kappa_{i,j}$

Larger $\kappa_{i,j}$ values result in a concentrated GVM distribution. The size of these concentration parameters can be altered by varying the noise variance in the observed signal. The noise variance has an inverse relationship with these parameters. We have devised the following tests to analyse the effect of the size of concentration parameters:

- T_4={3,$\sigma/\sqrt{2}$,0.25}
- T_5={3,σ,0.25}
- T_6={3,$\sqrt{2}\sigma$,0.25}

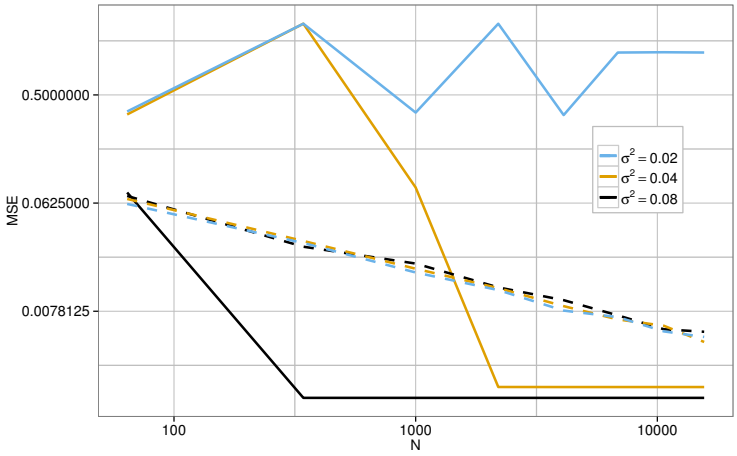


Figure 4.4: The effect of size of concentration parameters on estimating ψ_1 plotted against expense N. Solid lines correspond to the direct method while the dashed lines correspond to MCMC method.

Figure 4.4 summarises the mean errors in estimates of ψ_1 in experiments T_4, T_5 and T_6. Again the direct method is more sensitive to changes in the size of κ values than the MCMC method. Performance of the direct method is degraded when the size of the concentration parameters is increased. This is attributed to the fact that the number of significant terms contributing to equation (4.20) increases when the size of κ is increased. The effect of κ values on the normalising constant calculation is similar to that of point estimates as evidenced by Figure 4.5.

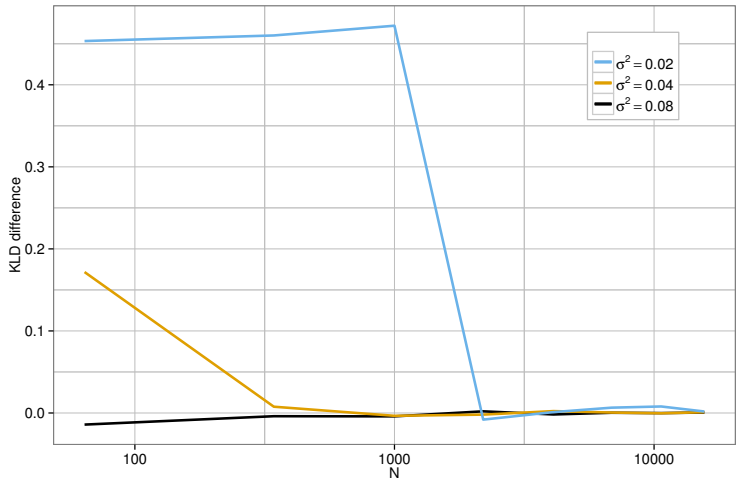


Figure 4.5: Effect of size of concentration parameters on KLD difference $L(\hat{p}_1, \hat{p}_2)$ plotted against expense N.

4.5.3 Effect of correlation between the phase variables

Correlation between phase variables can be easily controlled in our test set-up by altering the spacing between the carrier frequencies. Larger spacing results in less correlation and vice versa. Thus, the following set of tests are used to analyse the performance of the two methods:

- T_7={3,σ,1/8}
- T_8={3,σ,1/4}
- T_9={3,σ,1/2}

Figures 4.6 and 4.7 illustrate the effect of correlation between phases on point estimation and normalising constant calculation respectively using the two methods. High cross correlation between phase i and j results in a high $\kappa_{i,j}$ value. The direct method then requires a large truncating point M, and therefore a large computational expense, to perform well. The Gibbs sampling based MCMC approximation also becomes worse as correlation between phases increases although the deterioration in performance is more graceful than for the direct method.

4.6 Conclusion

We have considered the problem of estimating phases of multiple superimposed signals in additive Gaussian noise. The estimation problem is solved via a Bayesian conjugate framework which requires a particular multivariate generalisation of the

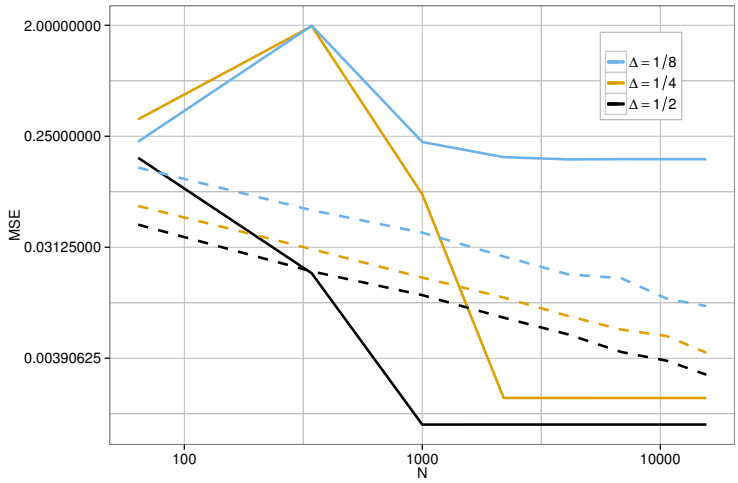


Figure 4.6: The effect of correlation between phases on estimating ψ_1 plotted against expense N. Solid lines correspond to the direct method while the dashed lines correspond to the MCMC method.

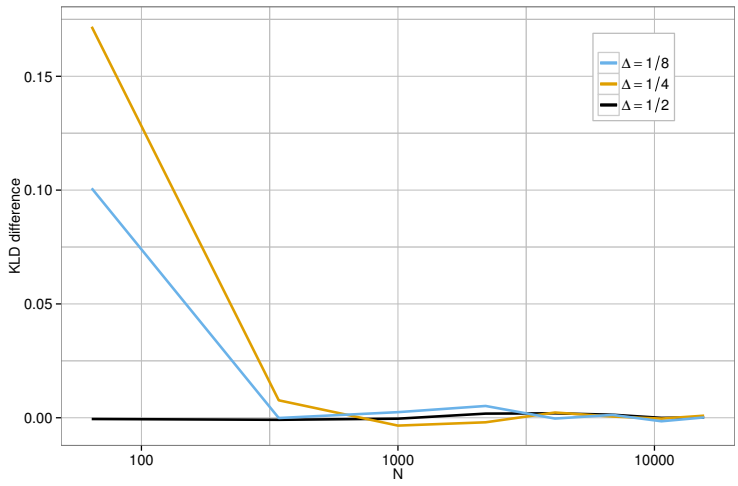


Figure 4.7: The effect of correlation between phases on KLD difference $L(\hat{p}_1, \hat{p}_2)$ plotted against expense N.

VM distribution. We have presented two methods, which can be used to calculate the posterior statistics of the proposed GVM distribution. The first method is a result of direct evaluation of an integral whereas the second method utilises MCMC methods. We have shown how the predicted measurement density can be obtained using the aforementioned methods. Performance of each method is assessed as a function of computational complexity subject to changes in parameters of the GVM distribution. The simulation results suggest that the performance of the di-

rect method is more sensitive to the parameters when compared to the MCMC based method. Further, when the allowed computational expense is restricted, in general the MCMC based method tends to perform better. However, the results suggest that when the allowed computational expense is increased, asymptotically, the difference in the performance of the methods becomes smaller.

Appendices

4.A Difference between circular and linear statistics

From an abstract perspective, a circular variable could be thought of as a point in $\mathbb{R}$ that is mapped on to a unique point on the unit circle defined on the field of complex numbers. These variables have the wrapping effect in the sense that, an increment of an integer multiple of 2π in the circular variable maps to the same point on the unit circle. Thus, without loss of generality, we can restrict a circular variable to a line segment of length 2π in $\mathbb{R}$ and define any probability measures on this line segment rather than on the entire real line (note that points at the beginning and end of the line segment represent the same point on the unit circle).

Because of this wrapping behaviour, the circular moments are defined differently than that of linear moments. For a circular variable ψ from a distribution $p(\psi)$ defined on Γ, where Γ is an interval in $\mathbb{R}$ of length 2π, the first moment m_1 and variance γ are defined by [Mardia 1972]:

$$m_1 = \int_{\psi\in\Gamma} \exp(j\psi)p(\psi)d\psi, \tag{4.30}$$

$$\gamma = 1 - |m_1|. \tag{4.31}$$

On the other hand, if ψ is a linear variable defined on Γ, the corresponding statistics are

$$m_1 = \int_{\psi\in\Gamma} \psi p(\psi)d\psi, \tag{4.32}$$

$$\gamma = \int (\psi - m_1)^2 p(\psi)d\psi. \tag{4.33}$$

Note that for circular variables the variance, which is a measure of spread, lies between 0 and 1, with 1 indicating maximum spread. To understand how the measure of spread works intuitively, think of a uniformly distributed phase variable (which has a maximum spread); the first moment m_1 is at the origin of $\mathbb{C}$ (due to the integration of $\exp(j\psi)$ over an interval of length 2π): thus, from (4.31) we have

the circular variance to be 1, which of course is to be expected for a quantity with a maximum spread. To phrase it a little more generally: the higher the circular spread, the smaller would be the size of the first moment vector and vice versa.

4.B Proof of Theorem 4.1

Consider evaluating $\tilde{N}_0(\boldsymbol{\nu})$ (Evaluating the more general $\tilde{N}_{i,k}(\boldsymbol{\nu})$ is similar).

Define the following quantities:

$$A_j(\psi_{j+1}, \psi_{j+2}, \ldots, \psi_q) \triangleq \prod_{i=j+1}^{q} \exp(\kappa_{i,i} \cos(\psi_i - \theta_{i,i})) \times \prod_{l=j+1}^{q-1} \prod_{m=l+1}^{q} \exp(\kappa_{m,l} \cos(\psi_l - \psi_m + \theta_{m,l})), \quad \text{for } j = 1, 2, \ldots, q-1, \tag{4.34}$$

$$M_j(\psi_{j+1}, \psi_{j+2}, \ldots, \psi_q) \triangleq \int \exp\left[\sum_{i=1}^{q} \kappa_{i,i} \cos(\psi_i - \theta_{i,i}) + \sum_{i=1}^{q-1} \sum_{j=i+1}^{q} \kappa_{i,j} \cos(\psi_i - \psi_j + \theta_{i,j})\right] \mathrm{d}\psi_1 \mathrm{d}\psi_2 \ldots \mathrm{d}\psi_j, \quad \text{for } j = 1, 2, \ldots, q, \tag{4.35}$$

$$= \int M_{j-1}(\psi_j, \psi_{j+1}, \ldots, \psi_q) \mathrm{d}\psi_j \qquad \text{for } j = 2, 3, \ldots, q. \tag{4.36}$$

From this point onwards, we denote $M_j(\psi_{j+1}, \psi_{j+2}, \ldots, \psi_q)$ simply by M_j, in favour of notational simplicity.

Note that $M_q = \tilde{N}_0(\boldsymbol{\nu})$. We carry out the integration sequentially over ψ_1 to ψ_q and in the process obtain M_1 to M_q. The first step is to integrate over ψ_1 and obtain M_1. We separate ψ_1 terms and express M_1 as

$$M_1 = \int A_1(\psi_2, \ldots, \psi_q) \left[\exp(\kappa_{1,1} \cos(\psi_1 - \theta_{1,1})) \prod_{m=2}^{q} \exp(\kappa_{1,m} \cos(\psi_1 - \psi_m + \theta_{1,m}))\right] \mathrm{d}\psi_1. \tag{4.37}$$

We then apply (4.19) and do the following manipulations:

$$M_1 = \int A_1(\cdot) \left(\sum_{n_1} I_{n_1}(\kappa_{1,1}) \cos\{n_1(\psi_1 - \theta_{1,1})\}\right)$$

$$\prod_{m=2}^{q}\sum_{n_{1,m}} I_{n_{1,m}}(\kappa_{1,m})\cos\{n_{1,m}(\psi_1-\psi_m+\theta_{1,m})\}\mathrm{d}\psi_1,$$
$$=\int A_1(\cdot)\left(\sum_{n_1} I_{n_1}(\kappa_{1,1})\cos\{n_1(\psi_1-\theta_{1,1})\}\right)$$
$$\times\left\{\sum_{n_{1,2},n_{1,3}\ldots,n_{1,q}}\prod_{m=2}^{q} I_{n_{1,m}}(\kappa_{1,m})\cos\{n_{1,m}(\psi_1-\psi_m+\theta_{1,m})\}\right\}\mathrm{d}\psi_1,$$
$$=\int A_1(\cdot)\sum_{n_1,n_{1,2},n_{1,3}\ldots,n_{1,q}} I_{n_1}(\kappa_{1,1})\left\{\prod_{m=2}^{q} I_{n_{1,m}}(\kappa_{1,m})\right\}$$
$$\cos\left\{\psi_1\left(\sum_{j=2}^{q} n_{1,j}+n_1\right)-\sum_{j=2}^{q} n_{1,j}(\psi_j-\theta_{1,j})-n_1\theta_{1,1}\right\}\mathrm{d}\psi_1. \tag{4.38}$$

Since the integration is performed on $\psi_1 \in (0, 2\pi]$, only the cosine terms of (4.38) in which the coefficient of ψ_1 is zero contribute to the result; that is

$$n_1+\sum_{j=2}^{q} n_{1,j}=0 \implies n_1=-\sum_{j=2}^{q} n_{1,j}. \tag{4.39}$$

Define $\tilde{n}_1$ as

$$\tilde{n}_1 \triangleq -\sum_{j=2}^{q} n_{1,j}. \tag{4.40}$$

Using $\tilde{n}_1$ to eliminate the nested summation over n_1 appearing in (4.38), we obtain

$$M_1=(2\pi)A_1(\cdot)\sum_{n_{1,2},n_{1,3}\ldots,n_{1,q}} I_{\tilde{n}_1}(\kappa_{1,1})\left\{\prod_{m=2}^{q} I_{n_{1,m}}(\kappa_{1,m})\right\}\cos\left\{-\sum_{j=2}^{q} n_{1,j}(\psi_j-\theta_{1,j})\right.$$
$$\left.-\tilde{n}_1\theta_{1,1}\right\}. \tag{4.41}$$

If we assume the general form of M_t for $t=1,2,\ldots,q-1$ to be

$$M_t=(2\pi)^t A_t(\cdot)\sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{t,t+1}\ldots n_{t,q}}\left\{\prod_{m=1}^{t} I_{\tilde{n}_m}(\kappa_{m,m})\right\}\left\{\prod_{l=1}^{t}\prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m})\right\}$$
$$\times\cos\left\{-\sum_{i=1}^{t}\sum_{j=t+1}^{q} n_{i,j}\psi_j+\sum_{i=1}^{t}\sum_{j=i+1}^{q} n_{i,j}\theta_{i,j}-\sum_{j=1}^{t}\tilde{n}_j\theta_{j,j}\right\}, \tag{4.42}$$

where

$$\tilde{n}_j=\sum_{i=1}^{j-1} n_{i,j}-\sum_{i=j+1}^{q} n_{j,i}, \tag{4.43}$$

then M_{t+1} is given by

$$
\begin{aligned}
M_{t+1} &= \int M_t d\psi_{t+1}, \\
&= (2\pi)^t A_{t+1}(\cdot) \int \sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{t,t+1}\ldots n_{t,q}} \left\{ \prod_{m=1}^{t} I_{\tilde{n}_m}(\kappa_{m,m}) \right\} \left\{ \prod_{l=1}^{t} \prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m}) \right\} \\
&\quad \times \cos\left\{ -\sum_{i=1}^{t} \sum_{j=t+1}^{q} n_{i,j}\psi_j + \sum_{i=1}^{t} \sum_{j=i+1}^{q} n_{i,j}\theta_{i,j} - \sum_{j=1}^{t} \tilde{n}_j \theta_{j,j} \right\} \\
&\quad \times \Bigg[\exp(\kappa_{t+1,t+1}\cos(\psi_{t+1} - \theta_{t+1,t+1})) \\
&\quad \times \prod_{m=t+2}^{q} \exp(\kappa_{t+1,m}\cos(\psi_{t+1} - \psi_m + \theta_{t+1,m})) \Bigg] \mathrm{d}\psi_{t+1}. \qquad (4.44)
\end{aligned}
$$

Using (4.19) on the exponential terms appearing in (4.44), we have

$$
\begin{aligned}
M_{t+1} &= (2\pi)^t A_{t+1}(\cdot) \int \sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{t,t+1}\ldots n_{t,q}} \left\{ \prod_{m=1}^{t} I_{\tilde{n}_m}(\kappa_{m,m}) \right\} \left\{ \prod_{l=1}^{t} \prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m}) \right\} \\
&\quad \times \cos\left\{ -\sum_{i=1}^{t} \sum_{j=t+1}^{q} n_{i,j}\psi_j + \sum_{i=1}^{t} \sum_{j=i+1}^{q} n_{i,j}\theta_{i,j} - \sum_{j=1}^{t} \tilde{n}_j \theta_{j,j} \right\} \\
&\quad \times \left(\sum_{n_{t+1}} I_{n_{t+1}}(\kappa_{t+1,t+1}) \cos\{n_{t+1}(\psi_{t+1} - \theta_{t+1,t+1})\} \right) \\
&\quad \times \Bigg\{ \sum_{n_{t+1,t+2}, n_{t+1,t+3}\ldots, n_{t+1,q}} \prod_{m=t+2}^{q} I_{n_{t+1,m}}(\kappa_{t+1,m}) \\
&\quad \times \cos\{n_{t+1,m}(\psi_{t+1} - \psi_m + \theta_{t+1,m})\} \Bigg\} \mathrm{d}\psi_{t+1}, \qquad (4.45) \\
&= (2\pi)^t A_{t+1}(\cdot) \int \sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{t+1,t+2}\ldots n_{t+1,q}} \left\{ \prod_{m=1}^{t} I_{\tilde{n}_m}(\kappa_{m,m}) \right\} I_{n_{t+1}}(\kappa_{t+1,t+1}) \\
&\quad \times \left\{ \prod_{l=1}^{t+1} \prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m}) \right\} \\
&\quad \times \cos\left\{ \psi_{t+1} \left(n_{t+1} - \sum_{i=1}^{t} n_{i,t+1} + \sum_{i=t+2}^{q} n_{t+1,i} \right) - \sum_{i=1}^{t+1} \sum_{j=t+2}^{q} n_{i,j}\psi_j \right. \\
&\quad \left. + \sum_{i=1}^{t+1} \sum_{j=i+1}^{q} n_{i,j}\theta_{i,j} - \sum_{j=1}^{t} \tilde{n}_j \theta_{j,j} - n_{t+1}\theta_{t+1,t+1} \right\} \mathrm{d}\psi_{t+1}. \qquad (4.46)
\end{aligned}
$$

By the use of the same argument in obtaining (4.39), we see that only the terms of which the coefficient of ψ_{t+1} in (4.46) equals 0 contribute to the result of the

integration over ψ_{t+1}. The coefficient of ψ_{t+1} is zero when $n_{t+1} = \tilde{n}_{t+1}$. We use $\tilde{n}_{t+1}$ to eliminate the summation over n_{t+1} and obtain the result after integrating over ψ_{t+1} as

$$M_{t+1} = (2\pi)^{t+1} A_{t+1}(\cdot) \sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{t+1,t+2}\ldots n_{t+1,q}} \left\{ \prod_{m=1}^{t+1} I_{\tilde{n}_m}(\kappa_{m,m}) \right\} \times \left\{ \prod_{l=1}^{t+1} \prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m}) \right\} \cos \left\{ -\sum_{i=1}^{t+1} \sum_{j=t+2}^{q} n_{i,j}\psi_j + \sum_{i=1}^{t+1} \sum_{j=i+1}^{q} n_{i,j}\theta_{i,j} - \sum_{j=1}^{t+1} \tilde{n}_j \theta_{j,j} \right\}. \tag{4.47}$$

Note that (4.47) and (4.42) share the same mathematical form. Further, observe that the expression for M_1 in (4.41) is a special case of the generic form (4.47). Thus, it follows by mathematical induction that (4.42) holds true for $t = 1, 2, \ldots, q-1$.

Finally, $\tilde{N}_0(\boldsymbol{\nu}) = M_q$ is obtained by setting $t = q-1$ in (4.42) as

$$M_q = (2\pi)^q \sum_{n_{1,2}\ldots n_{1,q},\ldots\ldots,n_{q-1,q}} \left\{ \prod_{m=1}^{q} I_{\tilde{n}_m}(\kappa_{m,m}) \right\} \left\{ \prod_{l=1}^{q-1} \prod_{m=l+1}^{q} I_{n_{l,m}}(\kappa_{l,m}) \right\} \times \cos \left\{ \sum_{i=1}^{q-1} \sum_{j=i+1}^{q} n_{i,j}\theta_{i,j} - \sum_{j=1}^{q} \tilde{n}_j \theta_{j,j} \right\}. \tag{4.48}$$

The exact form of the formula given in Theorem 4.1 follows from re-labelling the variables $\{n_{1,2}, n_{1,3}, \ldots, n_{1,q}, n_{2,3}, n_{2,4}, \ldots\ldots, n_{q-1,q}\}$ as $\{n_1, n_2, \ldots, n_{q(q-1)/2}\}$.

4.C Calculating $L(\hat{p}_1, \hat{p}_2)$

We propose a Monte Carlo method to approximate $L(\hat{p}_1, \hat{p}_2)$ as follows:

$$\begin{aligned} L(\hat{p}_1, \hat{p}_2) &= \int p(y|\boldsymbol{\mu}_5) \log\left(\frac{p(y|\boldsymbol{\mu}_5)}{\hat{p}_1(y)}\right) dy - \int p(y|\boldsymbol{\mu}_5) \log\left(\frac{p(y|\boldsymbol{\mu}_5)}{\hat{p}_2(y)}\right) dy, \\ &= \int \log\left(\frac{\hat{p}_2(y)}{\hat{p}_1(y)}\right) p(y|\boldsymbol{\mu}_5) dy, \\ &= E_{y|\boldsymbol{\mu}_5}\left[\log\left(\frac{\hat{p}_2(y)}{\hat{p}_1(y)}\right)\right], \\ &\approx 1/n \sum_{i=1}^{n} \log\left(\frac{\hat{p}_2(y^{(i)})}{\hat{p}_1(y^{(i)})}\right), \end{aligned} \tag{4.49}$$

where $y^{(i)}$ for $1 \leq i \leq n$ is the i^{th} sample from the predicted measurement density $p(y|\boldsymbol{\mu}_5)$.

The required samples from $p(y|\boldsymbol{\mu}_5)$ can be easily generated as demonstrated below:

$$\begin{aligned} p(y, \boldsymbol{\psi}|\boldsymbol{\mu}_5) &= p(y|\boldsymbol{\psi}, \boldsymbol{\mu}_5)p(\boldsymbol{\psi}|\boldsymbol{\mu}_5), \\ &= \mathcal{CN}(y; h(\boldsymbol{\psi}); \sigma^2)\mathcal{GMV}(\boldsymbol{\psi}|\boldsymbol{\mu}_5), \end{aligned} \tag{4.50}$$

where $h(\boldsymbol{\psi})$ is the measurement function corresponding to equation (4.26). Generating the i^{th} sample from $p(y|\boldsymbol{\mu}_5)$ can be accomplished in two steps:

- **step 1**: Generate a sample $\boldsymbol{\psi}^{(i)}$ from a Markov Chain with the stationary distribution being $\mathcal{GMV}(\boldsymbol{\psi}|\boldsymbol{\mu}_5)$ using the Gibbs sampler. This process is trivial since all the full conditional distributions(the univariate distribution obtained by conditioning all the variables but one from a multivariate distribution) are VM distributed as discussed in Section 4.4.3,
- **step 2**: Generate a sample $y^{(i)}$ from $\mathcal{CN}(y; h(\boldsymbol{\psi}^{(i)}); \sigma^2)$.

Sample $y^{(i)}$ can be used to infer any statistic from the desired distribution $p(y|\boldsymbol{\mu}_5)$.

CHAPTER 5

A particle filter for a partially known multipath environment

Summary

In Chapter 3 we introduced a novel multipath model, where the inherent uncertainty in the environment is taken into account in the radar measurement equation. In particular, recall that the uncertainty in the location of the walls is captured in the model by introducing a uniformly distributed random phase shift into the radar equation. Further, we relaxed the assumption that the reflectivity factors of the walls and the target are known; they are considered as random parameters. In this chapter, we propose a Markov Chain Monte Carlo (MCMC) based particle filter as a solution to this challenging filtering problem of estimating the target state amidst multipath reflections.

