PH.D. THESIS

Approximate Nonlinear Filtering with Applications to Navigation

by Babak Azimi-Sadjadi
Advisor: Prof. P.S. Krishnaprasad

PhD 2001-5

ISR develops, applies and teaches advanced methodologies of design and analysis to solve complex, hierarchical, heterogeneous and dynamic problems of engineering technology and systems for industry and government.

ISR is a permanent institute of the University of Maryland, within the Glenn L. Martin Institute of Technology/A. James Clark School of Engineering. It is a National Science Foundation Engineering Research Center.

Web site http://www.isr.umd.edu

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2001		**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Approximate Nonlinear Filtering with Applications to Navigation	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Army Research Office,PO Box 12211,Research Triangle Park,NC,27709**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**130**	

ABSTRACT

Title of Dissertation: Approximate Nonlinear Filtering with Applications to Navigation

Babak Azimi-Sadjadi, Doctor of Philosophy, 2001

Dissertation directed by: Professor P. S. Krishnaprasad
Department of Electrical and Computer Engineering

In this dissertation we address nonlinear techniques in filtering, estimation, and detection that arise in satellite based navigation. Here, we emphasize the theoretical aspect of these techniques, and we also address their applications.

We first introduce particle filtering for an exponential family of densities. We prove that under certain conditions the approximated conditional density converges to the true conditional density. For the case where the conditional density does not lie in an exponential family but stays close to it, we show that under certain assumptions the error of the estimate given by this approximate nonlinear filtering, *projection particle filtering*, is bounded. We give similar results for a family of mixture densities. We use projection particle filtering for an exponential family of densities to estimate the position of a mobile platform that has a combination of inertial navigation system (INS) and global positioning system (GPS),

referred to as an integrated INS/GPS. We show via numerical experiments that projection particle filtering exceeds regular particle filtering methods in navigation performance.

Using carrier phase measurements enables the differential GPS to reach centimeter level accuracy. The phase lock loop of a GPS receiver cannot measure the full cycle part of the carrier phase. This unmeasured part is called *integer ambiguity*, and it should be resolved through other means. Here, we present a new integer ambiguity resolution method. In this method we treat the integer ambiguity as a random digital vector. Using particle filtering, we approximate the conditional probability mass function of the integer ambiguity given the observation. The resolved integer is the MAP estimate of the integer given the observation.

Reliability of a positioning system is of great importance for navigation purposes. Therefore, an integrity monitoring system is an inseparable part of any navigation system. Failures or changes due to malfunctions in sensors and actuators should be detected and repaired to keep the integrity of the system intact. Since in most practical applications, sensors and actuators have nonlinear dynamics, this nonlinearity should be reflected in the corresponding change detection methods. In this dissertation we present a change detection method for nonlinear stochastic systems based on projection particle filtering. The statistic for this method is chosen in such a way that it can be calculated recursively, while the computational complexity of the method remains constant with respect to time. We present some simulation results that show the advantages of this method compared to linearization techniques.

Approximate Nonlinear Filtering with Applications to Navigation

by

Babak Azimi-Sadjadi

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2001

Advisory Committee:

Professor P. S. Krishnaprasad, Chairman
Professor Michael Fu
Professor William S. Levine
Professor Steven I. Marcus
Professor Armand M. Makowski
Professor Prakash Narayan

DEDICATION

To Alejandra for her limitless support and love.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor P.S. Krishnaprasad for his support, his careful criticism of my work, and for being a mentor. He is a person with an insatiable thirst for knowledge, which he imparts to his students. In the years that I have worked with him, I have come to know him as a rigorously ethical person, which made working with him a pleasure.

I would like to thank my wife, Alejandra Mercado, for her help and support throughout my studies in the University of Maryland. I was lucky to have her beside me all the time, not just because she was the right person to discuss a problem with, but because she was a source of motivation when the sky seemed dark.

I thank my committee members for taking the time to read this dissertation, and for their instructive comments during the defense.