5.1 Introduction

In Chapter 1, we discussed the changing perspective on multipath in the radar signal processing research community. In particular, recall that the conventional approach towards addressing multipath is to treat it as a nuisance during the filtering process [Rigling 2008, Mecca 2006]. Recent studies [Krolik 2006, Chakraborty 2010, Chakraborty 2011, Hayvaci 2012b, Hayvaci 2012a, Li 2011, Sen 2007, Sen 2011] indicate a deviation from this approach and promote multipath as a useful resource for tracking in urban environments. In order to exploit the information content in multipath we need good models of the multipath environment, and a sound formulation of the estimation problem. In Chapter 3, we proposed a novel model where the uncertainty in a realistic multipath environment is captured in the radar measurement equation. In this chapter, we propose a statistical filter to track a moving target under the proposed multipath model. First, we present a brief review of some of the relevant work appearing in the literature and also discuss how our work differs from those.

Among existing work, it is common to see geometrical restrictions imposed on the multipath environment. As an example in [Chakraborty 2010], multipath causing reflective surfaces are assumed to be parallel to each other. In [Sen 2007, Sen 2011] authors assume that all the multipath appear in a single range cell. This assumption is only valid in very specific situations such as when the multipath environment consists of a narrow canal. In [Hayvaci 2012a, Hayvaci 2012b] the number of multipaths available is limited to two. In the work presented in this chapter, we formulated the estimation problem without imposing any geometrical limitations on the environment. Additionally, our problem formulation accommodates an arbitrary (but known) number of walls (reflective surfaces) in the environment.

The existing literature can be broadly divided into two categories based on how the unknown parameters are perceived in the model: deterministic [Sen 2007, Sen 2011, Li 2011, Hayvaci 2012b] or random [Chakraborty 2010, Chakraborty 2011]. Treating the unknown parameters as random quantities has the advantage of incorporating prior knowledge and uncertainty into the model. However, it appears that the deterministic treatment of unknown parameters is commoner than the alternative stochastic treatment in the literature on multipath radar. In our work, we have stressed the incorporation of uncertainty about the environment. Thus, we have been led to the treatment of the unknown parameters as random quantities and thereby followed a Bayesian framework for inferencing tasks.

Not much work has been done particularly on the target tracking problem in urban multipath radar. The tracking problem involves recursive estimation of parameters as they evolve through time. Usually, the dynamic behaviour of the parameters is specified through a differential or difference equation known as the "process equation". Some examples of existing work on tracking a target in multipath radar environment are [Chakraborty 2010, Chakraborty 2011, Li 2011]. In contrast a static parameter formulation is considered in [Sen 2007, Sen 2011, Hayvaci 2012a, Krolik 2006]. In the work described in this chapter, we focus on the multipath target tracking problem while explicitly accounting for the uncertainty in the environment.

In Section 2.2.2, we discussed the two types of measurements (raw sensor measurements vs detection based measurements), which could be used for tracking. A common theme among most of the existing work on multipath radar is the use of radar detections as measurements for the target tracker [Chakraborty 2010, Chakraborty 2011, Li 2011]. However, recall that a recent study [Morelande 2007] suggest that using raw radar sensor measurements as input to the filter results in a lower PCRB. Additionally, the use of detections for tracking introduces the

data association problem, where the challenge is to map detections to physical paths. In [Chakraborty 2010, Chakraborty 2011] it is assumed that these associations have been optimally carried out. Motivated by these arguments, we opt to use raw radar sensor data as measurements in our tracker. Note that by using raw sensor data, the data association problem can be avoided since the measurement is a superposition of all the multipath signals and therefore the likelihood function can be specified without an exhaustive summation over the space of possible data associations.

At this point, we summarise the important aspects of our multipath model on which the target tracking filter proposed in this chapter is based. We do not impose any geometrical restrictions on the environment such as the obstacle walls being vertical, horizontal, or parallel. Once a radar signal is in contact with either the target or a wall, it is attenuated by a fraction we refer to as a "reflectivity factor". We treat the reflectivity factors of the walls and the target as unknown independent random quantities. Further, we assume that the location information is accurate only up to a few wavelengths. This uncertainty in the wall location information is modelled by introducing a uniformly distributed random phase shift to each of the multipath signals. From a Bayesian perspective, the flat prior (uniform distribution) imposed on the phase shifts is used to translate the uncertainty in the building locations into a mathematical model. The idea of introducing a uniformly distributed phase shift can also be seen in [Wilson 1999], but is used there in a different context. The assumptions that the locations of the walls are known up to a certain accuracy and the wall reflectivity factors are unknown random variables are quite practical. As an example, the available map of the terrain may contain errors which will be accounted for by the former assumption.

In accordance with this model of the environment, the state vector includes the phase variables and reflectivity factors in addition to the target dynamics. This amounts to a challenging filtering problem. Particularly, we emphasise that the uniform prior distribution imposed on the phase variables significantly contributes to the uncertainty of measurement. We have chosen to address this highly non-linear filtering problem by employing a particle filter.

We described the particle filter in detail in Section 2.1.4. In that discussion we introduced the typical procedure of generating particles using a Monte Carlo method known as importance sampling. However, it has been shown that there are drawbacks in using importance sampling such as divergence issues [Robert 2004, section 3.3.2], poor performance in high dimensional problems [Li 2005, Oh 1991] and difficulty in choosing a good importance density. It should be noted that importance sampling, though quite common, is not the only choice

for particle generation. Recently, researchers have considered MCMC methods to generate particles [Khan 2005]. MCMC techniques are quite attractive for particle generation because of the relative ease of constructing a Markov Chain with the stationary distribution being the desired empirical filtering density. MCMC methods such as Gibbs and Metropolis-Hastings (MH) algorithms are two simple and popular methods to construct a desired Markov Chain. In this chapter, we employ the MCMC based approach for particle generation because of these difficulties, which are especially pertinent to challenging high-dimensional filtering problems such as the one being addressed. However, it should be noted that MCMC methods are not without drawbacks such as the challenges in constructing a chain that converges quickly and diagnosing convergence [Kass 1998].

It is relevant to highlight some of the differences between the MCMC based particle filter proposed in [Khan 2005] and the one presented in this chapter. In [Khan 2005] the MH sampler was used with a proposal which adds, removes or predicts targets. Its disadvantages are that it relies on prior statistics and so is not influenced by the measurements. Further, it does not exploit the structure of a model, such as ours, in which full conditional marginals are available for most of the parameters and can be approximated quite accurately for the other parameters (kinematic states).

The dimension of our measurement vector is relatively large because of our use of raw sensor measurements. When using approximate algorithms such as particle filters, a single Bayesian update using a large measurement vector has the potential to introduce considerable error into the estimate [Oudjane 2000, (9)]. More generally the problem arises when a large amount of information (that is, signal to noise ratio is very high), all at once, is used on sub-optimal filtering algorithms. One solution to the problem is to gradually perform the Bayesian update, which is known as *progressive correction*. Progressive correction is achieved in two ways: partitioning the measurements into smaller batches and processing them sequentially as in [Chopin 2002] or decomposing the likelihood function into a series of products and processing each term sequentially as in [Oudjane 2000]. We have implemented progressive correction using the former approach of partitioning the measurement vector into smaller blocks.

Our contributions in this chapter can be summarised as follows. We address the multipath radar target tracking problem with emphasis on incorporating the environmental uncertainty. The tracker is implemented using an MCMC based particle filter, where we have exploited the fact that the full conditional marginal posterior distributions are readily available or can be well approximated. These concepts can be applied to any generic filtering problem where it is difficult to

design good importance distributions.

The rest of the chapter is organised as follows. We specify the multipath model introduced in Chapter 3 once again in Section 5.2 to safeguard the reader from flipping back and forth between chapters. The proposed particle filter design is presented in Section 5.3. Section 5.4 presents simulation results and a discussion followed by a conclusion to the chapter.

Some of the frequently used notations and acronyms are given in Table 5.1. Any notation/acronym which does not appear in Table 5.1 is defined at the first occurrence.

Table 5.1: Summary of common notations and acronyms.

Notation/Acronym	Description
$\mathbf{I}$	Identity matrix (The dimensions would be implied by the context unless explicitly stated through a subscript.)
$\otimes$	Kronecker product
$1:k$	$[1\ 2\ \ldots\ k]'\ k \in \mathbb{Z}^+$
$\mathbf{x_v}$	$[x_{v_1}\ x_{v_2} \ldots\ x_{v_N}]$ (x_n is the n^{th} element of $\mathbf{x}$. Here, the index vector $\mathbf{v}$ consist of N integer indices.)
$\mathcal{N}(\cdot;\boldsymbol{\Theta},\boldsymbol{\Lambda})$	Multivariate Gaussian distribution with parameters $\boldsymbol{\Theta}$ (mean) and $\boldsymbol{\Lambda}$ (covariance)
$\mathcal{CN}(\cdot;\boldsymbol{\Theta},\boldsymbol{\Lambda})$	Circular symmetric complex Gaussian distribution with parameters $\boldsymbol{\Theta}$ (mean) and $\boldsymbol{\Lambda}$ (covariance)
AOA	Angle of Arrival
BF	Bootstrap Filter
MCMC	Markov Chain Monte Carlo
MH	Metropolis-Hastings
OID	Optimum Importance Density
UT	Unscented Transformation

5.2 Modelling and notation

Consider a target travelling in an urban terrain. Multiple radar transmitters are placed at suitable locations to illuminate the radar environment. The radar sensors receive a superposition of multiple signals due to scattering of the transmitted signal.

The target state at time t_k is denoted by $\mathbf{x}_k = [x_k\ \dot{x}_k\ y_k\ \dot{y}_k]'$, where (x_k, y_k) is the target position in the Cartesian plane and $(\dot{x}_k, \dot{y}_k)$ are the respective velocities. The target state is assumed to transition from time t_{k-1} to t_k according to the following

dynamic equation:

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{w}_k, \qquad k = 1, 2, \ldots, \tag{5.1}$$

where

$$\mathbf{F}_k = \mathbf{I}_2 \otimes \begin{bmatrix} 1 & \tilde{T} \\ 0 & 1 \end{bmatrix}, \tag{5.2}$$

with $\mathrm{cov}(\mathbf{w}_{k_1}, \mathbf{w}_{k_2}) = \delta_{k_1 - k_2} \mathbf{Q}_k$ and

$$\mathbf{Q}_k = \mathbf{I}_2 \otimes \kappa \begin{bmatrix} \tilde{T}^3/3 & \tilde{T}^2/2 \\ \tilde{T}^2/2 & \tilde{T} \end{bmatrix}. \tag{5.3}$$

Here, $\tilde{T} = t_k - t_{k-1}$ is the state sampling period.

Consider the following setup:

- N transmitters are placed at suitable locations.
- Return signals are received by M uniform linear arrays with the m^{th} array composed of L_m elements.
- During the sampling period each transmitter transmits a sequence of P pulses of duration D and period T_1, with the first pulse being transmitted at t_k, for $k = 1, 2, 3, \ldots$
- Let $s_{k,n}$ denote the time series transmitted by the n^{th} transmitter at time t_k.
- Let the total number of paths between the n^{th} transmitter and the m^{th} receiver be denoted by $P_{n,m}$.
- Incoming data are sampled at a rate of T_2 seconds.
- The reflections off the walls are specular (that is, mirror reflections).
- The transmitters and receivers are coherent.
- The target is a point scatterer.
- There are B buildings in the terrain, which are numbered from 1 to B.
- Each time a multipath hits the b^{th} wall, the signal attenuates by a random reflectivity factor, which is distributed according to $\mathcal{N}(\cdot; \mu_b, \iota_b)$, where the prior parameters μ_b and ι_b are known.

- Each time a multipath hits the target, the signal attenuates by a random target reflectivity factor, which has the distribution $\mathcal{N}(\cdot; \mu_0, \iota_0)$, where the prior parameters μ_0 and ι_0 are known..
- The collection of all the wall and target reflectivity factors is denoted by the vector $\mathbf{z}_k$, and is assumed to be temporally uncorrelated.
- The locations of the buildings in the surveillance area are approximately known.
- Because of the uncertainty of the wall locations, the radar signal corresponding to the p^{th} path between the n^{th} transmitter and the m^{th} receiver is phase shifted by $\psi^p_{n,m}$, where
$$\psi^p_{n,m} \sim \mathcal{U}_{[0,2\pi)}.$$
The collection of all such phase variables between all the transmitters and receivers at time k is denoted by the vector $\boldsymbol{\psi}_k$. Let $\psi_{i,k}$ denote the i^{th} member of this vector.

The signal vector received by all the sensors at time $t_k + uT_2$ is

$$\mathbf{Y}_k(u) = \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; u) + \mathbf{e}(u), \text{ for } u = 0, \ldots, U-1, \tag{5.4}$$

where $\mathbf{e}(u)$ is a circular symmetric complex white Gaussian process with covariance matrix $2\sigma^2\mathbf{I}$, and

$$\mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; u) = \sum_{n=1}^{N} \begin{bmatrix} \boldsymbol{\mu}_{k,n,1}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; uT_2) \\ \vdots \\ \boldsymbol{\mu}_{k,n,M}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; uT_2) \end{bmatrix},$$

with the measurement function $\boldsymbol{\mu}_{k,n,m}(\cdot)$ for $n = 1, \ldots, N, m = 1, \ldots, M$, given by

$$\boldsymbol{\mu}_{k,n,m}(\mathbf{x}, \boldsymbol{\psi}, \mathbf{z}; t) = \sum_{p=1}^{P_{n,m}} \mathbf{g}^p_{n,m}(\mathbf{x}, \mathbf{z}; t) \exp(j\psi^p_{n,m}), \tag{5.5}$$

and

$$\begin{aligned} \mathbf{g}^p_{n,m}(\mathbf{x}, \mathbf{z}; t) = \alpha^p_{n,m}(\mathbf{x}, \mathbf{z}) s_{k,n}(t - \tau^p_{n,m}(\mathbf{x})) e^{j\nu^p_{n,m}(\mathbf{x})t} \\ \times \mathbf{a}_m(t; \theta^p_{n,m}(\mathbf{x}), \dot{\theta}^p_{n.m}(\mathbf{x})), \end{aligned} \tag{5.6}$$

with

$$\mathbf{a}_m(t; \theta, \dot{\theta}) = \begin{bmatrix} 1 & e^{-j\bar{d}_m r_t} & \ldots & e^{-j(L_m-1)\bar{d}_m r_t} \end{bmatrix}', \tag{5.7}$$

where

$$r_t = \cos\theta - \dot{\theta} t \sin\theta, \tag{5.8}$$
$$\bar{d}_m = 2\pi d_m / \lambda, \tag{5.9}$$
$$d_m = \text{the separation between the elements of the } m^{th} \text{ sensor array}, \tag{5.10}$$
$$\lambda = \text{the wavelength of the carrier signal.} \tag{5.11}$$

For the p^{th} path between the n^{th} transmitter and the m^{th} sensor array:

- $\alpha^p_{n,m}(\mathbf{x}, \mathbf{z})$ is the intensity of the return. This includes transmitted signal strength as well as path attenuation.
- $\theta^p_{n,m}(\mathbf{x})$ is the Direction of Arrival (DOA).
- $\dot{\theta}^p_{n,m}(\mathbf{x})$ is the rate of change of the DOA.
- $\tau^p_{n,m}(\mathbf{x})$ is the delay of the signal.
- $\nu^p_{n,m}(\mathbf{x})$ is the Doppler shift.

The mathematical expressions for the above five quantities are included in Appendix 5.A.

The entire measurement vector at time t_k is

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k) + \mathbf{e}, \tag{5.12}$$

where

$$\begin{aligned} \mathbf{Y}_k &= [\mathbf{Y}_k(0) \dots \mathbf{Y}_k(U-1)]', \\ \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k) &= [\mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; 0) \dots \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k; U-1)]', \\ \mathbf{e} &= [\mathbf{e}(0) \dots \mathbf{e}(U-1)]'. \end{aligned}$$

It is convenient to work with real vectors; thus, we split the complex measurement vector and complex $\mathbf{h}(\mathbf{x}_k)$ into real and imaginary parts, and concatenate them into real vectors as follows:

$$\mathbf{y}_k = [\mathrm{Re}(\mathbf{Y}_k)' \quad \mathrm{Im}(\mathbf{Y}_k)']', \tag{5.13}$$
$$\tilde{\mathbf{h}}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k) = [\mathrm{Re}\{\mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k)\}' \quad \mathrm{Im}\{\mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k)\}']'. \tag{5.14}$$

Let $\mathbf{y}^k = [\mathbf{y}_1 \ \mathbf{y}_2 \ \dots, \mathbf{y}_k]'$, denote the vector of all measurements up to and including time k. The main objective in tracking is to estimate the target state $\mathbf{x}_k$ after observing $\mathbf{y}^k$.

5.3 Theory/Methodology

Suppose a vector $\boldsymbol{\theta}_k \in \mathbb{R}^N$ can be partitioned into P components as

$$\boldsymbol{\theta}_k = [\boldsymbol{\theta}'_{1,k}, \boldsymbol{\theta}'_{2,k}, \ldots, \boldsymbol{\theta}'_{P,k}]'. \tag{5.15}$$

A simple method to construct a Markov Chain with $p(\boldsymbol{\theta}_k|\mathbf{y}^k)$ as its stationary distribution is the Gibbs sampling method [Geman 1984]. To use this method it is required to be able to draw samples from all the conditional marginal distributions. This means, sampling from $p(\boldsymbol{\theta}_{j,k}|\mathbf{y}^k, \boldsymbol{\theta}_{1:j-1,k}, \boldsymbol{\theta}_{j+1:P,k})$ for every $j = 1, 2, \ldots, P$; a luxury which is not computationally possible in general. A more general solution to designing a desired Markov chain is the MH algorithm [Metropolis 1953, Hastings 1970]. The MH algorithm samples from some proposal $q(\boldsymbol{\theta}_{j,k}|\mathbf{y}^k, \boldsymbol{\theta}_{1:j-1,k}, \boldsymbol{\theta}_{j+1:P,k})$, and accepts the new sample with a certain probability, such that the chain converges to the desired target density. Note that the Gibbs sampling method is a special case of the MH algorithm which occurs when the proposals are chosen as the full conditional marginals.

Consider the multipath filtering problem introduced in Section 5.2. The state vector at time k can be broadly divided into three components as: target kinematics $\mathbf{x}_k$, phase variables $\boldsymbol{\psi}_k$, and wall reflectivities $\mathbf{z}_k$. In the context of particle filtering, using the auxiliary variable framework described in Section 2.1.4, we have an additional variable i indicating the sample index.

We wish to construct a Markov chain with stationary distribution $p(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k, i|\mathbf{y}^k)$. The Markov chain is progressed from state $(\mathbf{x}, \boldsymbol{\psi}, \mathbf{z}, i)$ to a different state $(\mathbf{x}', \boldsymbol{\psi}', \mathbf{z}', i')$ over a series of draws using either Gibbs steps (when drawing from a full conditional marginal distribution is feasible) or MH steps which approximate a Gibbs step.

We present the discussion of the transition from $(\mathbf{x}, \boldsymbol{\psi}, \mathbf{z}, i)$ to $(\mathbf{x}', \boldsymbol{\psi}', \mathbf{z}', i')$ in four parts, as shown below:

$$(\mathbf{x}, \boldsymbol{\psi}, \mathbf{z}, i) \xrightarrow{\text{part 1}} (\mathbf{x}, \boldsymbol{\psi}', \mathbf{z}, i) \xrightarrow{\text{part 2}} (\mathbf{x}, \boldsymbol{\psi}', \mathbf{z}, i') \xrightarrow{\text{part 3}} (\mathbf{x}, \boldsymbol{\psi}', \mathbf{z}', i') \xrightarrow{\text{part 4}} (\mathbf{x}', \boldsymbol{\psi}', \mathbf{z}', i'). \tag{5.16}$$

5.3.1 Updating the phase vector

Consider the distribution $p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z})$. Bayes' rule leads to the following set of equations:

$$p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z}_k) \propto p(\mathbf{y}_k|\boldsymbol{\psi}_k, \mathbf{x}_k, \mathbf{z}_k) p(\boldsymbol{\psi}_k), \tag{5.17}$$

$$\propto \mathcal{CN}(\mathbf{Y}_k; \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k), 2\sigma^2\mathbf{I}). \tag{5.18}$$

By direct substitution of (5.5), followed by some simple manipulations, it can be shown that $p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z}_k)$ has the following form:

$$p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z}_k) \propto \exp\Big\{ \sum_{n_1=1}^{\Omega} \kappa_{n_1,n_1} \cos(\psi_{n_1,k} - \phi_{n_1,n_1}) + \sum_{n_1=1}^{\Omega-1} \sum_{n_2=n_1+1}^{\Omega} \kappa_{n_1,n_2} \cos(\psi_{n_1,k} - \psi_{n_2,k} + \phi_{n1,n2}) \Big\}, \tag{5.19}$$

where

Ω is the number of elements in $\boldsymbol{\psi}_k$,
κ_{n_1,n_2} and ϕ_{n_1,n_2} for $n_1 \leq n_2 \leq \Omega$ are functions of $\mathbf{x}_k, \mathbf{z}_k,$ and $\mathbf{y}_k$.

The formulae for κ_{n_1,n_2} and ϕ_{n_1,n_2} are included in Appendix 5.B. Equation (5.19) shows that $p(\boldsymbol{\psi}_k|\cdot)$ is the Generalised Multivariate von-Mises (GVM) considered in Chapter 4 and [Karunaratne 2012a]. Obtaining samples directly from (5.19) is not feasible. However, note that the full conditional marginal distributions for each of the phase variables are from the univariate von-Mises family; that is

$$p(\psi_{i,k}|\mathbf{y}_k, \mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}_{1:i-1,k}, \boldsymbol{\psi}_{i+1:\Omega,k}) = \mathcal{VM}(\psi_{i,k}; \mathbf{y}_k, \mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}_{1:i-1,k}, \boldsymbol{\psi}_{i+1:\Omega,k}). \tag{5.20}$$

Note that VM distribution is a *nice* distribution, where sampling can easily be performed [Best 1979]. An implication of (5.20), is that Gibbs sampling can be used, if the transition from $(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}, i)$ to $(\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}, i)$ is carried out over Ω Markov steps, where in each step only one phase is updated by drawing from a VM distribution.

The process is illustrated by the following state flow sequence.

$$(\mathbf{x}_k, \boldsymbol{\psi}_{1:\Omega,k}, \mathbf{z}_k, i) \rightarrow (\mathbf{x}_k, \psi'_{1,k}, \boldsymbol{\psi}_{2:\Omega,k}, \mathbf{z}_k, i) \rightarrow (\mathbf{x}_k, \boldsymbol{\psi}'_{1:2,k}, \boldsymbol{\psi}_{2:\Omega,k}, \mathbf{z}_k, i) \dashrightarrow (\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}_k, i)$$

The above procedure can be summarised as drawing $\boldsymbol{\psi}'_k$ from the proposal $q_{\boldsymbol{\psi}}(\boldsymbol{\psi}'_k|\boldsymbol{\psi}_k, \mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z})$, where

$$q_{\boldsymbol{\psi}}(\boldsymbol{\psi}'_k|\boldsymbol{\psi}_k, \mathbf{y}^k, \mathbf{x}_k, i, \mathbf{z}_k) = \prod_{j=1}^{\Omega} \mathcal{VM}(\psi'_{j,k}|\boldsymbol{\psi}'_{1:j-1,k}, \boldsymbol{\psi}_{j+1:\Omega,k}, \mathbf{y}_k, \mathbf{x}_k, \mathbf{z}_k). \tag{5.21}$$

5.3.2 Updating the particle index

This part involves moving the Markov Chain from the state $(\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}_k, i)$ to $(\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}_k, i')$, which is accomplished by drawing a particle index i' from $p(i'|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^k)$. First, the form of the discrete distribution $p(i'|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^k)$ is discussed.

Assume that we are regularising the particles at time $k-1$ using a Gaussian kernel. Then, the kernel density of $\mathbf{x}_{k-1}$ conditioned on a particle index i and the measurement history $\mathbf{y}^{k-1}$ is

$$\hat{p}_h(\mathbf{x}_{k-1}|i, \mathbf{y}^{k-1}) = \mathcal{N}(\mathbf{x}_{k-1}; \mathbf{x}^i_{k-1}, h^2\boldsymbol{\Sigma}), \tag{5.22}$$

where h is the smoothing bandwidth (set using the empirical formulae found in [Silverman 1986]) and $\boldsymbol{\Sigma}$ is the covariance of samples $\mathbf{x}^{(z)}_{k-1}$ for $z = 1, \ldots, Z$.

Now we apply Bayes' rule on $p(i'|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^k)$:

$$\begin{aligned}
p(i'|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^k) &\propto p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k) p(i|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^{k-1}), && (5.23)\\
&\propto p(\mathbf{x}_k|i', \mathbf{y}^{k-1}), && (5.24)\\
&= \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) \hat{p}_h(\mathbf{x}_{k-1}|i', \mathbf{y}^{k-1}) d\mathbf{x}_{k-1}, && (5.25)\\
&= \int \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k \mathbf{x}^{(i')}_{k-1}, \mathbf{Q}_k) \mathcal{N}(\mathbf{x}_{k-1}; \mathbf{x}^{i'}_{k-1}, h^2\boldsymbol{\Sigma}) d\mathbf{x}_{k-1}, && (5.26)\\
&= \mathcal{N}(\mathbf{x}_k; \mathbf{F}_k \mathbf{x}^{(i')}_{k-1}, \mathbf{Q}_k + h^2\boldsymbol{\Sigma}). && (5.27)
\end{aligned}$$

The result (5.27) is particularly important; we have a closed form solution to $p(i'|\mathbf{x}_k, \mathbf{z}_k, \boldsymbol{\psi}'_k, \mathbf{y}^k)$. Therefore, a Gibbs step is easily performed to accomplish part 2 of the transition, by evaluating (5.27) for all the particles, followed by randomly drawing a particle index with a probability proportional to those calculated values. It should be noted that the expense of drawing the auxiliary variable is $\mathcal{O}(Z^2)$.

5.3.3 Updating the reflectivity factors

Here, the objective is to progress the Markov Chain by updating the reflectivity factors from $\mathbf{z}_k$ to $\mathbf{z}'_k$. The reflectivity factors appear within the attenuation function $\alpha(\mathbf{x}, \mathbf{z})$, which in turn appears in the measurement function (5.6). The attenuation function $\alpha(\mathbf{x}, \mathbf{z})$ is explicitly defined in (5.46).

Consider an individual element of the vector $\mathbf{z}_k$. It is easy to see from (5.46) that this component affects the measurement function (5.6) linearly. This fact, together with a Gaussian prior distribution for the reflectivity factors, results in the full conditional marginal distribution of a single element in $\mathbf{z}_k$ being normally distributed. Thus, the Kalman filter explained in Section 2.1.3.3 is easily used to ob-

tain the full conditional marginal distribution. Consequently, the transition from $(\mathbf{x}, \boldsymbol{\psi}', \mathbf{z}_k, i')$ to $(\mathbf{x}, \boldsymbol{\psi}', \mathbf{z}'_k, i')$ is easily performed over a series of Gibbs steps: by updating one element at a time through sequentially drawing samples from full conditional marginal distributions. Recall that we used a similar procedure to update $\boldsymbol{\psi}_k$.

To describe the sampling procedure mathematically, let

$$\mathbf{z}_k = [z_{1,k}\ z_{2,k}, \ldots, z_{\Gamma,k}]',$$

where Γ is the number of reflectivity factors at time k. Then $\mathbf{z}'_k$ is drawn from the following distribution:

$$q_{\mathbf{z}}(\mathbf{z}'_k|\mathbf{x}_k, i', \boldsymbol{\psi}'_k, \mathbf{y}^k) = \prod_{j=1}^{\Gamma} p(z'_{j,k}|\mathbf{x}_k, i', \boldsymbol{\psi}'_k, \mathbf{z}'_{1:j-1,k}, \mathbf{z}_{j+1:\Gamma,k}), \tag{5.28}$$

$$= \prod_{j=1}^{\Gamma} \mathcal{N}(z'_{j,k}; \bar{z}_j, C_j), \tag{5.29}$$

where the parameters of the normal distribution $\bar{z}_j$ and C_j, are completely defined by the full conditional marginal distribution for $z'_{j,k}$; that is, $p(z'_{j,k}|\mathbf{x}_k, i', \boldsymbol{\psi}'_k, \mathbf{z}'_{1:j-1,k}, \mathbf{z}_{j+1:\Gamma,k})$. The parameters $\bar{z}_j$ and C_j follow from Kalman filter equations.

5.3.4 Updating the target kinematics

This part is concerned with the transition of the Markov Chain from $(\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}'_k, i')$ to $(\mathbf{x}'_k, \boldsymbol{\psi}'_k, \mathbf{z}'_k, i')$. The full conditional marginal distribution for $\mathbf{x}_k$ is not available in closed form; therefore, we use a MH step which approximates a Gibbs step. This involves finding an approximation to the full conditional marginal for the kinematic state.

We wish to bring to the attention of the reader the work presented by us in [Karunaratne 2012b], where the multipath filtering problem with known phases and reflectivity factors was considered. The full conditional marginal distribution for $\mathbf{x}_k$ considered in this section is exactly the posterior distribution for the filtering problem considered in [Karunaratne 2012b]; there, the posterior distribution for the target state was not known in closed form, but we proposed an efficient importance density based on the Unscented approximation to the OID.

Recall that the measurement vector is quite large as a result of the decision to use raw sensor data instead of radar detections. In such settings, the concept of progressive correction [Oudjane 2000, Morelande 2009] suggests performing the Bayesian correction process in multiple steps in order to reduce the approximation

error in particle filtering. Here, we adhere to that concept by splitting the large measurement block into smaller sub-blocks, and performing the Bayesian update sequentially over those sub-blocks. The procedure is explained next, along with how we have exploited the UT to approximate the full conditional marginal distribution $p(\mathbf{x}_k|\mathbf{y}^k, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$.

Let $\mathbf{y}_k$ and $\tilde{\mathbf{h}}(\mathbf{x}_k)$ be partitioned into J blocks with each block consisting of Λ elements as shown below:

$$\mathbf{y}_k = [\mathbf{y}'_{1,k} \quad \mathbf{y}'_{2,k} \quad \dots \quad \mathbf{y}'_{J,k}]', \tag{5.30}$$

$$\tilde{\mathbf{h}}(\mathbf{x}_k) = [\tilde{\mathbf{h}}_1(\mathbf{x}_k)' \quad \dots \quad \tilde{\mathbf{h}}_J(\mathbf{x}_k)']'. \tag{5.31}$$

Note that the two vectors should be partitioned consistently (that is, the i^{th} measurement partition $\mathbf{y}_{i,k}$ corresponds to the i^{th} partition $\tilde{\mathbf{h}}_i(\mathbf{x}_k)$ of the noiseless measurement vector $\tilde{\mathbf{h}}(\mathbf{x}_k)$).

Let the notation $\mathbf{y}_k^j$ denote the measurement vector consisting of measurement partitions 1 to j at time k. Assume that we have processed the measurement partitions 1 to $j-1$, so that the distribution $\hat{p}(\mathbf{x}_k|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$ is known (in the boundary case of $j = 1$ the relevant distribution is $\hat{p}(\mathbf{x}_k|\mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$). We intend to process the j^{th} partition to arrive at $p(\mathbf{x}_k|\mathbf{y}_k^j, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$.

Using Bayes' rule $p(\mathbf{x}_k|\mathbf{y}_k^j, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$ is given by

$$p(\mathbf{x}_k|\mathbf{y}_k^j, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i) = \frac{p(\mathbf{x}_k, \mathbf{y}_{k,j}|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)}{p(\mathbf{y}_{k,j}|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)}. \tag{5.32}$$

The distribution $p(\mathbf{x}_k, \mathbf{y}_{k,j}|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$ is not available in closed-form and hence we approximate it as by a normal density; that is

$$p(\mathbf{x}_k, \mathbf{y}_{j,k}|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i) \approx \mathcal{N}\left(\begin{bmatrix} \mathbf{x}_k \\ \mathbf{y}_{j,k} \end{bmatrix}; \boldsymbol{\mu}_{m,k}^j, \mathbf{C}_{i,k}^j\right), \tag{5.33}$$

where

$$\boldsymbol{\mu}_{i,k}^j = \begin{bmatrix} \hat{\mathbf{x}}_{i,k}^{j-1} \\ \hat{\mathbf{y}}_{i,k}^j \end{bmatrix},$$

$$\mathbf{C}_{i,k}^j = \begin{bmatrix} \mathbf{P}_{i,k}^{j-1} & \boldsymbol{\Psi}_{i,k}^j \\ \left\{\boldsymbol{\Psi}_{i,k}^j\right\}' & \mathbf{S}_{i,k}^j \end{bmatrix}.$$

Note that equation (5.33) implies that $(\hat{\mathbf{x}}_{i,k}^{j-1}, \mathbf{P}_{i,k}^{j-1})$ and $(\hat{\mathbf{y}}_{i,k}^j, \mathbf{S}_{i,k}^j)$ are moments

of the predicted state and measurement distributions:

$$p(\mathbf{y}_{j,k}|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i) \approx \mathcal{N}(\mathbf{y}_{j,k}; \hat{\mathbf{y}}_{i,k}^j, \mathbf{S}_{i,k}^j), \tag{5.34}$$

$$p(\mathbf{x}_k|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i) \approx \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{i,k}^{j-1}, \mathbf{P}_{i,k}^{j-1}). \tag{5.35}$$

The UT can be used to approximate the moments appearing in (5.33). How the UT is used in the Unscented Kalman filter is explained in Section 2.1.3. Basically, the moments of a non-linear function are approximated using the UT by evaluating the function at deterministic points known as sigma points, and assigning a weight (the sigma weight) to each of these points. These sigma points are transformed using the non-linear function, and their (weighted) sample statistics are used to approximate the moments of the non-linear function. There are many methods to choose sigma points and corresponding sigma weights, but we have used that suggested in [Julier 2000].

Suppose $\boldsymbol{\chi}_1, \ldots, \boldsymbol{\chi}_{\mathcal{S}}$ are $\mathcal{S}$ sigma points with weights given by $v_1, \ldots, v_{\mathcal{S}}$ respectively, and chosen to match the first two moments of $p(\mathbf{x}_k|\mathbf{y}_k^{j-1}, \mathbf{y}^{k-1}, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$. The procedure can be initiated by noting that the first two moments of $p(\mathbf{x}_k|\mathbf{y}_k^0, \mathbf{y}^{k-1}, i)$ are $\mathbf{F}\mathbf{x}_{k-1}^{(i)}$ and $\mathbf{Q}_k + h^2\boldsymbol{\Sigma}$ respectively.

Let $\boldsymbol{\mathcal{Y}}_m = \tilde{\mathbf{h}}_j(\boldsymbol{\chi}_m)$. The following equations provide expressions for moments appearing in (5.33):

$$\hat{\mathbf{y}}_{i,k}^j \approx \sum_{m=1}^{\mathcal{S}} v_m \boldsymbol{\mathcal{Y}}_m, \tag{5.36}$$

$$\mathbf{S}_{i,k}^j \approx \sigma^2\mathbf{I} + \sum_{m=1}^{\mathcal{S}} v_m (\boldsymbol{\mathcal{Y}}_m - \hat{\mathbf{y}}_{i,k}^j)(\boldsymbol{\mathcal{Y}}_m - \hat{\mathbf{y}}_{i,k}^j)', \tag{5.37}$$

$$\boldsymbol{\Psi}_{i,k}^j \approx \sum_{m=1}^{\mathcal{S}} v_m (\boldsymbol{\chi}_m - \hat{\mathbf{x}}_{i,k}^j)(\boldsymbol{\mathcal{Y}}_m - \hat{\mathbf{y}}_{i,k}^j)'. \tag{5.38}$$

With the moment approximations (5.36)-(5.38) in hand, we can now substitute (5.33) and (5.34) into (5.32), and do the necessary manipulations [Anderson 1979, Example 3.2] to obtain $p(\mathbf{x}_k|\mathbf{y}_k^j, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$ as

$$p(\mathbf{x}_k|\mathbf{y}_k^j, \boldsymbol{\psi}_k, \mathbf{z}_k, i) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{i,k}^j, \mathbf{P}_{i,k}^j), \tag{5.39}$$

where

$$\hat{\mathbf{x}}_{i,k}^j = \hat{\mathbf{x}}_{i,k}^{j-1} + \boldsymbol{\Psi}_{i,k}^j \left\{\mathbf{S}_{i,k}^j\right\}^{-1} (\mathbf{y}_{j,k} - \hat{\mathbf{y}}_{i,k}^j), \tag{5.40}$$

$$\mathbf{P}_{i,k}^j = \mathbf{P}_{i,k}^{j-1} - \boldsymbol{\Psi}_{i,k}^j \left\{\mathbf{S}_{i,k}^j\right\}^{-1} \left\{\boldsymbol{\Psi}_{i,k}^j\right\}'. \tag{5.41}$$

Once the last (J^{th}) measurement partition is processed, we have an approximation to the full conditional marginal density $p(\mathbf{x}_k|\mathbf{y}^k, \boldsymbol{\psi}_k, \mathbf{z}_k, i)$ as

$$p(\mathbf{x}_k|\mathbf{y}^k, \boldsymbol{\psi}_k, \mathbf{z}_k, i) \approx \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}^J_{i,k}, \mathbf{P}^J_{i,k}). \tag{5.42}$$

Let $q_{\mathbf{x}}(\mathbf{x}_k|\mathbf{y}^k, \boldsymbol{\psi}_k, \mathbf{z}_k, i) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}^J_{i,k}, \mathbf{P}^J_{i,k})$; we use $q_{\mathbf{x}}(\cdot)$ as an efficient proposal in the MH algorithm to update the target dynamics and complete the transition from $(\mathbf{x}_k, \boldsymbol{\psi}'_k, \mathbf{z}'_k, i')$ to $(\mathbf{x}'_k, \boldsymbol{\psi}'_k, \mathbf{z}'_k, i')$.

As our filter is based on MCMC, some implementation details on chain convergence related issues are discussed now. One of the main concerns in MCMC based methods is the rate of convergence of the Markov Chain [Kass 1998]. We have observed that for some initial values the mixing of samples in the Markov chain was poor. In such cases the chain was restarted with a different initial state. There are many methods developed for diagnosing chain convergence [Cowles 1996], but we use rather a simple method of monitoring the (approximated) effective sample size
[Arulampalam 2002].

5.4 Results and discussion

Consider the multipath environment illustrated in Figure 5.1. The environment consists of four walls, a single transmitter and a single receiver ($B = 4, M = 1, N = 1$). The Cartesian coordinates of the transmitter and the sensor are $(2060, 1425)$ and $(2120, 1620)$, respectively. The prior distribution for the target is assumed to be normal with mean $[2083\ 0\ 1280\ 12]'$ and covariance matrix $\mathbf{I}_2 \otimes \text{diag}([2\ 1/4])$, where $\text{diag}(\cdot)$ represents a diagonal matrix with its diagonal elements being specified by the argument to the function. The state sampling interval $\tilde{T}$ is set to 1 s. The process noise covariance matrix $\mathbf{Q}_k$ is fixed by setting $\kappa = 2.1$. The measurement noise covariance is set by $\sigma^2 = 0.4$. The trajectory of the target over 38 time scans is shown in Figure 5.1. Note that the trajectory of the target used for this setup does not follow the constant velocity motion model on which the filter is designed. Therefore, we have accounted for the model mismatch by assuming large enough uncertainty on the target dynamics through a sufficiently large process noise covariance matrix $\mathbf{Q}_k$. This model mismatch together with large process noise make the tracking problem even more challenging.

The signal transmitted from the n^{th} transmitter at time k is chosen as a linear

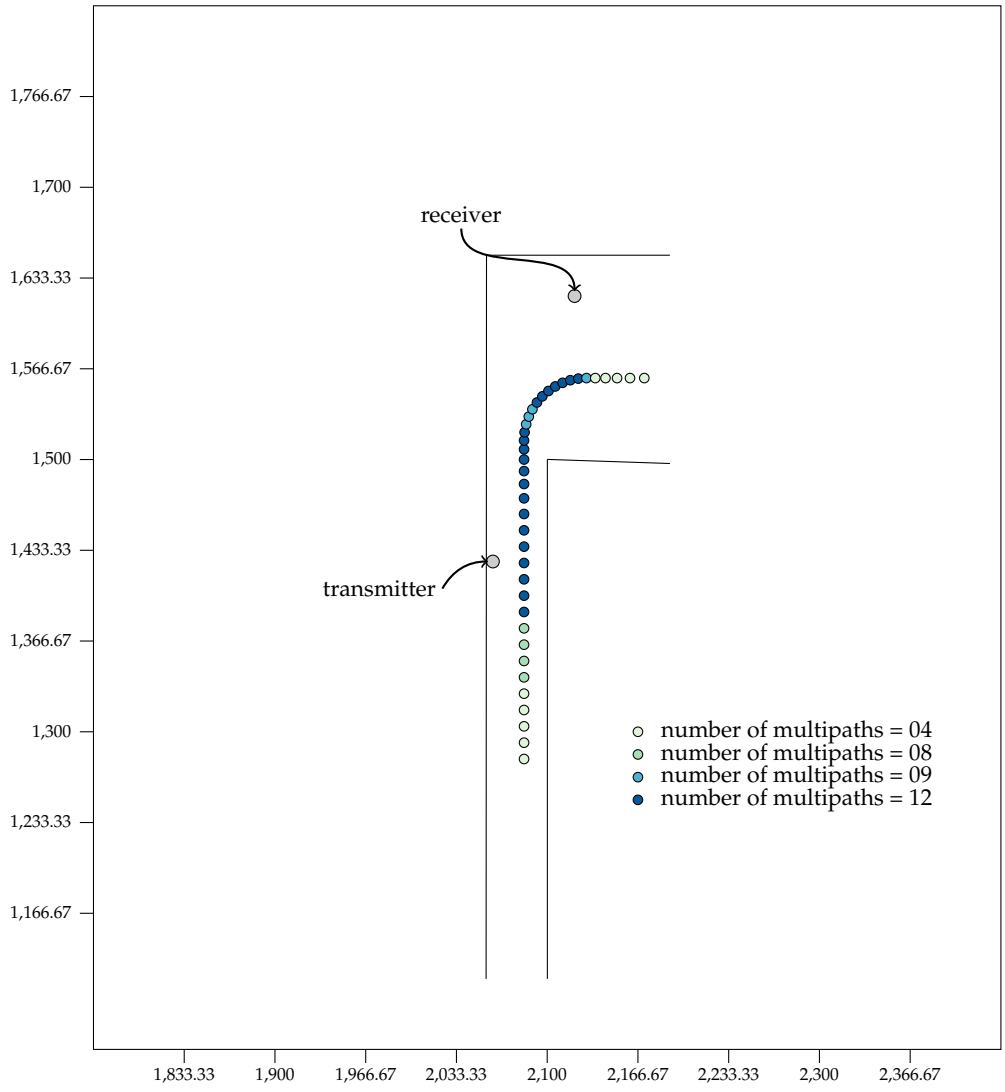


Figure 5.1: Simulation environment for a target travelling around a corner.

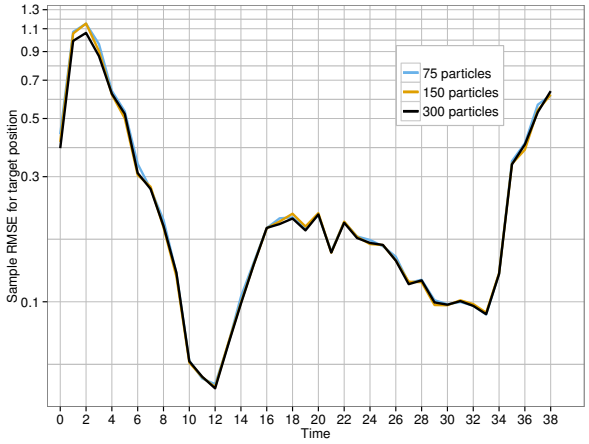
chirp signal with energy E given by

$$s_{k,n}(t) = \frac{\sqrt{E}}{\sqrt{P}} \sum_{p=-(P-1)/2}^{(P-1)/2} \frac{\exp\{[jv - 1/(2\rho^2)](t - pT_1)^2\}}{(\pi\rho^2)^{1/4}}. \tag{5.43}$$

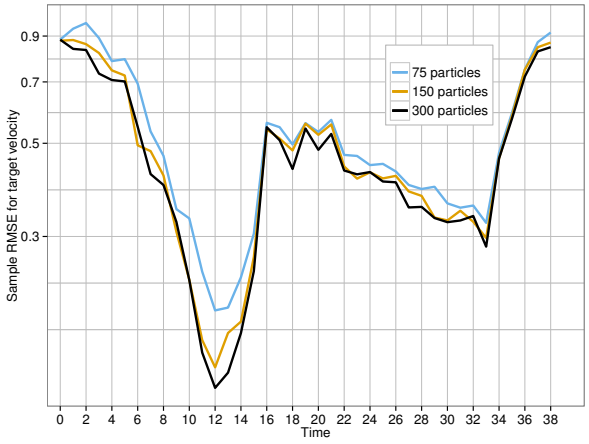
The number of P pulses transmitted is set to 3. The chirp parameter v is chosen such that the effective bandwidth of the signal is 40 MHz. The duration of the pulse is set to 250 ns, which is controlled by the parameter ρ. The interval between pulses T_1 is set to 100 μs. The number of elements of the sensor is fixed at $L_1 = 2$. The wavelength of the carrier frequency of the radar signal λ is set to 1.0 m. The signal energy E of the transmitted signal is calculated such that a desired average Signal to Noise Ratio (SNR) is achieved for the entire target trajectory. For each time scan of the trajectory, the SNR of the least attenuated path was considered to obtain the average SNR for the entire trajectory. We chose the desired SNR as 20dB at the output of the matched filter. The distribution parameters for the wall reflectivities are $\mu_b = 0.6$ and $\iota_b = .0025$ for $b \in \{1, 2, 3, 4\}$. The distribution parameters for the target reflectivity are $\mu_0 = 0.7$ and $\iota_0 = 0.0025$. The bandwidth for kernel regularisation was set using [Silverman 1986, p. 87] with further scaling by a factor of 0.5.

For generating simulation results in this section, we performed Monte Carlo experiments by repeating each simulation experiment 100 times, with the target trajectory fixed as in Figure 5.1. The reasons for fixing the trajectory instead of generating it each time through the stochastic kinetic motion model are twofold. First, we wanted to assess the filter against a challenging scenario such as one comprising a segment which includes the target taking a turn around a corner. Secondly, if the trajectory was randomly drawn each time from the target dynamics model, there is a chance that in some realisations the target would hit a wall.

One of the key parameters of particle based filters is the number of particles used. With the objective of assessing the sensitivity of the filter to sample size, we ran the simulation by varying the number of particles while keeping the rest of the parameters constant. The results are shown in Figure 5.2. Unsurprisingly the results suggest that the use of more particles leads to better performance. The effect of the number of particles is more visible in the RMSE graph for the velocity when compared against that of the position. The relatively small number of particles required for almost optimal performance is also evident from these results. An intuitive interpretation of the shape of the curves can be presented based on the target trajectory and the placement of the transmitter and the sensor as shown in Figure 5.1. The reduction of the error from time step 2 up to about 12 is due to the



(a)



(b)

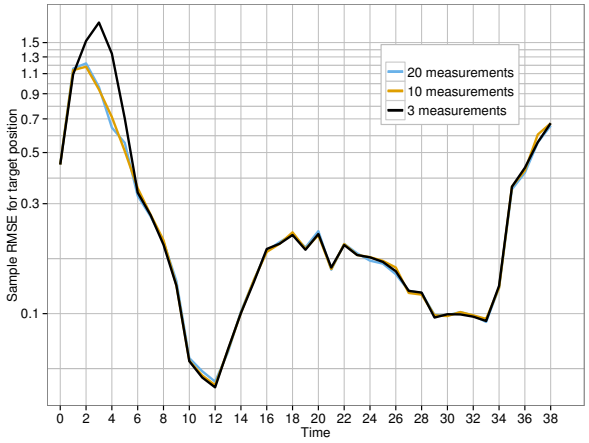
Figure 5.2: Results of the experiment to assess the effect of the number of particles. The diagram shows the RMSE of the filter for: (a) position in meters (b) velocity in meters per second.

target moving towards the transmitter and hence progressively producing strong reflections; the transmitter is closest to the target at time step 12. Afterwards, the error increases as it moves away from the transmitter until time step 18. The availability of more multipath and the fact that the target is approaching the sensor is the explanation for the reduction of error from time step 18 to 33. At $k = 33$, the target is closest to the sensor. During the last stages of the trajectory ($k = 34$ to 38) error increases due to the target shifting away from the sensor which leads to weak and fewer multipaths.

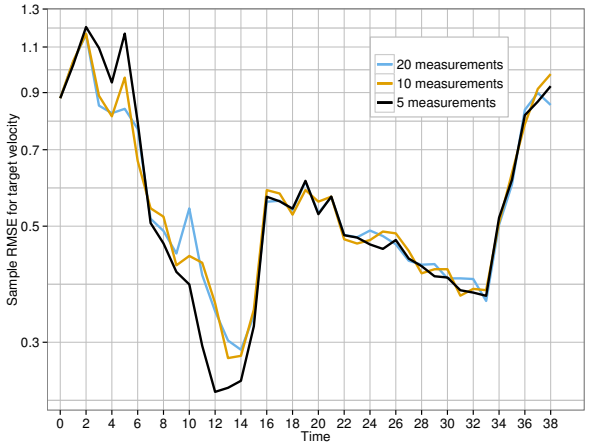
The use of progressive correction is a key element in the construction of an approximation to the full conditional marginal of the kinematic state. It is of some

interest to observe the effect of progressive correction on the performance of the particle filter. To that end, we varied the number of measurements (Λ) that were used in a single Bayesian update step (while keeping the number of particles fixed at 50). Here, a smaller number of measurements per update (small Λ) implies that more progressive correction is used. The results are shown in Figure 5.3, which prove to be interesting. Progressive correction appears to degrade the performance during the initial trajectory segment (around time steps 2 to 6) but has a significant favourable effect around time step 12. The explanation for this behaviour is as follows. There are two conflicting sources of error at play here. First, note that each time a progressive correction is made, we do a Gaussian approximation to the posterior distribution. Thus, the use of more progressive correction steps implies that the Gaussian approximation occurs more frequently leading to accumulation of error of this type. The second type of error, which progressive correction is intended to reduce is that introduced by the application of the Bayesian correction to an approximate prior rather than the true prior. This error is more severe in the presence of strong signals (see [Oudjane 2000, (9)]). Performing more progressive corrections does help to reduce this error but introduces the approximation error of the other type. The result in Figure 5.3 is an excellent illustration of the interplay between these two types of errors. During time steps 2 to 6, the received signals are relatively weak and so the error introduced at the Bayesian correction stage is not severe enough for progressive correction to be useful; in fact the error due to a large number of Gaussian approximations outweighs the benefit from progressive correction. This leads to performance degradation with more progressive corrections during the initial stage of the trajectory. On the other hand, around time step 12, the target is very close to the transmitter, which results in very strong signals. The progressive correction is more useful at this stage and the overall error reduced by using it far outweighs the error introduced by the use of more Gaussian approximations. We further noted that with less progressive correction, the Markov Chain had to be restarted many times to obtain a convergent chain, further confirming the difficulty faced by the filter in the presence of very strong signals.