I also would like to thank the faculty and friends in this department. Whatever I learned here I owe to them.

Finally, I would like to thank the Army Research Office, the Center for Dynamics and Control of Smart Structures, and the National Science Foundation for supporting this project [1].

[1]This research was supported in part by the Army Research Office under the ODDR&E MURI97 Program Grant No. DAAG55-97-1-0114 to the Center for Dynamics and Control of Smart Structures (through Harvard University), and by the National Science Foundation under the Learning and Intelligent Systems Initiative Grant CMS9720334.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Position estimation is one of the key issues in the automated control of a vehicle. Positioning methods can be classified into three different groups. Dead reckoning is based on piece-wise integration of the speed and heading of the vehicle to calculate the position with respect to a known starting point. Clearly, the estimate of the position gets worse as time goes on, i.e. the error in the estimation accumulates. The Inertial Navigation System (INS) type of positioning is based on Newton's second law. In the INS, the system uses the acceleration and its direction to find (again using integration) the current location. This method is more accurate than the dead-reckoning method, and it can be used in almost all applications. Since the calculation of the current position is based on the integration of the instantaneous acceleration, this method suffers from the same deficiencies as the dead-reckoning method does. The third type of positioning is based on measuring the distance of the unknown location from several known positions. The accuracy of this method depends on the accuracy of the measurement and the accuracy of our knowledge of the location of the known points. Unlike the INS and the dead-reckoning methods, this method does not suffer from the accumulation of error over the duration of

the measurement. In fact, the longer the measurement takes the more accurate the estimate becomes. The Global Positioning System (GPS) measurement is one of the third positioning types.

GPS was conceived as a ranging system from known positions of satellites in space to unknown positions on land, in sea, in air and in space. Effectively, the satellite signal is continuously marked with its own transmission time so that, when received, the signal travel time can be measured by a synchronized receiver. Apart from point positioning, the determination of a vehicle's instantaneous position and velocity, and precise coordination of time were original objectives of GPS [28].

GPS uses "pseudoranges" derived from the broadcast satellite signal. The pseudorange is derived either from measuring the travel time of the coded signal and multiplying it by its velocity or by measuring the phase of the signal. In both cases, the clocks of the receiver and the satellite are employed. Since these clocks are never perfectly synchronized, instead of true ranges pseudoranges are obtained where the synchronization error (denoted as clock error) is taken into account. Consequently, each equation of this type comprises four unknowns: the desired three point coordinates contained in true range, and the clock error. Thus, information from (at least) four satellites is necessary to solve for the four unknowns.

Differential GPS allows the user to obtain a more accurate measurement. It, in fact, allows the removal of a good portion of the positioning error from the estimation. This, along with other new technology, allows the users to use the carrier phase as part of the positioning information. This can increase the accuracy of the estimation to centimeter, or in the static case, to millimeter levels.

Unlike the applications in communication, in positioning one needs to know the exact phase difference between the received signal and the transmitted signal,

i.e. the exact number of full cycles and the portion of the phase that is less than a full cycle is needed for position estimation. A Phase Lock Loop (PLL) can provide a very accurate estimate of the portion of the phase that is less than a full cycle. Also, it can count the number of full cycles added to the phase once it starts tracking the signal continuously. The initial value of the full cycles, though is not known, therefore, the phase lock loop cannot provide that part of the phase information. This part, which is a constant integer number of full cycles, is called *integer ambiguity* and should be resolved through numerical methods [26, 28, 51, 52].

Although carrier phase differential GPS allows for very accurate positioning, it is very sensitive to obstacles that can block satellite signals. The loss in signal could be for a few moments or for a longer period of time. If the loss in signal is sufficiently short, the phase lock loop is unable to detect the loss in signal, therefore it is not able to record the added full cycles to the measured phase. This results in a jump in the measured phase. This phenomena is known as *cycle slip*. Any navigation method that uses carrier phase differential GPS should be able to detect and isolate the cycle slips whenever they occur.