For comparison purposes, we implemented the bootstrap particle filter for the multipath estimation problem. Results are shown in Figure 5.4. Unsurprisingly, the bootstrap filter, even with 1000 particles was not able to track the target. Although the results show that increasing the number of particles reduces the error in the bootstrap filter, it appears that a very large number of particles would be needed to achieve a comparable level of performance against the MCMC particle filter (with 150 particles). As discussed in Section 2.1.4, the main draw back of



(a)

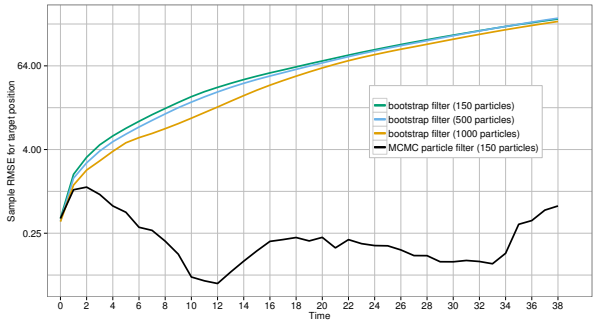


(b)

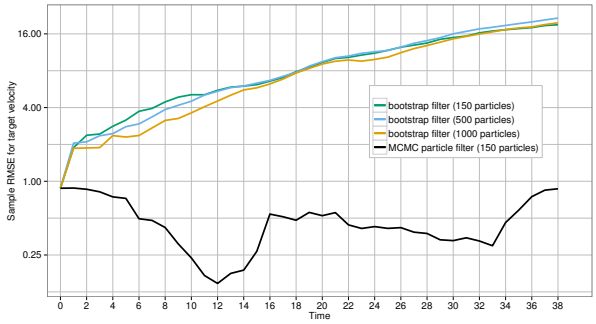
Figure 5.3: Results of the experiment to assess the effect of the progressive correction. The diagram shows the RMSE of the filter for: (a) position in meters (b) velocity in meters per second.

the bootstrap filter is that it does not use the current measurement for drawing the samples, thus performing very poorly in challenging filtering problems. However, it should be noted that the computational expense per particle is much lower in the BF, but this expense reduction does not make up for the much larger sample size required to get equivalent error performance to that of the MCMC particle filter.

In general, the simulation results suggest that the proposed filter is capable of tracking the target while exploiting multipath for a challenging scenario, despite the motion model of the filter not matching the dynamics of the simulated trajectory. We believe that introducing multiple motion models for different manoeuvres would enhance the filter performance. This could be easily accomplished



(a)



(b)

Figure 5.4: A comparison of the MCMC particle filter against the bootstrap particle filter. MCMC particle filter was configured with $\Lambda = 3$. The diagram shows the RMSE of the filters for: (a) position in meters (b) velocity in meters per second.

within the proposed MCMC framework by introducing an additional auxiliary variable to indicate the motion model used. However, the benefit of using multiple manoeuvring models comes with the cost of extra computation.

5.5 Conclusion

In this chapter, we have presented a statistical filter that exploits multipath for tracking a target in a multipath rich environment. The filter is based on a realistic model of the environment that takes account of uncertainty in the wall locations. This uncertainty is modelled by assuming that each multipath is phase shifted by a uniformly distributed random variable. Further, we have assumed that the wall and target reflectivities are unknown stochastic variables. A consequence of the assumed model is that the underlying filtering problem turned out to be very challenging. To address this challenging filtering problem, we have proposed an MCMC based particle filter. In the MCMC particle filter, the particles are extracted from a Markov Chain that has the desired posterior as the stationary dis-

tribution.The filter presented in this chapter is a testimony to the option of using MCMC for particle generation when it is difficult to design efficient importance densities. We have introduced kernel regularisation and progressive correction to the MCMC particle filter to improve the overall robustness of the tracker. The progressive correction is particularly beneficial when the filter is operating under high SNR, which might happen when the target is moving very close to the sensor or transmitter, or in a region with very strong multipath reflections. We have included simulation results involving a challenging setup, which includes the target taking a turn around a corner. The simulation results support the utility of the proposed MCMC filter.

Appendices

5.A Expressions for target quantities

For the p^{th} path between the n^{th} transmitter and the m^{th} receiver, consider the quantities: $\alpha^p_{n,m}(\mathbf{x}_k, \mathbf{z}_k), \theta^p_{n,m}(\mathbf{x}_k), \dot{\theta}^p_{n,m}(\mathbf{x}_k), \tau^p_{n,m}(\mathbf{x}_k)$, and $\nu^p_{n,m}(\mathbf{x}_k)$. For brevity, the subscripts identifying the specific transmitter, receiver, path, and time are dropped from the following discussion.

For the path under consideration assume that the location of the transmitter is (a_{tx}, b_{tx}) and that of the sensor is (a_{rx}, b_{rx}). Let the number of reflection points between the transmitter and target located at (x, y) be S, with the s^{th} reflection point denoted by (ω_s, ρ_s). The point corresponding to $s = 1$ is the reflection point immediately after the transmitter. Similarly, let the number of reflection points between the target and the sensor be L, with the l^{th} reflection point located at (ξ_l, ζ_l). The reflection point immediately before the sensor corresponds to $l = 1$. It is convenient to assign $(\omega_0, \rho_0) = (a_{tx}, b_{tx})$, $(\xi_0, \zeta_0) = (a_{rx}, b_{rx})$ and $(\omega_{S+1}, \rho_{S+1}) = (\xi_{L+1}, \zeta_{L+1}) = (x, y)$. Let the sequence of walls hit along the forward path be denoted by $q_1, \ldots, q_S$; similarly, let the sequence of walls hit along the return path be denoted by $u_1, \ldots, u_L$.

Define R_s and r_l to be the length of the s^{th} and l^{th} path segments in the forward and return journey of the signal respectively; then,

$$R_s = \sqrt{(\omega_s - \omega_{s-1})^2 + (\rho_s - \rho_{s-1})^2}, \tag{5.44}$$

$$r_l = \sqrt{(\xi_l - \xi_{l-1})^2 + (\zeta_l - \zeta_{l-1})^2}. \tag{5.45}$$

The attenuation factor $\alpha(\mathbf{x}, \mathbf{z})$ is given by

$$\alpha(\mathbf{x}, \mathbf{z}) = \sqrt{E} \frac{\rho \prod_{s=1}^{S+1} \varrho_s \prod_{l=1}^{L} \epsilon_l}{\sum_{s=1}^{S+1} R_s \sum_{l=1}^{L+1} r_l}, \tag{5.46}$$

where

$$\begin{aligned} E &\text{ is the energy of the transmitted signal,} \\ \varrho_s \sim \mathcal{N}(\cdot; \mu_{q_s}, \iota_{q_s}) &\text{ is a random wall reflectivity factor,} \\ \epsilon_l \sim \mathcal{N}(\cdot; \mu_{u_l}, \iota_{u_l}) &\text{ is a random wall reflectivity factor,} \\ \rho \sim \mathcal{N}(\cdot; \mu_0, \iota_0) &\text{ is the reflectivity factor of the target.} \end{aligned}$$

The AOA $\theta(\mathbf{x})$, at the sensor is given by

$$\theta(\mathbf{x}) = \arctan \left[\frac{\zeta_1 - b_{rx}}{\eta_1 - a_{rx}} \right]. \tag{5.47}$$

The function $\dot{\theta}(\mathbf{x})$ is the time derivative of (5.47), and the method to calculate it is given in Chapter 3 and in [Karunaratne 2011] .

The delay for the path is given by

$$\tau(\mathbf{x}) = 1/c \sum_{s=1}^{S+1} R_s + 1/c \sum_{r=1}^{L+1} r_l, \tag{5.48}$$

where c is the speed of light.

The doppler shift $\nu(\mathbf{x})$ is given by

$$\nu(\mathbf{x}) = \frac{2\pi(\dot{R}_{S+1} + \dot{r}_{L+1})}{\lambda}, \tag{5.49}$$

where the dot notation denotes differentiation with respect to time. Here again, the interested reader is referred to [Karunaratne 2011] or to Chapter 3.

5.B Full conditional marginal distribution of the phase vector

The full conditional marginal distribution for $\boldsymbol{\psi}_k$ is given by

$$p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, \mathbf{z}_k, i) \propto p(\mathbf{y}_k|\boldsymbol{\psi}_k, \mathbf{x}_k, \mathbf{z}_k)p(\boldsymbol{\psi}_k), \tag{5.50}$$
$$\propto p(\mathbf{y}_k|\boldsymbol{\psi}_k, \mathbf{x}_k, \mathbf{z}_k), \tag{5.51}$$
$$= \mathcal{CN}(\mathbf{Y}_k; \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k), 2\sigma^2\mathbf{I}). \tag{5.52}$$

Let $\boldsymbol{\Delta}_k = \mathbf{Y}_k - \mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k)$; then,

$$p(\boldsymbol{\psi}_k|\mathbf{y}^k, \mathbf{x}_k, \mathbf{z}_k, i) \propto \exp(-\boldsymbol{\Delta}_k^*(2\sigma^2\mathbf{I})^{-1}\boldsymbol{\Delta}_k), \tag{5.53}$$
$$\propto \exp\left(\frac{1}{\sigma^2}\left[\mathrm{Re}\{\mathbf{Y}_k^*\mathbf{h}(\cdot)\} - \mathbf{h}(\cdot)^*\mathbf{h}(\cdot)\right]\right), \tag{5.54}$$

where the operator "*" denotes conjugate transpose, and the arguments of the function $\mathbf{h}(\mathbf{x}_k, \boldsymbol{\psi}_k, \mathbf{z}_k)$ have been dropped for brevity. Now, we focus on the terms $\mathbf{Y}_k^*\mathbf{h}(\cdot)$ and $\mathbf{h}(\cdot)^*\mathbf{h}(\cdot)$ appearing in the above equation. By using (5.4) and (5.5) we have

$$\mathbf{Y}_k^*\mathbf{h}(\cdot) = \sum_{u=0}^{U-1}\sum_{n,m}\sum_{p=1}^{P_{n,m}} \mathbf{Y}_k^*\mathbf{g}_{n,m}^p(\cdot)\exp(j\psi_{n,m}^p), \tag{5.55}$$

and

$$\mathbf{h}(\cdot)^*\mathbf{h}(\cdot) = \sum_{u=0}^{U-1}\sum_{n_1,m_1,n_2,m_2}\sum_{p_1=1}^{P_{n_1,m_1}}\sum_{p_2=1}^{P_{n_2,m_2}} \mathbf{g}_{n_2,m_2}^{p_2}(\cdot)^*\mathbf{g}_{n_1,m_1}^{p_1}(\cdot)\exp\{j(\psi_{n_1,m_1}^{p_1} - \psi_{n_2,m_2}^{p_2})\}. \tag{5.56}$$

Let $\boldsymbol{\psi}_k$ be a vector containing all the phase variables $\psi_{n,m}^p$, for $1 \leq n \leq N$ and $1 \leq m \leq M$. Further, for notational simplicity, we relabel the phases as

$$\boldsymbol{\psi}_k = [\psi_{1,k}\ \psi_{2,k}\ \ldots\ \psi_{\Omega,k}]'. \tag{5.57}$$

This allows us to express (5.55) and (5.56) in simpler forms:

$$\mathbf{Y}_k^*\mathbf{h}(\cdot) = \sum_{m=1}^{\Omega}\mathbf{u}_m(\cdot)^*\exp(j\psi_{m,k}), \tag{5.58}$$

$$\mathbf{h}(\cdot)^*\mathbf{h}(\cdot) = \sum_{m_1=1}^{\Omega}\sum_{m_2=1}^{\Omega}\mathbf{u}_{m_2}^*(\cdot)\mathbf{u}_{m_1}(\cdot)\exp\{j(\psi_{m_1,k} - \psi_{m_2},k)\}, \tag{5.59}$$

$$= 2\mathrm{Re}\left\{\sum_{m_1=1}^{\Omega-1}\sum_{m_2=m_1+1}^{\Omega-1}\mathbf{u}_{m_2}^*(\cdot)\mathbf{u}_{m_1}(\cdot)\exp\{j(\psi_{m_1,k}-\psi_{m_2,k})\}\right\}$$
$$+\sum_{m=1}^{\Omega}\mathbf{u}_m^*(\cdot)\mathbf{u}_m(\cdot), \tag{5.60}$$

where the functions $\mathbf{u}_m(\cdot)$ for $m \leq \Omega$ are implicitly defined through (5.55), (5.56) and the re-labeling of (5.57). Substitution of (5.58) and (5.60) back into (5.54) produces the following proportionality:

$$p(\boldsymbol{\psi}_k|\mathbf{y}^k,\mathbf{x}_k,\mathbf{z}_k,i) \propto \exp\Big\{\sum_{n=1}^{\Omega}\kappa_{n,n}\cos(\psi_{n,k}-\phi_{n,n})+$$
$$\sum_{n_1=1}^{\Omega-1}\sum_{n_2=n_1+1}^{\Omega}\kappa_{n_1,n_2}\cos(\psi_{n_1,k}-\psi_{n_2,k}+\phi_{n_1,n_2})\Big\}, \tag{5.61}$$

where κ_{n_1,n_2} and ϕ_{n_1,n_2} for $1 \leq n_1 \leq \Omega-1$ and $n_1 \leq n_2 \leq \Omega$ are:

$$\kappa_{n_1,n_2} = \begin{cases} \dfrac{||\mathbf{u}_{n_1}(\cdot)||_2^2}{\sigma^2} & \text{if } n_1 = n_2, \\ -\dfrac{||\mathbf{u}_{n_2}(\cdot)^*\mathbf{u}_{n_1}(\cdot)||_2^2}{\sigma^2} & \text{otherwise,} \end{cases} \tag{5.62}$$

and

$$\psi_{n_1,n_2} = \begin{cases} \arg\{\mathbf{u}(n_1)(\cdot)\} & \text{if } n_1 = n_2, \\ \arg\{\mathbf{u}(n_2)^*(\cdot)\mathbf{u}_{n_1}(\cdot)\} & \text{otherwise.} \end{cases} \tag{5.63}$$

The distribution given in (5.61) is the GVM distribution introduced in Chapter 4 and in [Karunaratne 2012a].

CHAPTER 6

Multipath radar tracking with large uncertainty

Summary

We have relaxed some of the common assumptions and have built a filter to track a target moving in a partially known multipath environment in Chapter 5. In this chapter, we formulate the target tracking problem for radar in a multipath environment where significant uncertainty on the locations of the multipath causing obstacles (walls) is present. This significant uncertainty arises from a relaxation of the assumption that the wall locations in the environment are known, either partially as in Chapter 5 or completely. We propose a statistical filter and a data association method based on importance sampling to address these challenges. Recently introduced techniques in statistical signal processing such as Set JPDAF and progressive correction are incorporated into the proposed filter. Finally, simulation results are presented to investigate the performance of the filter under challenging tracking scenarios.

6.1 Introduction

Many interesting approaches have been proposed for the problem of tracking in a multipath environment where the geometry of the environment is known [Chakraborty 2010, Chakraborty 2011, Krolik 2006, Hayvaci 2012a, Hayvaci 2012b, Barbosa 2008, Pulford 1998]. A more detailed review of some of this work is given in Chapter 5. As a summary, most of these papers consider an environment where the number of obstacles is limited or configured in a special way, e.g., parallel walls. Another common feature of most of this work is the use of radar detections as the measurements for the tracker, but it was shown in [Morelande 2007] that better performance bounds could be achieved if raw sensor data were used as measurements for the tracker instead of the radar detections. The work done by us on a partially known multipath environment using raw measurements instead

of detections to track a target in an urban terrain where walls are not necessarily parallel to each other can be found in [Karunaratne 2012b, Karunaratne 2013]. These were also explained in more detail in Chapters 3 and 5. A natural extension to these previous work was to investigate the problem of tracking in a multipath environment where a large uncertainty about the environment prevails. We describe our work on that objective in this chapter; but first, we briefly review some relevant literature.

For investigating the problem of tracking in an unknown environment, we have chosen to use the detection based measurement model rather than the raw sensor data. This is done to avoid the expense of dealing with a large measurement vector and to make it easier to focus on the difficulties which arise from gross uncertainties in the environment. The use of detections in a multipath environment creates another problem known as *data association*. Basically, the problem is about resolving the ambiguity of the origin of each measurement. The data association problem has been extensively studied in the past. One of the popular suboptimal solutions to the data association problem is provided by the Probabilistic Data Association Filter (PDAF) [Bar-Shalom 2009]. The Joint Probabilistic Data Association Filter (JPDAF) is an extension to the PDAF for cases involving more than one target [Bar-Shalom 2009]. An application of probabilistic data association techniques for target tracking in clutter is presented in [Kirubarajan 1998], which focuses on the problem of discrimination of target returns from clutter returns, rather than that of exploiting multipath information. The use of a data association filter while exploiting multipath is considered in [Chakraborty 2011]. Therein the authors apply a modified Interacting Multiple Model PDAF (IMM-PDAF) [Houles 1989] to separate clutter while exploiting the information content in multipath. However, they assume optimal associations for the multipath measurements involving the target (optimal in the sense that the sources of the non-clutter measurements are assumed to be known). Another example of data association for a multipath model in "over the horizon radar" is presented in [Pulford 1998]. In that model, the multipath reflections are caused by two ionospheric layers located at known heights; hence, the multipath environment is considered to be known.

Our task of tracking in an uncertain multipath environment closely resembles the well established (Simultaneous Localisation and Mapping) SLAM problem appearing in the domain of robotics and control research. A review of SLAM developments is found in [Durrant-Whyte 2006, Bailey 2006]. Conceptually, from a high level perspective, both problems try to estimate the target state as well as the environment. The difference between the SLAM and tracking in an uncertain

multipath environment comes from the measurement model used; much of SLAM uses LiDAR technology where multipath is out of context. For the tracking problem considered in this paper, the relationship between the measurements and the environment is more indirect than in conventional SLAM setups. In SLAM one usually obtains direct position measurements of scatterers which compose the environment. This creates a direct relationship between the measurements and the environment. In contrast, in our problem, the environment is observed indirectly through the measurements which simultaneously contain information about the target and the environment. This gives rise to a difficult problem in non-linear estimation and data association.

Some recent work on radar tracking in an unknown multipath environment is found in [Li 2011]. The problem considered there is to track a moving radio frequency emitter. The authors do not impose any restrictions on the geometry of the walls. The problem formulation also allows for false alarms and missed detections. The data associations are performed using the well known Viterbi algorithm. The associations are chosen to maximise the log-likelihood. Note that evaluation of the log likelihood function for each possible association requires the knowledge of the target state, which is not available to us. The authors take a suboptimal approach of using the predicted state based on the previous measurements and the previous associations. The disadvantage of this approximation is that, if the prediction covariance is broad, the chances of the true target state being significantly different from the prediction are high. This has the undesired effect of using an inaccurate approximation of the cost function. An alternative approach would be to take the expectation of the likelihood with respect to the prior distribution [Bar-Shalom 2009], thus eliminating the dependency of the target state on the cost function.

In this chapter, we present our study on tracking a target in a highly uncertain multipath environment where the radar system (both the transmitter and the receiver) is not collocated with the target. The brute force calculation of all the association probabilities is computationally very expensive. In some special cases efficient methods exist to solve the data association problem involving very large number of possible hypotheses [Horridge 2006]. It is not immediately obvious to us how to extend [Horridge 2006] to the problem considered in this chapter. So we propose an efficient importance sampling procedure to draw samples from association events. The posterior moments conditioned on the sampled association events are also calculated using a Monte Carlo approach. Finally, the state estimate is obtained by averaging the conditional posteriors over association probabilities. Special care is needed in fusing the mixture of conditional posterior distributions.

This is due to the implicit ordering in a vector representation, which results in unnecessary confusion between the labels assigned to the walls. We used the Set JPDAF (SJPDAF) method suggested in [Svensson 2009] to address the issue which yields more accurate Gaussian approximations. Another approach to address this problem could be found in [Blom 2011]. Because it is desired to focus on the problems posed by an unknown environment, we analyse the case where the environment contained two walls and assume in this work perfect detections with no false alarms. The methodology developed here can be extended without too much difficulty to the case of an unknown number of walls along with missed detections and clutter. This extension is discussed but not implemented

The rest of the chapter is organised as follows. We first introduce the process and measurement models used in Section 6.2. In addition, the estimation problem is formulated in that section. The estimation method is then presented in Section 6.3. We do so by first setting up a basic filter and improving it gradually by incorporating the concept of progressive correction, SJPDAF algorithm, and a marginalising procedure using the conditional Gaussian formula. Simulation examples along with a discussion are presented in Section 6.4 followed by concluding remarks.

Some of the common notation and acronyms used in this chapter are summarised in Table 6.1. Any notation or acronym not appearing in Table 6.1 is defined at its first occurrence.

Table 6.1: Summary of common notation and Acronyms.

Notation/Acronym	Description
$\mathbf{x}^k$	$[\mathbf{x}_1' \ \mathbf{x}_2' \ \dots \mathbf{x}_k']'$ where $k \in \mathbb{Z}^+$
$1:k$	$[1 \ 2 \ \dots \ k]'$ $k \in \mathbb{Z}^+$
$\mathbf{x}_\mathbf{v}$	$[x_{v_1} \ x_{v_2} \dots \ x_{v_N}]$ (x_n is the n^{th} element of $\mathbf{x}$. Here, the index vector $\mathbf{v}$ consist of N integer indices)
$\{\mathbf{x}^{(j)}\}_{j=1}^J$	$\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(J)}\}$ (A set of samples)
$\mathcal{S}_n$	Set of all permutations of integers 1 to n where $n \in \mathbb{Z}^+$
MSE	Mean Squared Error
BMSE	Bayesian MSE
RMSE	Root MSE
MCMC	Markov Chain Monte Carlo
PDAF	Probabilistic Data Association Filter
JPDAF	Joint PDAF
SJPDAF	Set JPDAF

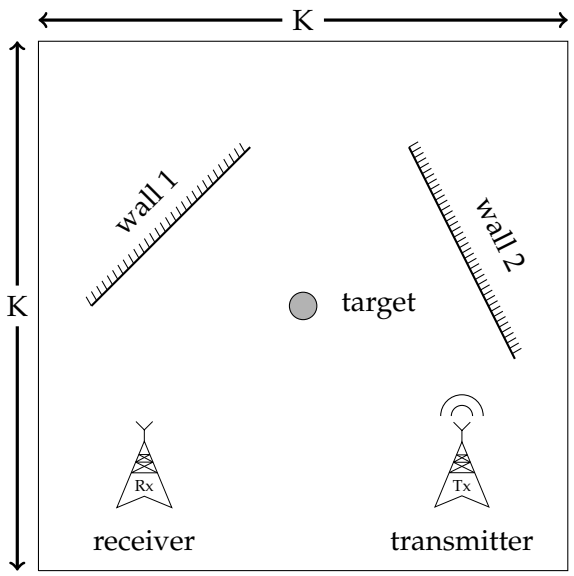


Figure 6.1: An illustration of a $K \times K$ surveillance area.

6.2 Modelling and notation

Consider a $K \times K$ surveillance area as shown in Figure 6.1. Two walls are located within this surveillance area. A point target is moving across the surveillance region and the kinematic state of it at discrete time k is denoted by $\mathbf{T}_k$. $\mathbf{T}_k$ consists of position and velocity information in the Cartesian plane; that is,

$$\mathbf{T}_k = [x_k \ \dot{x}_k \ y_k \ \dot{y}_k]',$$

where $(x_k, y_k) \in \mathbb{R}^2$ is the position of the target and the dot notation denotes differentiation with respect to time. Without loss of generality, we consider the origin $(0, 0)$ to be the bottom left most point of the surveillance region. A radar transmitter and a receiver are located at known positions within the surveillance region to collect measurements for use in localizing the target.

We use a straight line to represent a wall parametrised as follows:

$$x \sin\alpha - y \cos\alpha + \beta = 0.$$

Thus, the i^{th} wall where $i \in \{1, 2\}$ is represented by the vector

$$\mathbf{w}_i = [\alpha_i \ \beta_i]'. \tag{6.1}$$

Let $\mathbf{W}$ be defined as:

$$\mathbf{W} = [\mathbf{w}_1' \ \mathbf{w}_2']'.$$

Note that one of the limitations of our representation is that we assume the wall to exist throughout the intersection between the straight line represented by the wall parameters and the surveillance area. We believe that the extension to the general case involving line segments could be developed based on the foundation laid out in this chapter.

6.2.1 State dynamics

The kinematic state of the target is assumed to transition from time $k-1$ to k according to the following stochastic model:

$$\mathbf{T}_k = \mathbf{F}\mathbf{T}_{k-1} + \mathbf{v}_k, \tag{6.2}$$

where

$$\mathbf{F} = \mathbf{I}_2 \otimes \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \tag{6.3}$$

$$\mathbf{v}_k \sim \mathcal{N}(\cdot; \mathbf{0}, \mathbf{Q}), \tag{6.4}$$

with the symbols $\otimes$, $\mathbf{I}$, and $\mathbf{0}$ denoting the Kronecker operator, Identity matrix, and Zero matrix of appropriate dimensions respectively. The notation $\mathcal{N}(\cdot;, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

The vector of wall parameters ($\mathbf{W}$) is modelled as a vector of static random parameters. That is, the wall parameters are not time dependent.

$\mathbf{W}$ is assumed to be drawn from a prior distribution $p_w(\cdot)$. A highly uncertain prior distribution for $\mathbf{W}$ will be described in Section 6.2.4.

The state vector $\mathbf{x}_k$ at time k consists of both the target dynamics and wall parameters; that is,

$$\mathbf{x}_k = [\mathbf{T}_k' \ \mathbf{W}']'. \tag{6.5}$$

6.2.2 Measurements

The measurements at time k consist of range (r) and Angle of Arrival (AOA) (θ) of each multipath and the direct path. We define a multipath as a path from the transmitter to the radar sensor which has been in contact with the target as well as at least with one wall. Any path that has hit a wall more than twice is ignored in the model under the assumption that such paths are subject to severe attenua-

tion in signal power. We assume that the walls act as specular reflectors and that necessary processing has been carried out to reject clutter measurements.

The measurement vector at time k for N paths is given by

$$\mathbf{y}_k = [\mathbf{r}'_{1,k}\ \mathbf{r}'_{2,k} \dots\ \mathbf{r}'_{N,k}]' + \mathbf{u}_k, \tag{6.6}$$

$$= [\mathbf{y}'_{1,k}\ \mathbf{y}'_{2,k} \dots\ \mathbf{y}'_{N,k}]', \tag{6.7}$$

where

$\mathbf{r}_{i,k} = [r_{i,k}\ \theta_{i,k}]'$ denotes a vector containing (noiseless) range and AOA of the i^{th} path at time k,

$\mathbf{u}_k \sim \mathcal{N}(\cdot; \mathbf{0}, \mathbf{R})$ is the measurement noise,

$\mathbf{y}_{i,k}$ is the vector containing the range and AOA of the i^{th} multipath at time k.

The noise covariance matrix $\mathbf{R}$ is

$$\mathbf{R} = \mathbf{I}_N \otimes \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}.$$

We assume that all the possible paths arising from reflections from the two walls exist at time k.

6.2.3 Path configurations and the measurement association vector

A path can be decomposed into two segments: the forward path from the transmitter to the target and the reverse path from the target to the sensor. Assume that at most one wall is hit in each segment (because the strength of the reflections diminishes over multiple reflections, we ignore more than one reflection in each segment). Thus a path is uniquely identified by two numbers given by the labels of the walls that were hit in each segment respectively. If a wall is not hit on a particular segment, then that segment would be identified by "0". We denote this pair of non-negative integers identifying a path as a *path configuration*. As an example a path configuration of (2,1) is a multipath that has travelled in the following sequence; transmitter→wall 2→target→wall 1→sensor. A direct path is identified by the path configuration (0,0).

Note that for a 2 wall environment, 9 distinct path configurations exist. Let the 9×2 matrix $\mathbf{C}$ denote the collection of all the path configurations, where the i^{th} row $\mathbf{C}_i$ contains the ordered pair of numbers identifying the i^{th} path configuration; that

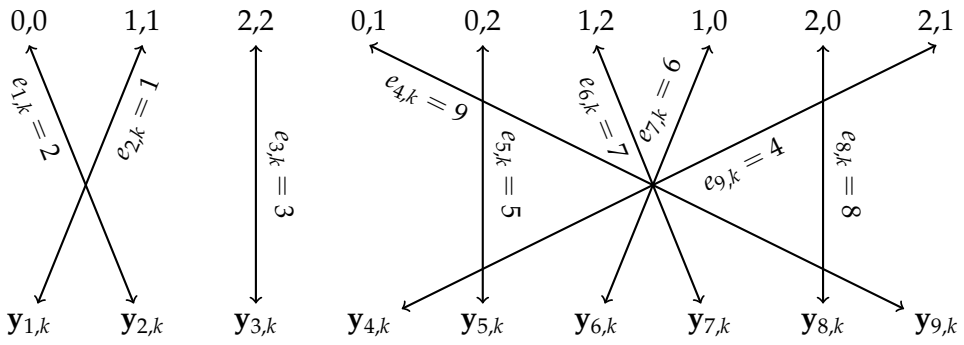


Figure 6.2: Associating measurements with path configurations for $\mathbf{e}_k = [2\ 1\ 3\ 9\ 5\ 7\ 6\ 8\ 4]'$.

is,

$$\mathbf{C} = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 2 & 2 \\ 0 & 1 \\ 0 & 2 \\ 1 & 2 \\ 1 & 0 \\ 2 & 0 \\ 2 & 1 \end{bmatrix}. \tag{6.8}$$

We name $\mathbf{C}$ the *path configuration matrix*.

By itself the measurement vector $\mathbf{y}_k$ does not provide information about how the measurements are associated with path configurations. The association vector $\mathbf{e}_k \in \mathcal{S}_9$ provides the mapping between measurements and path configurations. Then, the i^{th} element of $\mathbf{e}_k$, denoted by $e_{i,k}$ maps the path configuration represented by the i^{th} row of $\mathbf{C}$ to the $e_{i,k}^{th}$ measurement at time k. As an illustration we use an example in Figure 6.2 to further clarify how $\mathbf{y}_k$ is associated with path configurations $\mathbf{e}_k$. Suppose $\mathbf{e}_k = [2\ 1\ 3\ 9\ 5\ 7\ 6\ 8\ 4]'$. Then the measurement associations are shown in Figure 6.2.

Figure 6.2 also illustrates an important difference between conventional data association problems and the one considered here. Note that information about a particular wall is contained in more than one measurement. As an example, for the particular $\mathbf{e}_k$ used to draw Figure 6.2, the information about the wall labelled "1" is contained in measurements $\mathbf{y}_{1,k}$, $\mathbf{y}_{9,k}$, $\mathbf{y}_{7,k}$, $\mathbf{y}_{6,k}$, and $\mathbf{y}_{4,k}$. In contrast, only one measurement is associated with a single target in conventional data association problem formulations.

6.2.4 A highly uncertain prior for the wall parameters

Recall that we used the parametrization $x \sin\alpha - y\cos\alpha + \beta = 0$ to represent a wall. A continuum of line equations could be obtained by varying the slope parameter α and the intercept parameter β. We are interested only in lines that pass through the $K \times K$ surveillance area. Suppose we do not assume any knowledge about the wall location; then, a loose implication is that all the lines which pass through the surveillance area are equally likely. The joint distribution formed through the following marginal distribution for α and the conditional distribution for $\beta|\alpha$ is a reasonable prior distribution which captures such large uncertainty in the environment:

$$p(\alpha) = \mathcal{U}_{[0,\pi)}, \tag{6.9}$$

$$p(\beta|\alpha) = \mathcal{U}_{R_\beta(\alpha)}, \tag{6.10}$$

where

$$\mathcal{R}_\beta(\alpha) = \begin{cases} \{\beta; -K\sin(\alpha) \leq \beta \leq K\cos(\alpha)\} \text{ if } \alpha < \pi/2, \\ \{\beta; K(\cos(\alpha) - \sin(\alpha)) \leq \beta \leq 0\} \text{ Otherwise.} \end{cases} \tag{6.11}$$

It is easy to see that a uniform distribution in $[0, \pi)$ for the slope parameter α results in all the line slopes being *equally likely*. The conditional distribution $p(\beta|\alpha)$ is obtained by solving for the range of β values (with α fixed) that represent the lines passing through the surveillance area, and assigning a uniform distribution for β over that region.

From equations (6.9) and (6.10), it is possible to work out the mean and covariance of the joint distribution $p(\alpha, \beta)$. They are given by

$$\text{Mean } (\alpha, \beta) = \begin{bmatrix} \frac{\pi}{2} \\ -\frac{K}{\pi} \end{bmatrix}, \tag{6.12}$$

and

$$\text{Covariance } (\alpha, \beta) = \begin{bmatrix} \frac{\pi^2}{12} & -\frac{K}{\pi} \\ -\frac{K}{\pi} & \frac{K^2(2\pi^2 - 6 + \pi)}{6\pi^2} \end{bmatrix}. \tag{6.13}$$

6.2.5 Estimation problem

Let $\mathbf{y}^k = [\mathbf{y}_1' \ \mathbf{y}_2' \ \ldots \ \mathbf{y}_k']'$ denote the vector containing measurements up to time k. We are interested in estimating the state vector $\mathbf{x}_k$ upon observing $\mathbf{y}^k$.

If the performance of the estimator is measured in terms of the BMSE, then the

optimal estimator is the posterior mean of $\mathbf{x}_k$; that is, $\hat{\mathbf{x}}_k = \mathbb{E}(\mathbf{x}_k|\mathbf{y}^k)$, where the operator $\mathbb{E}(\cdot)$ denotes expectation [Kay 1993].

6.3 Theory/Methodology

Consider the posterior distribution of the target state $\mathbf{x}_k$. We write this as

$$p(\mathbf{x}_k|\mathbf{y}^k) = \sum_{\mathbf{e}_k} p(\mathbf{x}_k|\mathbf{y}^k, \mathbf{e}_k) p(\mathbf{e}_k|\mathbf{y}^k). \tag{6.14}$$

The first term in the summand is a conditional posterior of the target state (conditioned on the association vector), while the second term $p(\mathbf{e}_k|\mathbf{y}^k)$ is the posterior distribution of the association vector $\mathbf{e}_k$. We now concentrate on the latter; that is, $p(\mathbf{e}_k|\mathbf{y}^k)$:

$$\begin{aligned}
p(\mathbf{e}_k|\mathbf{y}^k) &\propto p(\mathbf{y}_k|\mathbf{e}_k, \mathbf{y}^{k-1}) p(\mathbf{e}_k|\mathbf{y}^{k-1}), && (6.15)\\
&\propto p(\mathbf{y}_{e_{1,k},k}|\mathbf{e}_k, \mathbf{y}^{k-1}) p(\mathbf{y}_{e_{2,k},k}|\mathbf{y}_{e_{1,k},k}, \mathbf{e}_k, \mathbf{y}^{k-1}) \cdots p(\mathbf{y}_{e_{N,k},k}|\mathbf{y}_{\mathbf{e}_{1:N-1,k},k}, \mathbf{e}_k, \mathbf{y}^{k-1}), && (6.16)\\
&= p(\mathbf{y}_{e_{1,k},k}|e_{1,k}, \mathbf{y}^{k-1}) p(\mathbf{y}_{e_{2,k},k}|\mathbf{e}_{1:2,k}, \mathbf{y}_{e_{1,k},k}, \mathbf{y}^{k-1}) \cdots \\
&\qquad p(\mathbf{y}_{e_{N,k},k}|\mathbf{e}_k, \mathbf{y}_{\mathbf{e}_{1:N-1,k},k}, \mathbf{y}^{k-1}), && (6.17)\\
&= \prod_{n=1}^{N} p(\mathbf{y}_{e_{n,k},k}|\mathbf{e}_{1:n,k}, \mathbf{y}_{\mathbf{e}_{1:n-1,k},k}, \mathbf{y}^{k-1}), && (6.18)
\end{aligned}$$

where $N = 9$ is the number of path configurations. Equation (6.16) follows from Bayes theorem applied to $p(\mathbf{y}_k|\mathbf{e}_k, \mathbf{y}^{k-1})$ along with the assumption that $p(\mathbf{e}_k|\mathbf{y}^{k-1}) = p(\mathbf{e}_k)$ is a uniform prior distribution.

Before proceeding further, we simplify the notation by introducing $p_n(\mathbf{y}_{e_{n,k},k}|\mathbf{y}^{k-1})$:

$$p_n(\mathbf{y}_{e_{n,k},k}|\mathbf{y}^{k-1}) = p(\mathbf{y}_{e_{n,k},k}|\mathbf{e}_{1:n,k}, \mathbf{y}_{\mathbf{e}_{1:n-1,k},k}, \mathbf{y}^{k-1}). \tag{6.19}$$

Here, we let the conditioning on $\mathbf{e}_{1:n,k}$ and $\mathbf{y}_{\mathbf{e}_{1:n-1,k},k}$ be implicitly implied by the subscript n of p_n, unless we explicitly include those as conditioning variables. Additionally, from this point onwards we drop the notation k in the subscript of $e_{i,k}$ in favour of brevity.

Using (6.18) and the shortened notation (6.19), we write $p(\mathbf{e}_k|\mathbf{y}^k)$ as follows:

$$p(\mathbf{e}_k|\mathbf{y}^k) \propto \prod_{n=1}^{N} p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}). \tag{6.20}$$

The posterior distribution for the target state is now expressed in terms of $p_n(\cdot)$

by substituting (6.20) into (6.14) to give

$$p(\mathbf{x}_k|\mathbf{y}^k) \propto \sum_{\mathbf{e}_k} \prod_{n=1}^{N} p(\mathbf{x}_k|\mathbf{y}^k, \mathbf{e}_k) p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}). \tag{6.21}$$

An optimal implementation of the filter using (6.21) is not possible because there are too many association hypotheses $\mathbf{e}_k$, as well as intractable integrals. Procedures for addressing these issues are given in the following sections.

6.3.1 A basic filter using Monte Carlo approximations

We now consider one of the intractable integrals that prevents us from implementing the optimal filter based on the exact posterior given in (6.21); namely the $p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1})$ terms appearing in (6.20). The integral that needs to be evaluated is given below:

$$p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}) = \int p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1}) p(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) d\mathbf{x}_k. \tag{6.22}$$

Though a closed form solution is not available for (6.22), we could approximate it by Monte Carlo using samples from $p(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$. Unfortunately, the distribution $p(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ is also not known in closed form, but for the sake of argument, assume that we have a reasonable approximation for it denoted by $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$. In particular, let $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ be a particle approximation of $p(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k})$ consisting of a finite number of weighted samples; then,

$$p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}) \approx \int p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1}) \hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) d\mathbf{x}_k, \tag{6.23}$$

$$\approx \frac{1}{M} \sum_{m=1}^{M} p_n(\mathbf{y}_{e_n,k}|\boldsymbol{\chi}_k^{(n,m)}, \mathbf{y}^{k-1}), \tag{6.24}$$

where $\boldsymbol{\chi}_k^{(n,m)} \sim \hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$.

The approximate distribution $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ that was used to draw samples in (6.24) can be updated to $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n},k}, \mathbf{y}^{k-1})$ as shown by (6.25), which satisfies the recursive dependence required to evaluate $p_{n+1}(\cdot)$:

$$\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n},k}, \mathbf{y}^{k-1}) \propto \sum_{m=1}^{M} p_n(\mathbf{y}_{e_n,k}|\boldsymbol{\chi}_k^{(n,m)}, \mathbf{y}^{k-1}) \delta(\mathbf{x}_k - \boldsymbol{\chi}_k^{(n,m)}). \tag{6.25}$$

Two common problems associated with particle based recursive procedures such as the process described by (6.24) and (6.25) are known as *particle degeneracy*

and *particle impoverishment* [Arulampalam 2002]. These problems in the context of particle filtering were discussed in Section 2.1.4. Degeneracy and impoverishment problems could be counteracted to a certain extent by introducing a simple resampling procedure followed by regularization as described in [Arulampalam 2002, p. 182]. These procedures in the context of the current filtering problem are briefly explained as follow. Suppose that the sample $\boldsymbol{\chi}_k^{(n,m)}$ is assigned a weight of $p_n(\mathbf{y}_{e_n,k}|\boldsymbol{\chi}_k^{(n,m)}, \mathbf{y}^{k-1})$, for $m = 1, 2, \ldots, M$. Let $\mathbf{E}_{k,n}$ denote the sample covariance of these weighted samples. Further, let $t_1, t_2, \ldots, t_M$ be M index variables drawn according to

$$\text{Prob}(t_m = i) \propto p_n(\mathbf{y}_{e_n,k}|\boldsymbol{\chi}_k^{(n,i)}, \mathbf{y}^{k-1}) \text{ for } m = 1, 2, \ldots, M. \tag{6.26}$$

Let $\tilde{\mathbf{x}}_k^{(n,m)} = \boldsymbol{\chi}_k^{(n,t_m)}$ for all $m = 1, 2, \ldots, M$. Both resampling and kernel regularisation are concurrently achieved by redrawing $\mathbf{x}_k^{(n,m)}$ for $m = 1, 2, \ldots, M$ according to:

$$\mathbf{x}_k^{(n,m)} \sim \mathcal{N}(\cdot, \tilde{\mathbf{x}}_k^{(n,m)}, h^2\mathbf{E}_{k,n}), \tag{6.27}$$

where h is a smoothing parameter known as the bandwidth [Silverman 1986].

The samples obtained in (6.27) are now used to approximate $p(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n},k}, \mathbf{y}^{k-1})$ as

$$\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n},k}, \mathbf{y}^{k-1}) \propto \sum_{m=1}^{M} \delta(\mathbf{x}_k - \mathbf{x}_k^{(n,m)}). \tag{6.28}$$

Compared to (6.25), the approximation (6.28) is less prone to particle degeneracy and impoverishment problems.

We use (6.24) in (6.20) to approximate the posterior distribution for the association vector by

$$p(\mathbf{e}_k|\mathbf{y}^k) \propto \prod_{n=1}^{N} \sum_{m=1}^{M} p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1}). \tag{6.29}$$

We wish to obtain samples from (6.29). Note that $p(\mathbf{e}_k|\mathbf{y}^k)$ is a discrete distribution, and thus directly sampling from it requires that (6.29) be calculated for all the possible association hypotheses. This is extremely expensive because of the enormous number of possible hypotheses. For example, with two walls there are $9! = 362880$ hypotheses. Hence we use the following importance distribution $q(\mathbf{e}_k|\mathbf{y}_k)$ to draw association vector samples:

$$q(\mathbf{e}_k|\mathbf{y}^k) = \prod_{n=1}^{N} \left(\frac{\sum_{m=1}^{M} p_n(\mathbf{y}_{r,k}|\mathbf{e}_{1:n-1}, e_n = r, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1})}{\sum_{v\in\mathcal{A}_n}\sum_{m=1}^{M} p_n(\mathbf{y}_{v,k}|\mathbf{e}_{1:n-1}, e_n = v, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1})} \right) \quad \text{where } r \in \mathcal{A}_n, \tag{6.30}$$

with $\mathcal{A}_n = \{1, 2, \ldots, N\} \backslash \{e_1, e_2, \ldots, e_{n-1}\}$.

Drawing a sample from $q(\mathbf{e}_k|\mathbf{y}^k)$ is a sequential process where one element of

the association vector is drawn at each step. In contrast, if we were to draw a sample from $p(\mathbf{e}_k|\mathbf{y}^k)$, the posterior association probabilities for all $\mathbf{e}_k \in \mathcal{S}_9$ need to be calculated. This is a good example of the usefulness of importance sampling, where we are able to greatly reduce the amount of computational expense by exploring only the significant areas of the sampling distribution.

In the context of importance sampling, the samples obtained from $q(\mathbf{e}_k|\mathbf{y}^k)$ are assigned a weight $\tilde{\phi}$ given by the ratio of $p(\mathbf{e}_k|\mathbf{y}^k)$ to $q(\mathbf{e}_k|\mathbf{y}^k)$:

$$\tilde{\phi} = p(\mathbf{e}_k|\mathbf{y}^k)/q(\mathbf{e}_k|\mathbf{y}^k), \tag{6.31}$$

$$\propto \prod_{n=1}^{N} \sum_{r\in\mathcal{A}_n} \sum_{m=1}^{M} p_n(\mathbf{y}_{r,k}|\mathbf{e}_{1:n-1}, e_n = r, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1}). \tag{6.32}$$

We obtain the posterior distribution for the target state as follows:

$$p(\mathbf{x}_k|\mathbf{y}^k) = \int p(\mathbf{x}_k|\mathbf{y}^k, \mathbf{e}_k) p(\mathbf{e}_k|\mathbf{y}^k) d\mathbf{e}_k, \tag{6.33}$$

$$\propto \sum_{j=1}^{J} \tilde{\phi}_j \hat{p}(\mathbf{x}_k|\mathbf{e}_k^{(j)}, \mathbf{y}^k), \tag{6.34}$$

where J is the sample size, and

$$\mathbf{e}_k^{(j)} \sim q(\mathbf{e}_k|\mathbf{y}^k), \tag{6.35}$$

$$\tilde{\phi}_j \propto \prod_{n=1}^{N} \sum_{r\in\mathcal{A}_{n,j}} \sum_{m=1}^{M} p_n(\mathbf{y}_{r,k}|\mathbf{e}_{1:n-1}^{(j)}, e_n^{(j)} = r, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1}), \tag{6.36}$$

with $\mathcal{A}_{n,j}$ given by

$$\mathcal{A}_{n,j} = \{1, 2, \ldots, N\} \backslash \{e_1^{(j)}, e_2^{(j)}, \ldots, e_{n-1}^{(j)}\}. \tag{6.37}$$

It should be noted that the statistics of the target samples obtained during the process of drawing associations $\mathbf{e}_k$ (as presented in (6.25)-(6.28)) are used to approximate $\hat{p}(\mathbf{x}_k|\mathbf{e}_k^{(j)}, \mathbf{y}^k)$ by a Gaussian.

The recursive calculation procedure involved in obtaining estimates at time k is summarised in algorithmic form in Algorithm 6.1, which is included in Appendix 6.B. Once again it should be noted that, prior to processing measurements at time $k+1$, we summarise the posterior distribution at time k by a Gaussian distribution using the sample moments.

6.3.2 Progressive correction

Recall (6.23) and (6.24), where we considered the probability distribution of the e_n^{th} measurement at time k, conditioned on the first n elements of the association vector $(\mathbf{e}_{1:n})$ as well as the $n-1$ measurements already associated $(\mathbf{y}_{\mathbf{e}_{1:n-1},k})$ and the past measurements $\mathbf{y}^{k-1}$:

$$p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}) \approx \int p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})\hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1},\mathbf{y}_{\mathbf{e}_{1:n-1},k},\mathbf{y}^{k-1})d\mathbf{x}_k, \tag{6.38}$$

$$\approx \frac{1}{M}\sum_{m=1}^{M} p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k^{(n,m)},\mathbf{y}^{k-1}), \tag{6.39}$$

The accuracy of the approximation (6.39) is important because of its direct involvement in the calculation of the sample weight of the associations (see (6.32)). An alternative way of approximating (6.38) is through the concept of progressive correction. We introduced the idea of progressive correction in Chapter 5 and used it in the proposed MCMC particle filter. In Chapter 5, the progressive correction was performed by partitioning the measurement vector and sequentially performing the Bayesian correction by considering one partition at a time. Another approach to progressive correction is to express the likelihood function as a series of products and performing the Bayesian correction sequentially over each term in the series as in [Oudjane 2000]. In this chapter we use the latter approach as follows: Let B real numbers in $[0,1]$ such that their sum is equal to 1, be denoted by $\gamma_1,\gamma_2,\ldots,\gamma_B$. Then we write $p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1})$ as:

$$p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}) \approx \int p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})\hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1},\mathbf{y}_{\mathbf{e}_{1:n-1},k},\mathbf{y}^{k-1})d\mathbf{x}_k, \tag{6.40}$$

$$= \int \left\{\prod_{b=1}^{B}[p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})]^{\gamma_b}\right\} \hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1},\mathbf{y}_{\mathbf{e}_{1:n-1},k},\mathbf{y}^{k-1})d\mathbf{x}_k, \tag{6.41}$$

$$= \int \left\{\prod_{b=2}^{B}[p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})]^{\gamma_b}\right\} \left\{\left[p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})\right]^{\gamma_1} \hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1},\mathbf{y}_{\mathbf{e}_{1:n-1},k},\mathbf{y}^{k-1})\right\} d\mathbf{x}_k, \tag{6.42}$$

$$= \omega_1 \int \left\{\prod_{b=2}^{B}[p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k,\mathbf{y}^{k-1})]^{\gamma_b}\right\} \hat{p}_{x,1}(\mathbf{x}_k|\mathbf{e}_{1:n-1},\mathbf{y}_{\mathbf{e}_{1:n-1},k},\mathbf{y}^{k-1})d\mathbf{x}_k, \tag{6.43}$$

where

$$\omega_1 = \int [p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1})]^{\gamma_1} \hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) d\mathbf{x}_k, \tag{6.44}$$

$$\hat{p}_{x,1}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) = \frac{p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1})]^{\gamma_1} \hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})}{\omega_1}. \tag{6.45}$$

The process is easily continued to give, recursively,

$$p_n(\mathbf{y}_{e_n,k}|\mathbf{y}^{k-1}) \approx \prod_{b=1}^{B} \omega_b, \tag{6.46}$$

where

$$\omega_b = \int [p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1})]^{\gamma_b} \hat{p}_{x,b-1}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) d\mathbf{x}_k, \text{ for } b = 2, 3, \ldots, B \tag{6.47}$$

with $\hat{p}_{x,b}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k})$ related to $\hat{p}_{x,b-1}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k})$ through

$$\hat{p}_{x,b}(\mathbf{x}_k|\mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1}) = \frac{[p_n(\mathbf{y}_{e_n,k}|\mathbf{x}_k, \mathbf{y}^{k-1})]^{\gamma_b} \hat{p}_{x,b-1}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})}{\omega_b}. \tag{6.48}$$

Note that $\hat{p}_{x,B}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ is an alternative approximation to (6.25). The difference between the two approximations is that a sequence of correction steps have been carried out while obtaining $\hat{p}_{x,B}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ whereas (6.25) is obtained by a single correction step. Progressive correction is known to reduce the approximation error in the context of particle filtering [Oudjane 2000], and is particularly well suited for the specific filtering problem considered in this chapter because of the challenges arising from the large uncertainty of the environment.