If the loss in signal is for a sufficiently long period of time, so that position information is needed while the GPS receiver has signals only from three or fewer satellites, the positioning techniques that are solely based on satellite navigation fail to function. In such cases the user should use other methods to be able to receive continuous position information [1, 16, 19, 20, 42, 56].

Integration of INS with GPS has proven to be robust and accurate. In an integrated INS/GPS, INS provides positioning information that is calibrated by GPS.

In most applications, such as integrated INS/GPS, or dead-reckoning/GPS, or vehicle dynamic/GPS, linearization of the dynamic and the GPS observation is the main tool for estimation [20, 21, 42, 44, 45]. It can be shown [16] that when the number of satellites is below a certain number or the geometry of the satellite constellation is near singular, the Extended Kalman Filter (EKF) diverges and fails to provide accurate estimation of the position. In this case, it is important to use nonlinear filtering for the estimation problem.

The results in [16] were a motivation for us to study nonlinear filtering, estimation, and detection methods and their applications to satellite based navigation. In this dissertation we are interested both in the theory and the application of such methods. For this reason our intention is to discover new tractable finite approximation methods for nonlinear filtering problems. We are specially interested in the approximation methods that are suited for satellite based navigation.

Our contribution in this dissertation can be categorized into three major areas. In all of these three areas we have developed the theory of the proposed approximation methods as well as the relevant application to navigation. In the rest of this chapter we introduce these three areas.

1.1 Approximate Nonlinear Filtering

Unlike the linear Gaussian case, no finite dimensional filtering method for general nonlinear systems exists. The most well known approximation method for nonlinear filters, Extended Kalman Filtering (EKF), is merely an *ad hoc* method [46]. The performance of EKF depends on the specific application and it is not guaranteed.

Projection filtering is another approximation method for nonlinear filtering [9, 11, 12]. The main assumption in projection filtering is that the conditional density of the state given the observations can be projected onto a family of densities without significant error. In [11] the conditional density is projected onto an exponential family of densities. Since the exponential family has a finite dimensional parametric representation, the projected nonlinear filter also has a finite dimensional form.

In a different approach [9], the conditional density of the system given the observations is approximated by a summation of basis functions. Then, a Galerkin approximation method is used to propagate the coefficients of the approximated density.

Although both methods in [9] and [11] provide better approximation methods than EKF, the convergence of the approximated conditional density to the actual conditional density is not studied [1].

An entirely different approach for approximating the conditional density is simulation based filtering. Grid-less simulation based filtering, now known by many different names such as *particle filtering* [34, 40], the Condensation Algorithm [29], the Sequential Monte Carlo (SMC) Method [22], and Bayesian Bootstrap Filtering [24], was first introduced in [24] and then it was rediscovered independently in [29] and [32]. Henceforth we refer to this filtering method as particle filtering. The results in [24] are the extension of the results in [48] and [2] to the dynamic case and is based on a method called Sampling/Importance Resampling (SIR). SIR is key element of the grid-less simulation based filtering methods. SIR allows these

[1] In [9] a convergence proof is reported but in a remark the authors note that: "The requirement in the hypothesis of Theorem 1 is somewhat unsatisfactory because it is not clear at this stage how to guarantee that this is true a priori".

methods to have automatically high resolution grids in areas where the conditional density is significant and low resolution in the areas where the conditional density is small.

Particle filtering is a Monte Carlo based method for nonlinear filtering. The particles in this method refer to independent samples generated with the Monte Carlo method. In [40] it was shown that the optimal nonlinear filter can be approximated with an arbitrarily small error by a finite dimensional filter. The problem of this method is that for high dimensional systems, and for small errors, computational complexity grows, and the method is not always implementable in real time applications. The other shortcoming of particle filtering is its vulnerability to sample impoverishment [15], so that the particle distribution gives a poor approximation of the required conditional density. In extreme cases, after a sequence of updates the particle system can collapse to a single point. In less extreme cases, although several particles may survive, there is so much internal correlation that summary statistics behave as if they are derived from a substantially smaller sample. To compensate, large numbers of particles are required in realistic problems [15].

In the cases where we have some prior information about the distribution, we should expect to achieve higher performance if we take this information into account. By higher performance, we mean a reduction in the computational cost and an increase in the convergence rate. Here we assume that the conditional distribution has a density in an exponential family of densities, or at least stays close to it in a sense that we will define. Using this assumption, we replace the empirical distribution in [40] with the Maximum Likelihood Estimate (MLE) of the parameters of an exponential density. We call this new method *projection particle*

filtering. In Theorem 4.1.6 we show that if the conditional density of the state given the observations lies in an exponential family of densities then the estimated conditional density converges to the true conditional density in a sense that will be defined. In Theorem 4.2.7 for the case where the true conditional density stays close to an exponential family of densities we show that the error of the estimate given by projection particle filtering is bounded.

As stated in [11], finding the proper exponential family of densities for a dynamical system is quite challenging. To overcome this problem and motivated by the results in Theorems 4.1.6 and 4.2.7, we studied projection filtering for a family of mixture densities. In this case, we also show that if the family of mixture densities is close (in a sense that will be defined later) to the true conditional density, the error of estimate given by approximate filtering is bounded.

One of the applications of projection particle filtering is position estimation for an integrated INS/GPS. We are particularly interested in the cases where linearization methods fail. One such case is when the number of GPS satellites in view is below a critical number (for three dimensional positioning, this critical number is four). We demonstrate numerically that in this situation the position estimation given by the EKF diverges, while the approximate nonlinear filtering methods provide a reasonable estimate of the position. We also show via numerical results that the performance of the projection particle filter exceeds the performance of the particle filter for the same number of particles.

1.2 Integer Ambiguity Resolution

Integer ambiguity resolution methods are an inseparable part of positioning techniques that use carrier phase differential GPS as part of their measurement.

The available integer ambiguity resolution methods are mostly based on a rough estimate of the integer ambiguity and a search method to find the correct integer value [3]. In the case of the Ambiguity Function Method (AFM), since the integer ambiguity cancels out as a result of the cosine function in the ambiguity function, the search is done over the position grid [38, 47]. In the least square ambiguity search technique, first the `float` solution for integer ambiguity is found by minimizing the square of the error associated to the position and the integer estimate. If the covariance matrix of the error for these estimates of the integer ambiguity is diagonal, the best integer vector that minimizes the error is the nearest integer vector, but usually this is not the case. Therefore, the correct solution is found by searching the area near the `float` solution [27]. The size of this area depends on the covariance matrix and the size of the integer vector, i.e. the number of satellites. In the Least-squares AMBiguity Decorrelation Approach (LAMBDA), a linear transformation of the GPS observables that maps integer vectors to integer vectors, is chosen in such a way that the transformed covariance matrix is dominantly diagonal [52]. This transformation helps to reduce the size of the search space. Variations of these methods have been used. For example, in [26] a Kalman filter is used to estimate the `float` least square estimation of the integer ambiguity and the same type of decorrelation is applied to the observable to reduce the size of the search space.

In most of these methods the integer ambiguity is treated as an unknown integer vector. In this dissertation we present a new method that treats the integer ambiguity as a random integer vector. Inspired by our results in Theorems 4.1.6 and 4.2.7, we present a method for approximating the conditional probability mass function (pmf) of this integer vector given the observations. The estimate of the

integer value then is simply the point that maximizes the pmf. In this method, similar to the projection particle filtering method, we find a family of exponential distributions that is close to the true pmf. The integer ambiguity is then resolved through the estimation of the parameter of the family.

1.3 Detection of Abrupt Changes in Nonlinear Dynamical Systems

In [43] the change detection problem is stated as follows:

"Whenever observations are taken in order it can happen that the whole set of observations can be divided into subsets, each of which can be regarded as a random sample from a common distribution, each subset corresponding to a different parameter value of the distribution. The problems to be considered in this paper are concerned with the identification of the subsamples and the detection of changes in the parameter value".