The progressive correction procedure in algorithmic form to obtain $\hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{\mathbf{e}_{1:n},k}, \mathbf{y}^{k-1})$ from $\hat{p}_x(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{\mathbf{e}_{1:n-1},k}, \mathbf{y}^{k-1})$ is listed in Algorithm 6.2, which is included in the Appendix 6.B. It should be noted that after each progressive correction, the resulting distributions $\hat{p}_{x,b}(\cdot)$, for $b = 1, 2, \ldots, B$, are approximated as Gaussian distributions using the statistics of the weighted samples obtained at each step. This step of summarising the distribution using a Gaussian distribution requires special attention. First, if the underlying distribution is multi-modal, then a Gaussian approximation to that is not appropriate. Secondly, when certain elements in the state vector are not directly affected by a measurement (but, indirectly affected through the remaining elements in the state vector), then a Gaussian

approximation leads to additional loss in accuracy for the indirectly affected elements. We address these issues in Section 6.3.3 and 6.3.4.

6.3.3 The need for the Set JPDAF

A common problem with data association algorithms such as PDAF, JPDAF, and Multiple Hypothesis Tracking (MHT) filter [Blackman 2004] is the requirement of these algorithms to retain the label assigned to each target throughout the trajectory. This is a consequence of the implicit labelling when parameter vectors are concatenated into a vector. However, in some instances preserving the identity of the targets is not required. As an example in our situation, it is not important to preserve the identity of the two walls. In other words, we are not concerned about which label we assign to the walls (1 or 2) as long as we can estimate both wall parameters with reasonable accuracy. As an illustration, consider two possible associations for the first measurement $\mathbf{y}_{1,k}$ given by $e_{6,k} = 1$ and $e_{9,k} = 1$. The association given by $e_{6,k} = 1$ corresponds to the path configuration $(1, 2)$ and thus maps the measurement $\mathbf{y}_{1,k}$ to a path that hits the wall identified by label "1" during the forward path and the wall with the label "2" during the return path respectively. On the other hand, $e_{9,k} = 1$ corresponds to the path configuration $(2, 1)$ and thus hits the walls labelled "2" and "1" on the forward path and reverse path respectively. However, the interesting thing to note is that $e_{6,k} = 1$ represents the same association as $e_{9,k} = 1$ if the labels attached to the walls are switched under any one of those hypotheses. Extending the example further, the association vector samples $\mathbf{e}_k^{(1)}$ and $\mathbf{e}_k^{(2)}$ given below represent the same measurement association, if the labels assigned to the walls are switched under any one of those hypothesis:

$$\mathbf{e}_k^{(1)} = [2\ 9\ 4\ 3\ 5\ 1\ 6\ 8\ 7]', \tag{6.49}$$

$$\mathbf{e}_k^{(2)} = [2\ 4\ 9\ 5\ 3\ 7\ 8\ 6\ 1]'. \tag{6.50}$$

The consequences of this effect on the estimates are significant, as we explain next.

Consider the posterior estimates of just the slope parameters of the two walls (that is, α_1 and α_2). Assume that the true slope parameters are given by $\pi/4$ and $-\pi/6$. Recall from (6.34) that we approximated the posterior of α_1 and α_2 by a Monte Carlo approach:

$$p(\alpha_1, \alpha_2|\mathbf{y}_k) \propto\!\!\!\propto \sum_{i=1}^{J} \tilde{\phi}_i \hat{p}(\alpha_1, \alpha_2|\mathbf{e}_k^{(i)}, \mathbf{y}^k). \tag{6.51}$$

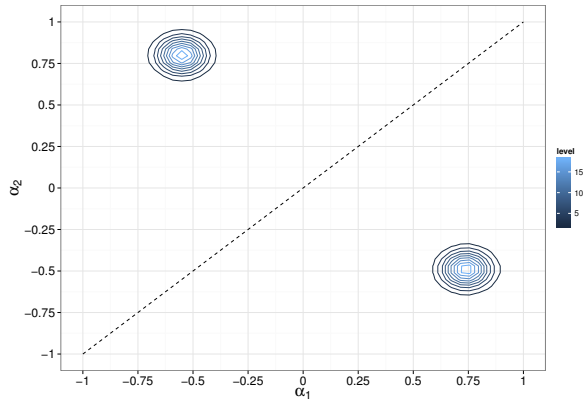
For the purpose of demonstration we restrict the number of $\mathbf{e}_k$ samples to just

the 2 samples explicitly given by (6.49) and (6.50). Further, assume that the two hypotheses are equally weighted; that is, $\tilde{\phi}_1 = \tilde{\phi}_2 \propto .5$. Finally assume that the conditional posteriors are given by the following two equations:

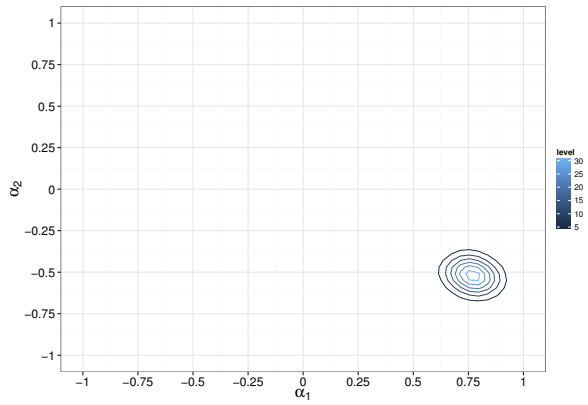
$$\hat{p}(\alpha_1, \alpha_2 | \mathbf{e}_k^{(1)}, \mathbf{y}^k) = \mathcal{N}\left(\cdot; \begin{bmatrix} 0.74 \\ -0.49 \end{bmatrix}, 0.01\mathbf{I}_2\right) \tag{6.52}$$

$$\hat{p}(\alpha_1, \alpha_2 | \mathbf{e}_k^{(2)}, \mathbf{y}^k) = \mathcal{N}\left(\cdot; \begin{bmatrix} -0.55 \\ 0.8 \end{bmatrix}, 0.01\mathbf{I}_2\right) \tag{6.53}$$

The contour plot for the posterior obtained by substituting (6.52) and (6.53) into (6.51) is shown in Figure 6.3(a). It is evident that the posterior is bimodal. Therefore, the posterior mean, given by $[0.095\ 0.155]'$, is quite distant from the true value and a Gaussian, as used in JPDAF, is a poor approximation to the posterior.



(a)



(b)

Figure 6.3: Contour plot of $p(\alpha_1, \alpha_2 | \mathbf{y}_k)$: (a) before label switching (b) after label switching.

Suppose now, we exchange the wall labels under the data association hypothesis $\mathbf{e}_k^{(2)}$. Then the conditional posteriors are approximated by:

$$\hat{p}(\alpha_1, \alpha_2 | \mathbf{e}_k^{(1)}, \mathbf{y}^k) = \mathcal{N}\left(\cdot; \begin{bmatrix} 0.74 \\ -0.49 \end{bmatrix}, 0.01\mathbf{I}_2\right) \tag{6.54}$$

$$\hat{p}(\alpha_1, \alpha_2 | \mathbf{e}_k^{(2)}, \mathbf{y}^k) = \mathcal{N}\left(\cdot; \begin{bmatrix} 0.8 \\ -0.55 \end{bmatrix}, 0.01\mathbf{I}_2\right) \tag{6.55}$$

The contour plot for the posterior distribution after label switching is given in Figure 6.3(b). Note that the posterior approximation after label switching resembles a bivariate Gaussian distribution and the posterior mean $[0.77 \ -0.52]'$ is much closer to the true parameters. Of course we could have alternatively exchanged the labels under $\mathbf{e}_k^{(1)}$ instead of $\mathbf{e}_k^{(2)}$ and obtained $[-0.52 \ 0.77]'$ as the posterior mean. The mean of $[-0.52 \ 0.77]'$ would be a poor estimate if we are concerned about the MSE. However, if we are concerned only about knowing the wall positions and not so much about the identity of the walls, either estimate of $[0.77 \ -0.52]'$ or $[-0.52 \ 0.77]'$ is equally preferable to the estimate obtained without the label exchange.

The above example illustrates one of the drawbacks of the traditional probabilistic data association algorithms, when preserving the identities of the targets is not essential. The posterior approximation obtained without label exchange is analogous to the approach taken by traditional probabilistic data association algorithms. In conventional JPDAF algorithm, the main objective is to minimize the RMSE, and as such the first approach (without label switching) is better on average than the alternative. However, a more suitable measure of performance for multi-object probabilistic data association algorithms where target identity retention is not important is known as *Optimal Sub Pattern Assignment* (OPSA) [Schuhmacher 2008b, Schuhmacher 2008a]. A modification to the traditional JPDAF algorithm with the emphasis on reducing the Mean OPSA (MOPSA) is known as SJPDAF and proposed in [Svensson 2009]. The SJPDAF does permit as well as promote label switching to obtain a posterior that resembles a multivariate Gaussian distribution.

We incorporate the SJPDAF algorithm into our filter design for the reasons presented above. A complete account of the SJPDAF algorithm is beyond the scope of this thesis. However, the intuition behind the SJPDAF algorithm is as follows. The SJPDAF algorithm describes the multi-object state (in our problem, multi-objects are the two walls) as a Random Finite Set (RFS). The RFS representation provides a framework for formulating problems related to unlabelled multiple object state

estimation [Mahler 2007]. The SJPDAF algorithm exploits the fact that there are many labelled densities that give rise to the same RFS density. A set of labelled densities that gives rise to the same RFS density is said to belong to the same RFS family. Svensson *et al* prove in [Svensson 2009] that any member of the same RFS family when updated with the same likelihood function results in labelled densities that belong to the same (updated) RFS family. Thus, any member density of the same RFS family can be used in place of another, which allows replacing labelled densities with a different one in exchange of computational convenience or a gain in estimation performance. In the context of the example used here, the densities before and after label switching (that is, (6.53) and (6.55)) belong to the same RFS and either one could be used in place of the other. The SJPDAF algorithm switches between the label switched densities and (ideally) uses the ones that result in a posterior distribution close to a multivariate Gaussian.

In Algorithm 6.3 (which is included in the Appendix 6.B), we have listed the SJPDAF in algorithmic form to obtain the posterior distribution parameters when provided with the association vector samples $\mathbf{e}_k^{(j)}$, association probability ϕ_j, and parameters of the conditional distribution $\hat{p}(\mathbf{x}_k|\mathbf{e}_k^{(j)}, \mathbf{y}^k)$ for $j = 1, \ldots, J$.

6.3.4 Marginalising using conditional Gaussian formula

Observation of (6.34) makes it clear that a good approximation for the conditional posterior $p(\mathbf{x}_k|\mathbf{e}_k, \mathbf{y}^k)$ is desired since it is directly related to the posterior approximation of $p(\mathbf{x}_k|\mathbf{y}^k)$. Note that in the preceding discussions $p(\mathbf{x}_k|\mathbf{e}_k, \mathbf{y}^k)$ was obtained sequentially, where we considered one association at a time. In other words, as explained through the equations (6.20)-(6.25), the approximation to $p(\mathbf{x}_k|\mathbf{e}_k, \mathbf{y}^k)$ is obtained through a process where N intermediate densities $p(\mathbf{x}_k|e_1, \mathbf{y}_{e_1,k}, \mathbf{y}^{k-1})$, $p(\mathbf{x}_k|\mathbf{e}_{1:2}, \mathbf{y}_{\mathbf{e}_{1:2},k}, \mathbf{y}^{k-1}), \ldots, p(\mathbf{x}_k|\mathbf{e}_{1:N}, \mathbf{y}_{e_{1:N},k}, \mathbf{y}^{k-1})$ were encountered. Thus to obtain the n^{th} intermediate density, the measurement $\mathbf{y}_{e_{n,k},k}$ is used to update the prior $p(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{e_{1:n-1},k}, \mathbf{y}^{k-1})$. In Section 6.3.2 we discussed a particular technique, namely progressive correction, that could be applied to reduce the approximation error when updating $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n-1}, \mathbf{y}_{e_{1:n-1},k}, \mathbf{y}^{k-1})$ to arrive at $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{e_{1:n},k}, \mathbf{y}^{k-1})$. Now we present yet another improvement that is used to obtain even better approximations.

Note that the update process is happening under the condition that the new measurement $\mathbf{y}_{e_n,k}$ is associated with the path configuration corresponding to the n^{th} row of the path configuration matrix $\mathbf{C}$. Apart from the path configurations $(1,2)$ and $(2,1)$ all other path configurations are tied to at most one wall. As an example path configurations $(1,0)$ or $(2,2)$ are tied to walls labelled 1 and 2 respec-

tively, whereas the direct path $(0,0)$ is not tied to any wall at all. Thus, the likelihood function $p(\mathbf{y}_{e_n,k}|\mathbf{x}_k, e_n)$ only depends on a subset of elements of the state vector, namely the kinematic state $\mathbf{T}_k$ and the elements related to the walls that are tied to the path configuration represented by $\mathbf{C}_n$. Conversely, wall parameters that are not linked to $\mathbf{C}_n$ do not affect the likelihood function under consideration. We partition the state vector into two components $\mathbf{\Omega}_{1,k}$ and $\mathbf{\Omega}_{2,k}$, where $\mathbf{\Omega}_{1,k}$ corresponds to the elements of $\mathbf{x}_k$ that influence the likelihood function and $\mathbf{\Omega}_{2,k}$ to house to the rest of the elements of $\mathbf{x}_k$. So far the approach has been to use particle filtering on the entire state vector $\mathbf{x}_k$ to approximate $\hat{p}(\mathbf{x}_k|\mathbf{e}_{1:n}, \mathbf{y}_{e_{1:n},k}, \mathbf{y}^{k-1})$. A better approach would be to use particle filtering only on $\mathbf{\Omega}_{1,k}$ to obtain $\hat{p}(\mathbf{\Omega}_{1,k}|\mathbf{e}_{1:n}, \mathbf{y}_{e_{1:n},k}, \mathbf{y}^{k-1})$ and then use an analytical method to obtain $\hat{p}(\mathbf{\Omega}_{2,k}|\mathbf{e}_{1:n}, \mathbf{y}_{e_{1:n},k}, \mathbf{y}^{k-1})$. The motivation behind this approach is that it is desirable to use analytical methods whenever possible as it yields a lower variance approximation. This is also the concept underlying marginalised particle filter [Cappé 2007, Schon 2005]. Next, we explain how marginalising could be applied using the conditional Gaussian formula.

Without loss of generality consider the generic problem of approximating $p(\mathbf{\Omega}_1, \mathbf{\Omega}_2|\mathbf{y})$ where $p(\mathbf{y}|\mathbf{\Omega}_1, \mathbf{\Omega}_2) = p(\mathbf{y}|\mathbf{\Omega}_1)$. In other words, the likelihood function does not depend on $\mathbf{\Omega}_2$. Suppose we wish to approximate $p(\mathbf{\Omega}_1, \mathbf{\Omega}_2|\mathbf{y})$ as a multivariate Gaussian as shown in (6.56).

$$\hat{p}(\mathbf{\Omega}_1, \mathbf{\Omega}_2|\mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{\Omega}_1 \\ \mathbf{\Omega}_2\end{bmatrix}; \begin{bmatrix}\hat{\mathbf{\Omega}}_1 \\ \hat{\mathbf{\Omega}}_2\end{bmatrix}, \begin{bmatrix}\hat{\mathbf{D}}_{1,1} & \hat{\mathbf{D}}_{1,2} \\ \hat{\mathbf{D}}_{2,1} & \hat{\mathbf{D}}_{2,2}\end{bmatrix}\right). \tag{6.56}$$

Assume that the prior $p(\mathbf{\Omega}_1, \mathbf{\Omega}_2)$ is given by the following with all the parameters known:

$$\hat{p}(\mathbf{\Omega}_1, \mathbf{\Omega}_2) = \mathcal{N}\left(\begin{bmatrix}\mathbf{\Omega}_1 \\ \mathbf{\Omega}_2\end{bmatrix}; \begin{bmatrix}\tilde{\mathbf{\Omega}}_1 \\ \tilde{\mathbf{\Omega}}_2\end{bmatrix}, \begin{bmatrix}\tilde{\mathbf{D}}_{1,1} & \tilde{\mathbf{D}}_{1,2} \\ \tilde{\mathbf{D}}_{2,1} & \tilde{\mathbf{D}}_{2,2}\end{bmatrix}\right). \tag{6.57}$$

In Section 6.3.2 we described a Monte Carlo procedure for approximating the moments in (6.56). Using this procedure to approximate all moments in (6.56) introduces unnecessary variance. Since the measurements depend only on $\mathbf{\Omega}_1$, we can instead apply the procedure of Section 6.3.2 to find approximations of $\hat{\mathbf{\Omega}}_1$ and $\hat{\mathbf{D}}_{1,1}$ only. The remaining posterior moments can be found using the conditional Gaussian theorem [Zhang 2005, p. 186]:

Theorem 6.1. *If*

$$p(\mathbf{x}_1, \mathbf{x}_2) = \mathcal{N}\left(\cdot; \begin{bmatrix}\boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2\end{bmatrix}, \begin{bmatrix}\mathbf{\Sigma}_{1,1} & \mathbf{\Sigma}_{1,2} \\ \mathbf{\Sigma}_{2,1} & \mathbf{\Sigma}_{2,2}\end{bmatrix}\right);$$

then,

$$p(\mathbf{x}_1|\mathbf{x}_2) = \mathcal{N}\left(\cdot; \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{1,1} - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\Sigma}_{2,1}\right).$$

Proof. See Appendix 6.A. □

We now show how $\hat{\boldsymbol{\Omega}}_2$ can be analytically obtained using $\tilde{\boldsymbol{\Omega}}_1, \tilde{\mathbf{D}}_{1,1}$, and Theorem 6.1:

$$\begin{aligned}
\hat{\boldsymbol{\Omega}}_2 &= \int \boldsymbol{\Omega}_2 p(\boldsymbol{\Omega}_2|\mathbf{y}) d\boldsymbol{\Omega}_2, && (6.58)\\
&= \int\int \boldsymbol{\Omega}_2 p(\boldsymbol{\Omega}_2|\mathbf{y}, \boldsymbol{\Omega}_1) p(\boldsymbol{\Omega}_1|\mathbf{y}) d\boldsymbol{\Omega}_1 d\boldsymbol{\Omega}_2, && (6.59)\\
&= \int \left[\int \boldsymbol{\Omega}_2 p(\boldsymbol{\Omega}_2|\boldsymbol{\Omega}_1) d\boldsymbol{\Omega}_2\right] p(\boldsymbol{\Omega}_1|\mathbf{y}) d\boldsymbol{\Omega}_1, && (6.60)\\
&= \int \left[\int \boldsymbol{\Omega}_2 \mathcal{N}(\boldsymbol{\Omega}_2; \tilde{\boldsymbol{\Omega}}_2 + \mathbf{A}(\boldsymbol{\Omega}_1 - \tilde{\boldsymbol{\Omega}}_1), \tilde{\mathbf{D}}_{2,2} - \mathbf{A}\tilde{\mathbf{D}}_{1,2}) d\boldsymbol{\Omega}_2\right] \mathcal{N}(\boldsymbol{\Omega}_1; \hat{\boldsymbol{\Omega}}_1, \hat{\mathbf{D}}_{1,1}) d\boldsymbol{\Omega}_1, && (6.61)\\
&= \int \left[\tilde{\boldsymbol{\Omega}}_2 + \mathbf{A}(\boldsymbol{\Omega}_1 - \tilde{\boldsymbol{\Omega}}_1)\right] \mathcal{N}(\boldsymbol{\Omega}_1; \hat{\boldsymbol{\Omega}}_1, \hat{\mathbf{D}}_{1,1}) d\boldsymbol{\Omega}_1, && (6.62)\\
&= \tilde{\boldsymbol{\Omega}}_2 + \mathbf{A}(\hat{\boldsymbol{\Omega}}_1 - \tilde{\boldsymbol{\Omega}}_1), && (6.63)
\end{aligned}$$

where

$$\mathbf{A} = \tilde{\mathbf{D}}_{2,1}\tilde{\mathbf{D}}_{1,1}^{-1}.$$

Equation (6.59) follows from Chapman-Kolmogorov formula, while equation (6.60) follows from the use of Fubini's theorem to switch the order of integration along with the fact that $\boldsymbol{\Omega}_2$ and $\mathbf{y}$ are independent of each other when conditioned on $\boldsymbol{\Omega}_1$. Equation (6.61) follows from the application of Theorem 6.1. Finally (6.62) and (6.63) are obtained by noting that the solutions of the integration operations in question are the means of the respective normal distributions.

Use of a similar procedure makes it straightforward to obtain the expressions for the remaining three parameters of the distribution (6.56). They are

$$\begin{aligned}
\hat{\mathbf{D}}_{2,2} &= \tilde{\mathbf{D}}_{2,2} - \mathbf{A}\tilde{\mathbf{D}}_{1,2} + \mathbf{A}\hat{\mathbf{D}}_{1,1}\mathbf{A}', && (6.64)\\
\hat{\mathbf{D}}_{2,1} &= \mathbf{A}\hat{\mathbf{D}}_{1,1}, && (6.65)\\
\hat{\mathbf{D}}_{1,2} &= \hat{\mathbf{D}}_{2,1}'. && (6.66)
\end{aligned}$$

Algorithm 6.4, which is included in the Appendix 6.B, illustrates how all the improvements (SJPDAF, progressive correction, and marginalising) are integrated into a single filter.

6.3.5 Handling a mix of circular and linear variables in the state vector

Note that the state variable is a mixture of linear and circular parameters where the circularity is contributed by the slope parameters of the walls. Thus, whenever a parameter like sample mean is calculated special attention is required for the circular components of the variable. As an example for the case of calculating the sample mean of the state vector, the directional statistics [Mardia 1972] are used to obtain the sample means of the circular variables due to the reasons highlighted in Appendix 4.A. Additionally when a difference between two slope variables is calculated (one example of this occurring is while calculating the likelihood function which involves subtracting a circular quantity from the AOA measurement) we use modulo 2π operation to preserve the circularity.

6.4 Results and discussion

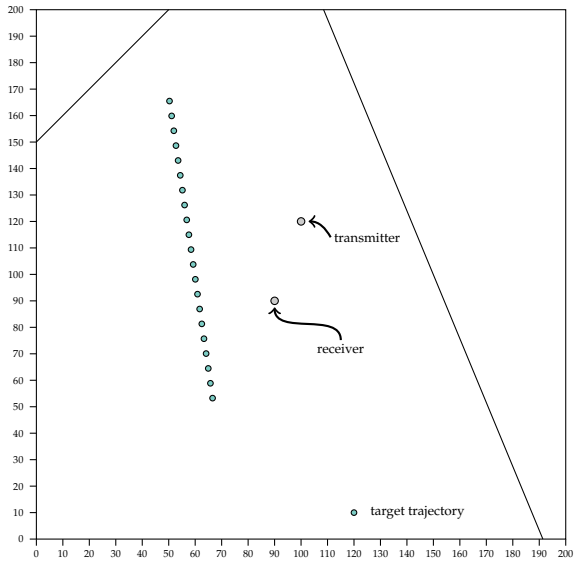


Figure 6.4: Multipath environment.

Consider the simulation setup depicted in Figure 6.4. The size of the area is set by letting $K = 200$. The two walls are represented by the parameters (α, β) given by $[\pi/4 \;\; 150\cos(\pi/4)]'$ and $[5\pi/8 \;\; 250\cos(3\pi/4)]'$ respectively. The coordinates of the radar transmitter and receiver are $(100, 120)$ and $(90, 90)$ respectively. The initial prior for the target kinematic state $\mathbf{T}_0$ is a multivariate normal with mean $[60\ 0\ 50\ 5]'$ and the covariance matrix is diagonal with the diagonal elements given

by $[9\ 1\ 9\ 1]'$. The covariance matrix $\mathbf{Q}$ for the process noise is

$$\mathbf{Q} = \begin{bmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{bmatrix} \otimes \begin{bmatrix} \frac{(\Delta T)^3}{3} & \frac{(\Delta T)^2}{2} \\ \frac{(\Delta T)^2}{2} & \Delta T \end{bmatrix}, \tag{6.67}$$

where $\kappa_1 = \kappa_2 = 0.5$ and the state sampling interval ΔT is set as 1. Measurement noise parameters are set by $\sigma^2 = 1$ and $\sigma_\theta = \pi/90$.

We tested the filter against two separate prior distributions for the wall parameters. The first is a normal approximation to the prior that was derived in Section 6.2.4. Let this prior be denoted by $p_{w,1}(\cdot)$. The second prior $p_{w,2}(\cdot)$ is less uncertain about the environment compared to $p_{w,1}(\cdot)$, but still enforces relatively large uncertainty on the environment. The two prior distributions $p_{w,1}$ and $p_{w,2}$ are given by the following formulae:

$$p_{w,1}(\mathbf{w}_1, \mathbf{w}_2) = \mathcal{N}(\cdot; \boldsymbol{\mu}_{w,1}, \mathbf{C}_{w,1}), \tag{6.68}$$

$$p_{w,2}(\mathbf{w}_1, \mathbf{w}_2) = \mathcal{N}(\cdot; \boldsymbol{\mu}_{w,2}, \mathbf{C}_{w,2}), \tag{6.69}$$

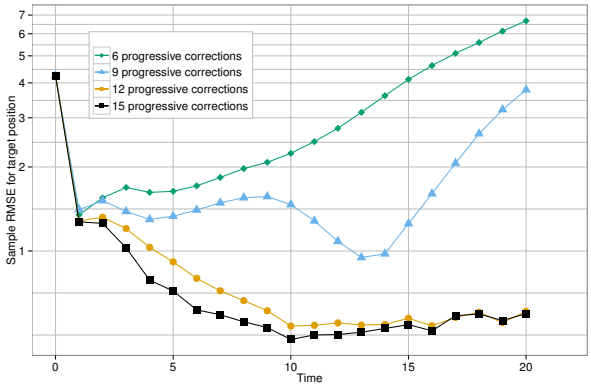
where

$$\begin{aligned}
\boldsymbol{\mu}_{w,1} &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \pi/2 \\ -200/\pi \end{bmatrix}, \\
\mathbf{C}_{w,1} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} \pi^2/12 & -200/\pi \\ -200/\pi & 200^2(2\pi^2 - 6 + \pi)/(6\pi^2) \end{bmatrix}, \\
\boldsymbol{\mu}_{w,2} &= \begin{bmatrix} \pi(1/4 + 27/180) \\ 150\cos(\pi/4) - 35 \\ \pi(5/8 - 21/180) \\ 250\cos(3\pi/4) + 40 \end{bmatrix}, \\
\mathbf{C}_{w,2} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} (30\pi/180)^2 & 0 \\ 0 & 70^2 \end{bmatrix}.
\end{aligned}$$

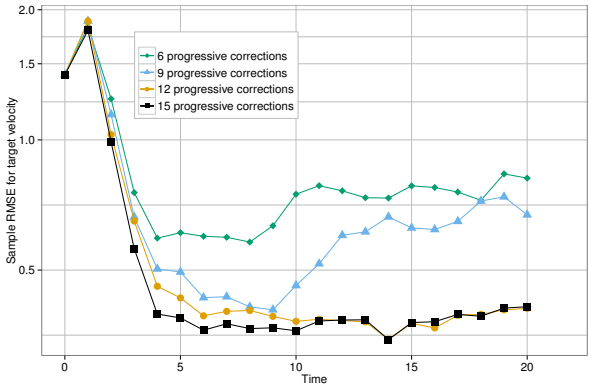
To put the level of uncertainty assumed by the prior distributions in some context, note that for $p_{w,1}$, the standard deviation (of the marginal distributions) for the intercept (β) is approximately 106.8 and for the angle parameter (α) is approximately 52 degrees. The corresponding numbers for $p_{w,2}$ are 70 and 30 respectively.

In the previous section, we discussed three techniques as improvements to the basic algorithm outlined in Algorithm 6.1, namely "progressive correction", "SJPDAF", and "marginalising using conditional Gaussian formula". First, we explore the contribution of each of these methods on the overall performance. Figure

6.5 shows the effect of the number of correction steps on the filter performance. For this simulation $p_{w,2}(\cdot)$ was used with $M = 1000$ target samples drawn at each progressive correction step. The number J of association samples drawn is fixed at 300 for all the simulations appearing in this section. Further, all the simulation results that follow are produced by Monte Carlo using 150 realisations of each experiment.



(a)



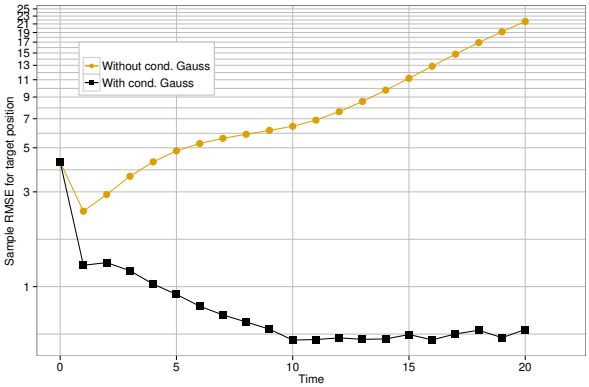
(b)

Figure 6.5: Simulating the effect of progressive correction by varying the number of correction steps. In (a) position RMSE in meters is plotted against time while in (b) velocity RMSE in meters per second is plotted against time.

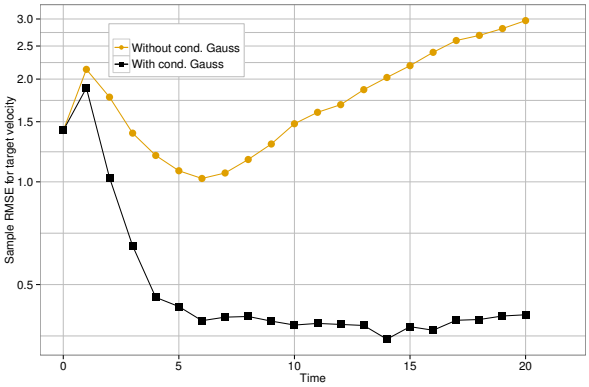
It is evident from Figure 6.5 that the progressive correction plays an important role on the filter performance. With fewer corrections such as 6 and 9, the filter lost track, but 12 correction steps were sufficient to produce good results. Progressive correction was particularly important in our filtering problem because of the large uncertainty introduced by the prior.

Figure 6.6 shows the results of a similar experiment, but this time the objective was to investigate the utility of marginalising using the conditional Gaussian

formula. Again the prior used here is $p_{w,2}$ and for both curves 12 progressive corrections with $M = 1000$ samples were used. The results clearly show the effec-



(a)



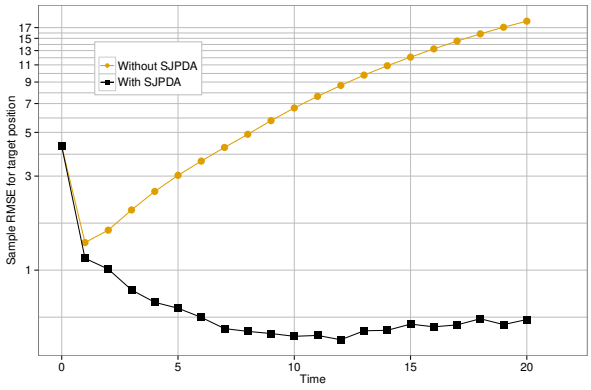
(b)

Figure 6.6: Simulating the utility of marginalising using the conditional Gaussian formula. In (a) position RMSE in meters is plotted against time while in (b) velocity RMSE in meters per second is plotted against time.

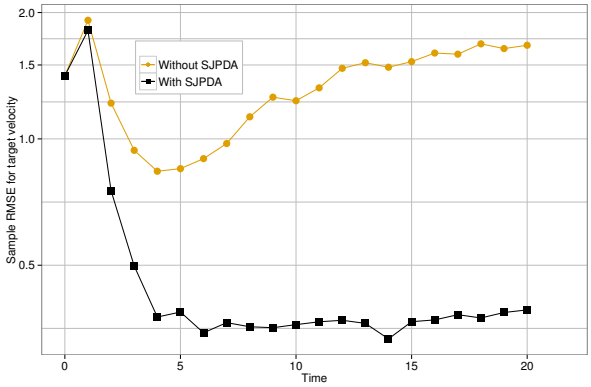
tiveness of marginalising. To achieve the same level of performance as the black curve whilst not marginalising, we would have had to set the sample size M to a much higher value at a cost of computational expense.

The more uncertain prior $p_{w,1}$ was used to assess the contribution of the SJPDAF algorithm and the result is illustrated in Figure 6.7. Twelve progressive corrections were used for this experiment along with M set to 3000. The algorithm lost track when the SJPDAF was not used, which is not unexpected due to the reasoning discussed in Section 6.3.3.

Since the main objective of the work presented in this chapter was to provide a filtering solution to a largely uncertain multipath environment, it follows naturally to study the effect of the level of prior uncertainty on the filter performance. The



(a)

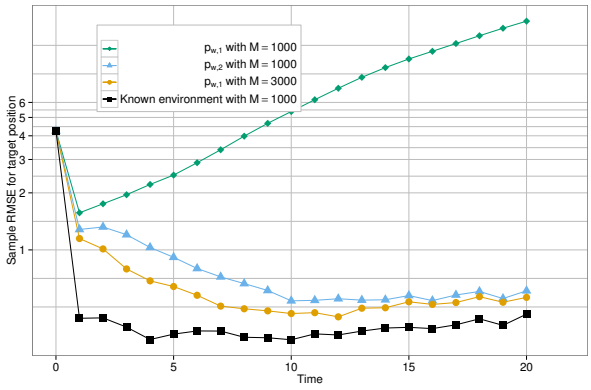


(b)

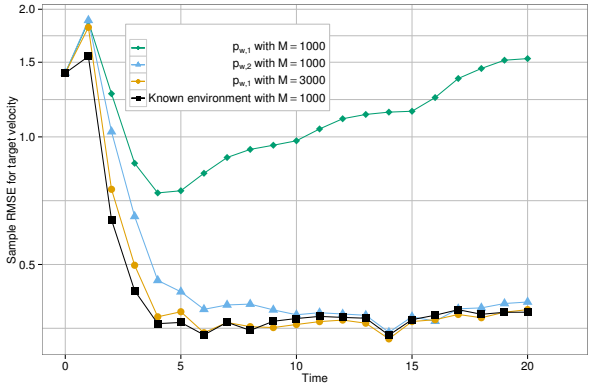
Figure 6.7: Simulating the utility of SJPDA algorithm. In (a) position RMSE in meters is plotted against time while in (b) velocity RMSE in meters per second is plotted against time.

filter was run against various uncertainty levels ranging from a deterministic environment (that is, where the wall parameters are assumed to be exactly known) through to $p_{w,1}$ and $p_{w,2}$. The results are shown in Figure 6.8. By observing the three curves in Figure 6.8 which had used $M = 1000$ samples we conclude that the results confirm the intuition that larger uncertainty makes the tracking problem harder. In fact with $M = 1000$ for $p_{w,1}$ the filter loses the track. However increasing the sample size to $M = 3000$ produces good results for the prior even with the largest uncertainty.

Note that the mean of the slope parameter under $p_{w,1}$ is $\pi/2$ and therefore the maximum divergence from this parameter occurs when the slope is 0 (that is, when the wall is horizontal). With the intention of assessing the filter performance with a maximum divergence from the mean slope parameter, we changed one of the walls to have the parameters $[0\ 35]'$ and ran the simulation against prior $p_{w,1}$ with



(a)



(b)

Figure 6.8: Assessing the effect of uncertainty on the filter performance. In (a) position RMSE in meters is plotted against time while in (b) velocity RMSE in meters per second is plotted against time.

12 progressive corrections and M set to 3000. The new multipath environment and the filter performance are shown in Figures 6.9 and 6.10 respectively. The results suggest that the algorithm was able to track the target under this challenging setup.

We observed while drawing association samples $\mathbf{e}_k^{(j)}$ for $j = 1, 2, \ldots, J$ that some associations are commoner than others, which is not unusual. We were able to exploit this fact to significantly improve the execution time by storing the result of probability calculations (6.39) and reusing them when needed.

The overall simulation results suggest that the proposed algorithm was able to track a target in a highly uncertain multipath environment. Each of the methods "progressive correction", "SJPDAF" and "marginalising using conditional Gaussian formula" contributed significantly to the success of the overall method while omitting any one of those led to severe performance degradation and even track loss for the simulations carried out.

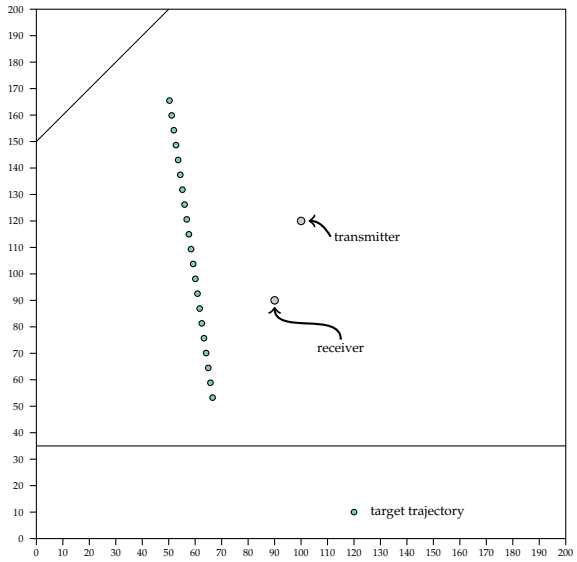
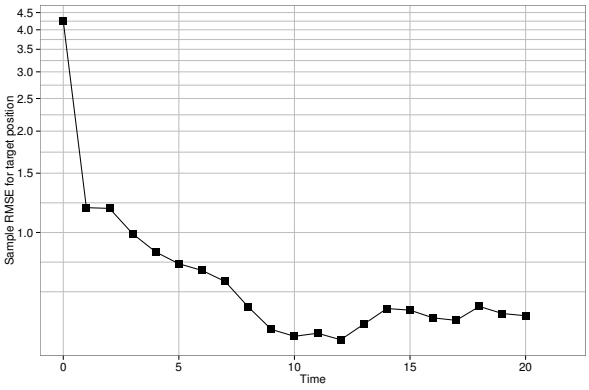


Figure 6.9: Multipath environment with a horizontal wall.

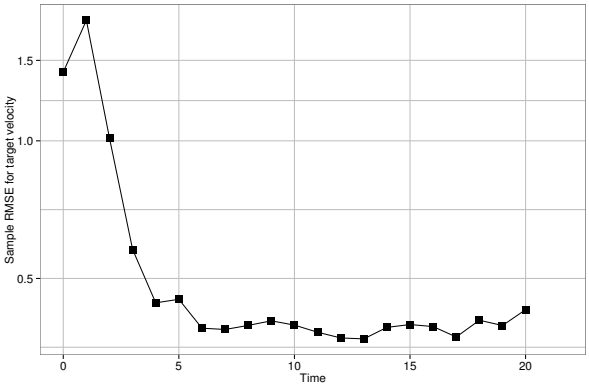
Though we have considered only a two wall setup and a clutter free model, the framework can be extended to include clutter and multiple unknown number of walls. The computational expense of the exact solution (6.21) increases exponentially with the number of clutter measurements and walls. The importance sampling based approach we proposed would not be subject to this increase as, for a fixed sample size J, its expense depends polynomially on the number of walls and clutter measurements. However, an increased sample size may be required to achieve good performance. The challenge, when the number of walls is unknown, can be handled within the same framework proposed in this chapter by introducing multiple target tracking techniques such as [Garcia-Fernandez 2011].

6.5 Conclusion

In this chapter, we have formulated a challenging problem of tracking a target in a highly uncertain multipath environment. We took a Bayesian approach in deriving a filtering solution to the problem. An importance sampling based filter was proposed to solve the tracking problem. Additionally, we proposed an importance sampling based method to avoid the typical exhaustive calculation involving a summation over the potentially large number of possible data association hypotheses. This method can be applied to general data association problems. The proposed filter incorporated recent techniques such as Set JPDAF algorithm and progressive correction. We included simulation results to illustrate the per-



(a)



(b)

Figure 6.10: Simulating the filter with a horizontal wall. In (a) position RMSE in meters is plotted against time while in (b) velocity RMSE in meters per second is plotted against time.

formance of the proposed filter. Though we have assumed a clutter free model with two walls the framework presented can be extended to incorporate clutter and multiple unknown walls whilst staying within the vicinity of the proposed solution.

Appendices

6.A Proof of Theorem 1

Proof. Define a random variable as follows:

$$\begin{bmatrix} \mathbf{x}_1 - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\mathbf{x}_2 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{I} & -\boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}. \tag{6.70}$$

Note that the distribution of the random vector in (6.70) is a multivariate normal with mean

$$\begin{bmatrix} \mathbf{I} & -\boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu}_1 - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\mu}_2 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \tag{6.71}$$

and covariance matrix

$$\begin{bmatrix} \mathbf{I} & -\boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{1,1} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,2} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\left(\boldsymbol{\Sigma}_{2,2}^{-1}\right)' \boldsymbol{\Sigma}_{1,2}' & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_{1,1} - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\Sigma}_{2,1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{2,2} \end{bmatrix}. \tag{6.72}$$

The above form implies that the marginal distribution of $\mathbf{x}_1 - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\mathbf{x}_2$ is multivariate normal with mean $\boldsymbol{\mu}_1 - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\mu}_2$ and covariance $\boldsymbol{\Sigma}_{1,1} - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\Sigma}_{2,1}$, which would lead to the desired result:

$$p(\mathbf{x}_1|\mathbf{x}_2) = \mathcal{N}\left(\cdot; \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{1,1} - \boldsymbol{\Sigma}_{1,2}\boldsymbol{\Sigma}_{2,2}^{-1}\boldsymbol{\Sigma}_{2,1}\right).$$

□

6.B Algorithms

Algorithm 6.1: A basic state estimation algorithm for the multipath filtering problem

Input: $\boldsymbol{\mu}_{k|k-1}, \mathbf{E}_{k|k-1}, \mathbf{y}_k$

for $j = 1$ **to** J **do**
- Set $\mathcal{A}_{1,j} = \{1, 2, \ldots, N\}$.
- Draw M samples $\mathbf{x}_k^{(1,1)}, \mathbf{x}_k^{(1,2)}, \ldots, \mathbf{x}_k^{(1,M)}$ from $\mathcal{N}(\cdot; \boldsymbol{\mu}_{k|k-1}, \mathbf{E}_{k|k-1})$.
- **for** $n = 1$ **to** N **do**
  - Let the t^{th} element in $\mathcal{A}_{n,j}$ be denoted by $a_{n,t}$ where $t \in \{1, 2, \ldots, N-n+1\}$.
  - **for** $t = 1$ **to** $N - n + 1$ **do**
    - Calculate $\psi_{n,t} = \sum_{m=1}^{M} p_n(\mathbf{y}_{e_n^{(j)},k} | \mathbf{e}_{1:n-1}^{(j)}, e_n^{(j)} = a_{n,t}, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1})$.
  - Draw an index t with $\text{Prob}(t = \tilde{t}) \propto \psi_{n,\tilde{t}}$.
  - Set $e_n^{(j)} = a_{j,t}$.
  - Set $\mathcal{A}_{n+1,j} = \mathcal{A}_{n,j} \backslash \{e_n^{(j)}\}$.
  - `/* Draw new samples from the updated conditional posterior + resample + regularize */`
  - Let $\boldsymbol{\mu}_{k,n}^{(j)}$ and $\mathbf{E}_{k,n}^{(j)}$ be the sample mean and covariance of the weighted particles denoted by the sample-weight pairs $\{\mathbf{x}_k^{(n,m)}, \tau_k^{(n,m)}\}_{m=1}^{M}$ where, $\tau_k^{(n,m)} = p_n(\mathbf{y}_{e_n^{(j)}} | \mathbf{e}_{1:n}^{(j)}, \mathbf{x}_k^{(n,m)}, \mathbf{y}^{k-1})$.
  - **for** $m = 1$ **to** M **do**
    - Draw an index l_m with $\text{Prob}(l_m = \tilde{m}) \propto \tau_k^{(n,\tilde{m})}$.
    - Draw $\mathbf{x}_k^{(n+1,m)} \sim \mathcal{N}(\cdot; \mathbf{x}_k^{(n,l_m)}, h^2 \mathbf{E}_{k,n}^{(j)})$ where h is the bandwidth parameter.
- Set $\mathbf{e}_k^{(k)} = [e_1^{(j)}\ e_2^{(j)}\ \ldots\ e_N^{(j)}]'$.
- Set $\tilde{\phi}_j = \prod_{n=1}^{N} \sum_{t=1}^{N-j+1} \psi_{n,t}$.

Set $\phi_i = \tilde{\phi}_i / \sum_{j=1}^{J} \tilde{\phi}_j$ for all $i = 1, 2, \ldots, J$.

Set $\boldsymbol{\mu}_{k|k} = \sum_{j=1}^{J} \phi_j \boldsymbol{\mu}_{k,N}^{(j)}$.

Set $\mathbf{E}_{k|k} = \sum_{j=1}^{J} \phi_j \mathbf{E}_{k,N}^{(j)} + \sum_{j=1}^{J} \phi_j (\boldsymbol{\mu}_{k,N}^{(j)} - \boldsymbol{\mu}_{k|k})(\boldsymbol{\mu}_{k,N}^{(j)} - \boldsymbol{\mu}_{k|k})'$.

Output: $\boldsymbol{\mu}_{k|k}, \mathbf{E}_{k|k}$.

Algorithm 6.2: Progressive correction algorithm

Input: $\boldsymbol{\mu}_0, \mathbf{E}_0, n, e_n, \mathbf{y}_{e_n}$

/* The index n defines the path configuration considered in the current context (that is, the n^{th} row of $\mathbf{C}$). */

/* e_n is the n^{th} element of the association vector. */

/* $\mathbf{y}_{e_n}$ is the measurement associated with the path configuration given by $\mathbf{C}_n$ (n^{th} row of $\mathbf{C}$). */

Draw $\tilde{\mathbf{x}}_1^{(m)} \sim \mathcal{N}(\cdot; \boldsymbol{\mu}_0, \mathbf{E}_0)$ for $m = 1, 2, \ldots, M$.

Choose B exponents $\gamma_1, \gamma_2, \ldots, \gamma_B$ to be used in the progressive correction such that $0 \leq \gamma_b \leq 1$ and $\sum_{b=1}^{B} \gamma_b = 1$ for $b = 1, 2, \ldots, B$.

for $b = 1$ **to** B **do**

  for $m = 1$; **to** M **do**

    Let $\tau_{b,m} = [p(\mathbf{y}_{e_n} | e_n, \tilde{\mathbf{x}}_b^{(m)})]^{\gamma_b}$.

  Let $\omega_b = \frac{1}{B} \sum_{b=1}^{B} \tau_{b,m}$.

  /* Resample and generate the next set of samples */

  Let $\mathbf{E}$ be the sample covariance of $\tilde{\mathbf{x}}_b^{(m)}$ with weight $\tau_{b,m}$ for $m = 1, 2, \ldots, M$.

  for $m = 1$ **to** M **do**

    Draw an index l_m with $\text{Prob}(l_m = \tilde{m}) \propto \tau_{b,\tilde{m}}$.

    Draw $\tilde{\mathbf{x}}_{b+1}^{(m)} \sim \mathcal{N}(\cdot; \tilde{\mathbf{x}}_b^{(l_m)}, h^2 \mathbf{E})$ where h is the bandwidth parameter.

Set normalising constant $\phi = \prod_{b=1}^{B} \omega_b$.

Set $\boldsymbol{\mu}, \mathbf{E}$ to be the sample mean and covariance of $\{\tilde{\mathbf{x}}_{B+1}^{(m)}\}_{m=1}^{M}$, respectively.

Output: $\boldsymbol{\mu}, \mathbf{E}$, and ϕ.

Algorithm 6.3: Set JPDAF adapted to the multipath filtering problem

Input: $\{\boldsymbol{\mu}^{(j)}, \mathbf{E}^{(j)}, \phi_j, \mathbf{e}^{(j)}\}_{j=1}^{J}$.

Let R be the number of unique data association vectors among $\{\mathbf{e}^{(j)}\}_{j=1}^{J}$.

Let $\mathbf{h}_r$ be the r^{th} association hypothesis, for $r = 1, 2, \ldots, R$.

Calculate all R association probabilities where for the r^{th} hypothesis $\mathbf{h}_r$ the probability is given by:

$$\text{Prob}(\mathbf{h}_r) = \sum_j \phi_j \quad \text{where the summation is over } j \text{ such that } \mathbf{e}^{(j)} = \mathbf{h}_r.$$

Set $\tilde{\boldsymbol{\mu}}_r$, $\tilde{\mathbf{E}}_r$ to be the mean and covariance under hypothesis $\mathbf{h}_r$ (for $r = 1, \ldots, R$); that is, the mixture mean and covariance of all the mixture components given by mean $\boldsymbol{\mu}^{(j)}$ and covariance $\mathbf{E}^{(j)}$ for all j such that $\mathbf{e}^{(j)} = \mathbf{h}_r$.

Define $\tilde{\mathbf{w}}_r^l$ to be the vector of parameters (α and β parameters) of the wall labelled l ($l \in \{1, 2\}$) extracted from $\tilde{\boldsymbol{\mu}}_r$.

while *not converged* **do**

  For $l = 1, 2$, calculate $\hat{\mathbf{w}}^l = \sum_{r=1}^{R} \text{Prob}(\mathbf{h}_r)\tilde{\mathbf{w}}_r^l$.

  Draw a hypothesis t randomly with $\Pr(t = \tilde{t}) = \text{Prob}(\mathbf{h}_{\tilde{t}})$.