We refer to a change or an abrupt change as any change in the parameters of the system that happens either instantaneously, or much faster than any change that the nominal bandwidth of the system allows.

The key difficulty of all change detection methods is that of detecting intrinsic changes that are not necessarily directly observed but are measured together with other types of perturbations [8].

The change detection could be off-line or on-line. In on-line change detection, we are only interested in detecting the fact that a change happened. In this case, we are only interested in detecting the change as quickly as possible (for example, to minimize the detection delay with fixed mean time between false alarms), and

the estimate of the time when the change occurs is not of importance. In off-line change detection, we assume that the whole observation sequence is available at once. In this case, the estimate of the time of change could be one of the goals of the detection method. In this dissertation we limit our concern to on-line detection of abrupt changes.

The change detection methods that are studied in this dissertation can be classified under the general name of Likelihood Ratio methods. CUmulative SUM (CUSUM) and Generalized Likelihood Ratio (GLR) tests are among these methods. CUSUM was first proposed in [43]. The most basic CUSUM algorithm assumes that the observation signal is a sequence of stochastic variables which are independent and identically distributed with known common probability density function before the change time, and independent and identically distributed with another known probability density after the change time. In the CUSUM algorithm the log-likelihood ratio for the observation from time i to time k is calculated and its difference with its current minimum is compared with a certain threshold. If this difference exceeds the threshold an alarm is issued.

Properties of the CUSUM algorithm have been studied extensively. The most important property of the CUSUM algorithm is its asymptotic optimality, which was first proven in [37]. More precisely, CUSUM is optimal, with respect to the worst mean delay, when the mean time between false alarms goes to infinity. This asymptotic point of view is convenient in practice, because a low rate of false alarms is always desirable.

In the case of unknown system parameters after change, the GLR algorithm can be used as a generalization of the CUSUM algorithm. Since in this algorithm the exact information of the change pattern is not known, the likelihood ratio is

maximized over all possible change patterns [2].

For stochastic systems with linear dynamics and linear observations, the observation sequence is not independent and identically distributed. Therefore, the regular CUSUM algorithm cannot be applied for detection of changes in such systems. However, if such systems are driven by Gaussian noise, the innovation process associated with the system can be generated. This process is known to be a sequence of independent random variables. The regular CUSUM algorithm or its more general counterpart, GLR, can be applied to this innovation process [8, 55].

In this dissertation we are interested in the change detection problem for stochastic systems with nonlinear dynamics and observations. We show that for such systems, the complexity of the CUSUM algorithm grows with respect to time. This growth in complexity cannot be tolerated in practical problems. Therefore, instead of the statistic used in the CUSUM algorithm we introduce an alternative statistic. We show that with this statistic, the calculation of the likelihood ratio can be done recursively and the computational complexity of the method stays constant with respect to time. This new method is used for the cycle slip detection for an integrated INS/GPS.

1.4 Dissertation Outline

In Chapter 2 we briefly review the GPS signal structure and explain different GPS observables. Chapter 3 is devoted to the review of different approximate nonlinear filtering methods as well as a statement of the general nonlinear filtering framework. In Chapter 4 we present our main results on projection particle filtering for an exponential family of densities. Chapter 5 addresses the applications of

[2] If the maximum does not exist, the supremum of the likelihood ratio should be calculated.

the results in Chapter 4 to position estimation for an integrated INS/GPS under critical conditions. In Chapter 6 we present our results on projection particle filtering on a family of mixture densities. We introduce our new integer ambiguity resolution method based on projection particle filtering in Chapter 7. In Chapter 8 we present our results in change detection for nonlinear systems and its application to cycle slip detection for an integrated INS/GPS. Finally, in Chapter 9 we state conclusions and an outline of future work.