  Calculate:

$$\gamma = 2\text{Prob}(\mathbf{h}_t)\left\{(\hat{\mathbf{w}}^1 - \hat{\mathbf{w}}^2)'(\tilde{\mathbf{w}}_t^1 - \tilde{\mathbf{w}}_t^2) + \text{Prob}(\mathbf{h}_t)\|\tilde{\mathbf{w}}_t^1 - \tilde{\mathbf{w}}_t^2\|^2\right\}.$$

  if $0 < \gamma$ **then**

    `/* Swap the labels */`

    Appropriately swap the elements related to wall labels 1 and 2 in the mean vector $\tilde{\boldsymbol{\mu}}_t$ and covariance matrix $\tilde{\mathbf{E}}_t$.

Set $\boldsymbol{\mu} = \sum_{r=1}^{R} \Pr(\mathbf{h}_r)\tilde{\boldsymbol{\mu}}_r$.

Set $\mathbf{E} = \sum_{r=1}^{R} \Pr(\mathbf{h}_r)\tilde{\mathbf{E}}_r + \sum_{r=1}^{R} \Pr(\mathbf{h}_r)(\tilde{\boldsymbol{\mu}}_r - \boldsymbol{\mu})(\tilde{\boldsymbol{\mu}}_r - \boldsymbol{\mu})'$.

Output: $\boldsymbol{\mu}$, $\mathbf{E}$.

Algorithm 6.4: Improved recursive state estimation algorithm

Input: $\boldsymbol{\mu}_{k|k-1}, \mathbf{E}_{k|k-1}, \mathbf{y}_k$

for $j = 1$ **to** J **do**
- Set $\mathcal{A}_{1,j} = \{1, 2, \ldots, N\}$
- Set $\boldsymbol{\mu}_{k,0}^{(j)} = \boldsymbol{\mu}_{k|k-1}$
- Set $\mathbf{E}_{k,0}^{(j)} = \mathbf{E}_{k|k-1}$
- **for** $n = 1$ **to** N **do**
  - Let the t^{th} element in $\mathcal{A}_{n,j}$ be denoted by $a_{n,t}$ where $t \in \{1, 2, \ldots, N-n+1\}$.
  - **for** $t = 1$ **to** $N - n + 1$ **do**
    - - Let $[\boldsymbol{\mu}_{k,n,t}, \mathbf{E}_{k,n,t}, \psi_{n,t}]$ = return values of the progressive correction algorithm with input arguments $[\boldsymbol{\mu}_{k,n-1}^{(j)}, \mathbf{E}_{k,n-1}^{(j)}, n, a_{n,t}, \mathbf{y}_{a_{n,t},k}]$.
    - - Group the state variable in to two components, $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$ such that the likelihood function $p(\mathbf{y}_{a_{n,t},k}|\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, e_n = a_{n,t}) = p(\mathbf{y}_{a_{n,t},k}|\boldsymbol{\Omega}_1, e_n = a_{n,t})$.
    - - From $\boldsymbol{\mu}_{k,n,t}$ and $\mathbf{E}_{k,n,t}$ extract the sub-blocks corresponding to $\boldsymbol{\Omega}_1$.
    - - Use the marginalising method explained in 6.3.4 on the extracted sub-blocks from the above step to replace the sub blocks of $\boldsymbol{\mu}_{k,n,t}$ and $\mathbf{E}_{k,n,t}$ relating to $\boldsymbol{\Omega}_2$.
  - Draw an index t with $\text{Prob}(t = \tilde{t}) \propto \psi_{n,\tilde{t}}$.
  - Set $e_n^{(j)} = a_{n,t}$.
  - Set $\mathcal{A}_{n+1,j} = \mathcal{A}_{n,j} \backslash \{e_n^{(j)}\}$.
  - Set $\boldsymbol{\mu}_{k,n}^{(j)} = \boldsymbol{\mu}_{k,n,t}$.
  - Set $\mathbf{E}_{k,n}^{(j)} = \mathbf{E}_{k,n,t}$.
- Set $\mathbf{e}_k^{(j)} = [e_1^{(j)} \; e_2^{(j)} \; \ldots \; e_N^{(j)}]'$
- Set $\tilde{\phi}_j = \prod_{n=1}^{N} \sum_{t=1}^{N-j+1} \psi_{n,t}$

Set $\phi_i = \tilde{\phi}_i / \sum_{j=1}^{J} \tilde{\phi}_j$ for all $i = 1, 2, \ldots, J$

Set $[\boldsymbol{\mu}_{k|k}, \mathbf{E}_{k,k}]$ to be the output of the SJPDAF algorithm with input parameters $\{\boldsymbol{\mu}_{k,N}^{(j)}, \mathbf{E}_{k,N}^{(j)}, \phi_n, \mathbf{e}_k^{(j)}\}_{j=1}^{J}$.

Output: $\boldsymbol{\mu}_{k|k}, \mathbf{E}_{k|k}$.

CHAPTER 7

Conclusion

This thesis has focused on the study of exploiting multipath reflections in radar to track the state of a moving target. These research efforts contribute to the current paradigm shift occurring in the radar statistical signal processing research community, where the long perceived view of multipath as a nuisance is challenged.

We have placed heavy emphasis on modelling the uncertain environment. Thus, naturally we were led to embrace the Bayesian framework for inference where unknown parameters are treated as random quantities.

In Chapter 3 we have presented a model for tracking a moving target in a partially known multipath radar environment. Within the same chapter, the performance bounds for tracking a moving target using multipath were derived. The model formulation permits an arbitrary number of multipath causing walls without imposing any restriction on their relative placement. The term *partially known* means that the locations of the walls are known only up to the accuracy of several wavelengths. This assumption is realistic and captures the mild errors present in the map used to assess the environment. We modelled this uncertainty in the environment by introducing a uniformly distributed random phase shift to the radar measurement equation. We further relaxed the assumption that the reflectivity factors of the walls are known and treated them as random parameters. However, these assumptions, which are valid in a practical setting, presented a challenging estimation problem. We proposed a Markov Chain Monte Carlo based particle filtering method in Chapter 5 to track a target under these assumptions. In Chapter 6 we further increased the uncertainty imposed in a multipath environment; as an example consider a situation arising when a map of the environment is not available. The resulting data association problem and the tracking problem yet again turned out to be very challenging. We proposed an importance sampling based method to solve these problems. In Chapter 4 we presented a generalisation of a multivariate von-Mises distribution that was developed while trying to solve the partially known multipath environment problem. This generalisation can be used to conveniently solve inference problems that might arise in real world applications involving multiple circular varialbles.

We believe that the outcome of this thesis will contribute to accelerate the widespread adaptation of multipath reflections in modern radar based tracking. It is our sincere hope that some of the methods proposed in this thesis could be adapted to solve general signal processing problems.

7.1 Future Work

- We believe that the resource allocation in multipath radar raises important research questions. The optimal design of radar waveforms and scheduling in multipath tracking is a topic that should be pursued in the future. Additionally the scheduling problem extends in space dimension as well; that is answering questions such as "where should the transmitters and receivers be placed to improve tracking?".

- In the work presented in Chapter 6, the analysis was restricted to the situation where only two walls exist in the environment. We assumed perfect radar detections and did not consider the possibility of clutter. Further, the walls were represented by continuous lines rather than by line segments. More work is required to relax these assumptions.

- We saw in Chapter 5 that Bayesian progressive correction could either enhance or detract the filter performance depending on the signal-to-noise ratio. A method to adaptively apply progressive correction based on the signal strength is desirable.

- In statistical procedures model checking is an important step. Thus, after extending the proposed methods from 2-dimensional geometry to 3-dimensional geometry, it would be worthwhile to assess the proposed models against real data. Since our model assumptions are based on substantive reasoning, we believe that the proposed models and methods fall to the latter category in the famous quote by G.E. Box: "All models are wrong; some models are useful".

Bibliography

[Abramowitz 1964] M. Abramowitz and I. A. Stegun. Handbook of mathematical functions with formulas, graphs, and mathematical tables. Dover Publications, 1964. (Cited on page 64.)

[Anderson 1979] Brian Anderson and John B. Moore. Optimal filtering. Prentice-Hall Information and System Sciences Series, Englewood Cliffs: Prentice-Hall, 1979, 1979. (Cited on pages 20 and 96.)

[Arulampalam 2002] 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, no. 2, pages 174–188, 2002. (Cited on pages 22, 25, 60, 97 and 120.)

[Bailey 2006] T. Bailey and H. F. Durrant-Whyte. *Simultaneous localization and mapping (SLAM): Part II*. IEEE Robotics & Automation Magazine, pages 108–117, September 2006. (Cited on page 110.)

[Bar-Shalom 2009] Y. Bar-Shalom, F. Daum and J. Huang. *The probabilistic data association filter*. IEEE Control Systems Magazine, pages 82–100, December 2009. (Cited on pages 110 and 111.)

[Barbosa 2008] P. R. Barbosa. *Multitarget-multisensor tracking in an urban environment: A closed-loop approach*. Signal and Data Processing of Small Targets, vol. 6969, pages W9690–W9690, 2008. (Cited on pages 37 and 109.)

[Best 1979] D. J. Best and N.I. Fisher. *Efficient simulation of the von Mises distribution*. Journal of the Royal Statistical Society. Series C (Applied Statistics), vol. 28, pages 152–157, 1979. (Cited on pages 66 and 92.)

[Blackman 2004] S. S. Blackman. *Multiple hypothesis tracking for multiple target tracking*. Aerospace and Electronic Systems Magazine, IEEE, vol. 19, no. 1, pages 5–18, 2004. (Cited on page 124.)

[Blom 2011] H. A. P. Blom and E. A. Bloem. *Optimal decomposed particle filtering of two closely spaced Gaussian targets*. In IEEE Conference on Decision and Control and European Control Conference, pages 7896–7901, 2011. (Cited on page 112.)

[Cappé 2007] O. Cappé, S. J. Godsill and E. Moulines. *An overview of existing methods and recent advances in sequential Monte Carlo*. In Proceedings of the IEEE, 2007. (Cited on page 128.)

[Chakraborty 2010] B. Chakraborty, Y. Li, J. J. Zhang, T. Trueblood, A. Papandreou-Suppappola and D. Morrell. *Multipath exploitation with adaptive waveform design*

for tracking in urban terrain. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3894–3897, 2010. (Cited on pages 37, 83, 84, 85 and 109.)

[Chakraborty 2011] B. Chakraborty, J. J. Zhang, A. Papandreou-Suppappola and D. Morrell. *Urban terrain tracking in high clutter with waveform-agility*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3640–3643, 2011. (Cited on pages 37, 83, 84, 85, 109 and 110.)

[Chib 1995] S. Chib. *Marginal likelihood from the Gibbs output*. Journal of the American Statistical Association, vol. 90, no. 432, pages 1313–1321, 1995. (Cited on pages 58, 67 and 68.)

[Chopin 2002] N. Chopin. *A sequential particle filter method for static models*. Biometrika, vol. 89, no. 3, pages 539–552, 2002. (Cited on page 86.)

[Cowles 1996] M. K. Cowles and B. P. Carlin. *Markov Chain Monte Carlo convergence diagnostics: A comparative review*. Journal of the American Statistical Association, vol. 91, no. 434, pages 883–904, 1996. (Cited on page 97.)

[Durrant-Whyte 2006] H.F. Durrant-Whyte and T. Bailey. *Simultaneous localization and mapping: Part I*. Robotics & Automation Magazine, pages 99–108, June 2006. (Cited on page 110.)

[Garcia-Fernandez 2011] Ángel F. Garcia-Fernandez. *Detection and tracking of multiple targets using wireless sensor networks*. PhD thesis, Universidad Politécnica De Madrid, 2011. (Cited on page 136.)

[Geman 1984] S. Geman and D. Geman. *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*. IEEE Transactions on Pattern Analysis and Machine Learning, vol. PAMI-6, no. 6, pages 721–741, 1984. (Cited on page 91.)

[Gilks 2001] W. R. Gilks and C. Berzuini. *Following a moving target - Monte Carlo inference for dynamic Bayesian models*. Journal of the Royal Statistical Society: Series B (Statistical Methodology), vol. 63, no. 1, pages 127–146, February 2001. (Cited on page 25.)

[Gill 1995] R. D. Gill and B. Y. Levit. *Applications of the van Trees Inequality: A Bayesian Cramer-Rao Bound*. Bernoulli, vol. 1, pages 59–79, 1995. (Cited on page 36.)

[Gordon 1993] N. J. Gordon, D. J. Salmond and A. F. M. Smith. *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*. IEE Proceedings F, Radar and Signal Processing,, vol. 140, no. 2, pages 107–113, 1993. (Cited on page 22.)

[Hamilton 1992] M. Hamilton and P. M. Schultheiss. *Passive ranging in multipath dominant environments, Part I: Known multipath parameters*. IEEE Transactions on Signal Processing, vol. 40, no. 1, pages 1–12, 1992. (Cited on page 36.)

[Hastings 1970] W. K. Hastings. *Monte Carlo sampling methods using Markov chains and their applications*. Biometrika, vol. 57, no. 1, pages 97–109, 1970. (Cited on page 91.)

[Hayvaci 2012a] H. T. Hayvaci, Antonio De Maio and D. Erricolo. *Performance analysis of diverse GLRT detectors in the presence of multipath*. In IEEE Radar Conference (RADAR), pages 902–906, 2012. (Cited on pages 83, 84 and 109.)

[Hayvaci 2012b] H. T. Hayvaci, P. Setlur and N. Devroye. *Maximum likelihood time delay estimation and Cramer-Rao bounds for multipath exploitation*. IEEE Radar Conference (RADAR), pages 764–768, 2012. (Cited on pages 36, 83, 84 and 109.)

[Ho 1964] Y. C. Ho and R. Lee. *A Bayesian approach to problems in stochastic estimation and control*. IEEE Transaction on Automatic Control, pages 333–338, 1964. (Cited on page 14.)

[Horridge 2006] P. Horridge and S. Maskell. *Real-time tracking of hundreds of targets with efficient exact JPDAF implementation*. In International Conference on Information Fusion, pages 1–8, 2006. (Cited on page 111.)

[Houles 1989] A. Houles and Y. Bar-Shalom. *Multisensor tracking of a maneuvering target in clutter*. IEEE Transactions on Aerospace and Electronic Systems, vol. 25, no. 2, pages 176–189, 1989. (Cited on page 110.)

[Ianniello 1986] J. P. Ianniello. *Large and small error performance limits for multipath time delay estimation*. IEEE Transactions on Accoustic, Speech, and Signal Processing, vol. ASSP-34, pages 245–250, 1986. (Cited on page 36.)

[Julier 2000] S. Julier, J. Uhlmann and H. F. Durrant-Whyte. *A new method for the nonlinear transformation of means and covariances in filters and estimators*. IEEE Transaction on Automatic Control, vol. 45, no. 3, pages 477–482, 2000. (Cited on pages 20 and 96.)

[Karunaratne 2011] B. S. Karunaratne, M. Morelande, B. Moran and S. Howard. *Performance bounds for tracking in a multipath environment*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3652–3655, 2011. (Cited on page 105.)

[Karunaratne 2012a] B. S. Karunaratne, M. R. Morelande and B. Moran. *Bayesian conjugate analysis for multiple phase estimation*. IEEE International Conference on Information Fusion, pages 1927–1934, 2012. (Cited on pages 92 and 107.)

[Karunaratne 2012b] B. S. Karunaratne, M. R. Morelande and Bill Moran. *Target tracking in a multipath environment*. In IET International Conference on Radar Systems (Radar), pages 1–6, Glasgow, UK, 2012. (Cited on pages 94 and 110.)

[Karunaratne 2013] B. S. Karunaratne, M. R. Morelande and B. Moran. *MCMC particle filter for tracking in a partially known multipath environment*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6332–6336, Vancouver, BC, 2013. (Cited on page 110.)

[Kass 1998] R. E. Kass, B. P. Carlin and A. Gelman. *Markov Chain Monte Carlo in practice: A roundtable discussion* . American Statistician, vol. 52, pages 93–100, 1998. (Cited on pages 86 and 97.)

[Kay 1993] Steven M. Kay. Fundamentals of statistical signal processing: estimation theory. Prentice-Hall, Inc., Upper Saddle River, NJ, 1993. (Cited on page 118.)

[Kay 1998] Steven M. Kay. Fundamentals of statistical signal processing: detection theory. Prentice Hall, Inc, Upper Saddle River, NJ, 1998. (Cited on pages 32 and 35.)

[Khan 2005] Z. Khan, T. Balch and F. Dellaert. *MCMC based particle filtering for tracking a variable number of interacting targets*. IEEE Transactions on Pattern Analysis and Machine Learning, vol. 27, 2005. (Cited on page 86.)

[Kirubarajan 1998] T. Kirubarajan, Y. Bar-Shalom, W. D. Blair and G. A. Watson. *IMM-PDAF for radar management and tracking benchmark with ECM*. IEEE Transactions on Aerospace and Electronic Systems, vol. 34, no. 4, pages 1115–1134, 1998. (Cited on page 110.)

[Krolik 2006] J. L. Krolik, J. Farrell and A. Steinhardt. *Exploiting multipath propagation for GMTI in urban environments*. In IEEE Conference on Radar, page 4 pp., 2006. (Cited on pages 83, 84 and 109.)

[Lerner 2002] U. N. Lerner. *Hybrid Bayesian networks for reasoning about complex systems*. PhD thesis, Stanford University, 2002. (Cited on page 20.)

[Levanon 2004] N. Levanon and E. Mozeson. Radar signals. John Wiley and Sons, 2004. (Cited on pages 29, 30, 31, 32 and 33.)

[Li 2005] Bo Li, Thomas Bengtsson and Peter Bickel. *Curse-of-dimensionality revisited: collapse of importance sampling in very large scale systems*. Rapport technique, 2005. (Cited on page 85.)

[Li 2011] L. Li and J. L. Krolik. *Target tracking in uncertain multipath environments using Viterbi data association*. IEEE International Conference on Information Fusion, pages 1–7, 2011. (Cited on pages 83, 84 and 111.)

[Macrae 1974] E. C. Macrae. *Matrix derivatives with an application to an adaptive linear decision problem*. The Annals of Statistics, vol. 2, pages 337–346, 1974. (Cited on page 43.)

[Mahler 2007] R. P. S. Mahler. Statistical multisource-multitarget information fusion. Artech House, 2007. (Cited on page 127.)

[Mardia 1972] K. V. Mardia. Statistics of directional data. Academic Press, 1972. (Cited on pages 76 and 130.)

[Mardia 2010] K. V. Mardia. *Bayesian analysis for bivariate von Mises distributions*. Journal of Applied Statistics, vol. 37, no. 3, pages 515–528, March 2010. (Cited on pages 58 and 64.)

[Mecca 2006] V. F. Mecca, D. Ramakrishnan and J. L. Krolik. *MIMO radar space-time adaptive processing for multipath clutter mitigation*. IEEE Workshop on Sensor Array and Multichannel Processing, pages 249–253, 2006. (Cited on page 83.)

[Metropolis 1953] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth and A. H. Teller. *Equation of state calculations by fast computing machines*. The Journal of Chemical Physics, vol. 21, no. 6, pages 1087–1092, 1953. (Cited on page 91.)

[Morelande 2007] M. R. Morelande, S. Suvarova and B. Moran. *Performance bounds and algorithms for tracking with a radar array*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1105–1108, 2007. (Cited on pages 33, 36, 84 and 109.)

[Morelande 2009] M. R. Morelande and B. Ristic. *Radiological source detection and localisation using Bayesian techniques*. IEEE Transactions Signal Processing, vol. 57, no. 11, pages 4220–4231, 2009. (Cited on page 94.)

[Musso 2001] C. Musso, N. Oudjane and F. Le Gland. *Improving regularized particle filters*. In A. Doucet, N. de Freitas and N. Gordon, editeurs, Sequential Monte Carlo Methods in Practice. New York. Springer-Verlag, 2001. (Cited on page 25.)

[Oh 1991] M. S. Oh. *Monte Carlo integration via importance sampling: dimensionality effect and an adaptive algorithm*. In Statistical Multiple Integration, pages 165–187. American Mathematical Society, 1991. (Cited on page 85.)

[Oudjane 2000] N. Oudjane and C. Musso. *Progressive correction for regularized particle filters*. In IEEE International Conference on Information Fusion, 2000. (Cited on pages 86, 94, 101, 122 and 123.)

[Pezeshki 2007] Ali Pezeshki, Robert Calderbank, Stephen D. Howard and William Moran. *Doppler Resilient Golay Complementary Pairs for Radar*. In SSP '07: Proceedings of the 2007 IEEE/SP 14th Workshop on Statistical Signal Processing, pages 483–487. IEEE Computer Society, August 2007. (Cited on page 32.)

[Pitt 1999] M. K. Pitt and N. Shephard. *Filtering via simulation: auxiliary particle filters*. Journal of the American Statistical Association, vol. 94, no. 446, pages 590–599, 1999. (Cited on pages 22 and 23.)

[Pulford 1998] G. W. Pulford and R. J. Evans. *A multipath data association tracker for over-the-horizon radar*. IEEE Transactions on Aerospace and Electronic Systems, vol. 34, no. 4, pages 1165–1183, 1998. (Cited on pages 109 and 110.)

[Quinn 2011] A. Quinn, J. P. Barbot and P. Larzabal. *The Bayesian inference of phase*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4276–4279, 2011. (Cited on page 58.)

[Rendas 1991] M. J. D. Rendas and J. M. F. Moura. *Cramer-Rao bound for location systems in multipath environments*. IEEE Transactions on Signal Processing, vol. 39, pages 2593–2610, 1991. (Cited on page 36.)

[Richards 2010] M. A. Richards, J. Scheer and W. A. Holm, editeurs. Principles of modern radar: basic principles. SciTech Publishing, 2010. (Cited on page 29.)

[Rigling 2008] B. D. Rigling. *Urban RF multipath mitigation*. IET Radar, Sonar & Navigation, vol. 2, no. 6, pages 419–425, 2008. (Cited on page 83.)

[Rivest, L. P. 1988] Rivest, L. P. *A distribution for dependent unit vectors*. Communications in Statistics-Theory and Methods, vol. 17, no. 2, pages 461–483, 1988. (Cited on page 58.)

[Robert 2004] C. P. Robert and G. Casella. Monte Carlo statistical methods. Springer, 2004. (Cited on pages 19, 58, 66 and 85.)

[Schon 2005] T. Schon, F. Gustafsson and P. J. Nordlund. *Marginalized particle filters for mixed linear/nonlinear state-space models*. IEEE Transactions on Signal Processing, vol. 53, no. 7, pages 2279–2289, 2005. (Cited on page 128.)

[Schuhmacher 2008a] D. Schuhmacher, B. T. Vo and B. N. Vo. *A consistent metric for performance evaluation of multi-object filters*. IEEE Transactions on Signal Processing, vol. 56, no. 8, pages 3447–3457, 2008. (Cited on page 126.)

[Schuhmacher 2008b] D. Schuhmacher, B. T. Vo and B. N. Vo. *On performance evaluation of multi-object filters*. In IEEE International Conference on Information Fusion, 2008. (Cited on page 126.)

[Sen 2007] S. Sen and A. Nehorai. *Slow-time multi-frequency radar for target detection in multipath scenarios*. In IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. (Cited on pages 83 and 84.)

[Sen 2011] S. Sen and A. Nehorai. *Adaptive OFDM radar for target detection in multipath scenarios*. IEEE Transactions on Signal Processing, vol. 59, no. 1, pages 78–90, 2011. (Cited on pages 83 and 84.)

[Silverman 1986] B. W. Silverman. Density estimation for statistics and data analysis. Chapman and Hall/CRC Monographs on Statistics and Applied Probability. Chapman and Hall/CRC, first édition, 1986. (Cited on pages 25, 26, 27, 93, 99 and 120.)

[Singh 2002] H. Singh and V. Hnizdo. *Probabilistic model for two dependent circular variables*. Biometrika, 2002. (Cited on pages 58 and 64.)

[Svensson 2009] L. Svensson, D. Svensson and P. Willett. *Set JPDA algorithm for tracking unordered sets of targets*. In IEEE International Conference on Information Fusion, pages 1187–1194, 2009. (Cited on pages 112, 126 and 127.)

[Tanabe 2007] A. Tanabe, K. Fukumizu, S. Oba, T. Takenouchi and S. Ishii. *Parameter estimation for von Mises–Fisher distributions*. Computational Statistics, vol. 22, no. 1, pages 145–157, March 2007. (Cited on page 66.)

[Tichavsky 1998] P. Tichavsky, C. H. Muravchik and A. Nehorai. *Posterior Cramer-Rao bounds for discrete-time nonlinear filtering*. IEEE Transactions on Signal Processing, vol. 46, no. 5, pages 1386–1396, 1998. (Cited on pages 33, 36 and 43.)

[Van Trees 1968] H. L. Van Trees. Detection, Estimation, and Modulation Theory. John Wiley and Sons, 1968. (Cited on page 33.)

[Vincent 2000] F. Vincent and O. Besson. *Estimating time-varying DOA and Doppler shift in radar array processing*. IEE Proceedings- Radar, Sonar and Navigation, vol. 147, no. 6, pages 285–290, 2000. (Cited on page 41.)

[Wilson 1999] S. L. Wilson and B. D. Carlson. *Radar detection in multipath*. In IEE Proceedings Radar, Sonar and Navigations, pages 45–54, 1999. (Cited on page 85.)

[Zhang 2005] F. Zhang, editeur. The Schur complement and its applications. Springer, 2005. (Cited on page 128.)