Chapter 2

A Short Review of GPS

The NAVSTAR (Navigation Satellite Timing and Ranging) GPS is a satellite based, worldwide, all weather navigation system. This system provides accurate positioning for a receiver that is capable of receiving signals from at least four satellites [28].

The main part of the GPS signal is a coded message that is simply a clock signal. This coded message and the time that this message was sent is completely known by the receiver. The receiver measures the time when this signal is received and from that measures the travel time and, consequently, the distance between the receiver and the corresponding satellite. Since the clocks in the satellites and in the receiver are never synchronized the measured distance is not the true range. In GPS literature this distance is called pseudorange.

All generated signals including the carrier in GPS are synchronized with the main atomic clock, therefore the carrier phase (if known completely) can also be used as a ranging signal.

The accuracy of the positioning depends on many factors including the type of user, the quality of the receiver, and the positioning technique. The U.S. De-

partment of Defense deliberately adds uncertainty to the positioning signal; for civilians this is one of the major sources of error. There are other sources of error that degrade the position accuracy. These sources include, ionospheric and tropospheric delay, satellite position uncertainty, satellite and receiver clock bias, multipath, and the usual channel noise [44]. In a relatively small area, for example distances of less that 100 Km, some of these errors are highly correlated [45]. The uncertainty added by the military, satellite clock bias [1], and satellite position uncertainty are clearly the same for all users that are using the same satellite. The tropospheric and the ionospheric delays are also highly correlated in short distances. By locating a receiver in a known position one can estimate the common errors and send the correction signal to the other users. This idea caused a revolution in satellite aided radio positioning. This technique is called differential GPS, and is widely used for surveying as well as real time navigation [45].

Today's technology allows the use of the carrier as part of the navigation information. Due to the periodic nature of the carrier, one can only measure the phase of the carrier, modulo 2π, i.e. the PLL can never measure the exact phase. The unknown part of the phase is known to be an integer number times 2π. Since the measurement noise for the carrier is much smaller than the measurement noise for the clock signal, it is essential that we should estimate the exact phase of the signal. It is shown that in the case of differential GPS, the receiver can estimate the ambiguity in the integer number of unmeasurable cycles. This method is called carrier phased differential GPS.

[1] After May 2000 the US department of defense eliminated this uncertainty for all users. The satellite position accuracy though, is higher for military users.

2.1 GPS Signal Structure

The GPS signal consists of a clock signal and a navigation message that are amplitude modulated. Using the clock signal and navigation message, one can estimate its position, if at least four satellites are in view.

Each satellite sends the clock signal in two different bands, L_1 and L_2. These signals are as follows [50]:

$$L_1^i(t) = a_1 P^i(t) D^i(t) cos(2\pi f_1 t) + b_1 C/A^i(t) D(t) sin(f_1 t)$$

$$L_2^i(t) = a_2 P^i(t) D^i(t) cos(2\pi f_2 t)$$

Where:

- i: Number of the satellite.
- $P^i(t)$: Precise clock signal generated with a random number generator with frequency 10.23 *MHz* and a period of 38 weeks. Each satellite has its unique code.
- C/A^i: Course acquisition code, the clock for non-military positioning generated with frequency 1.023 *MHz* and a period of 1 *ms*.
- $D^i(t)$: Navigation data with a bit rate of 50 *bit/sec*.
- f_1: Carrier of L_1, $f_1 = 154 * 10.23$ *MHz* synchronized with the central clock.
- f_2: Carrier of L_2, $f_2 = 120 * 10.23$ *MHz* synchronized with the central clock.
- a_1,b_1,a_2: Amplitudes of the carriers.

The GPS receiver, receives the signal corrupted by noise and other sources of error. The raw measurements of the code and the carrier phase can be presented as follows [30]:

$$P^i(t_k) = \rho^i(t_k) + c[dT(t_k) - dt^i(t_k)] + T^i(t_k) + I^i(t_k) + E^i(t_k) + \epsilon^i(t_k)$$

$$\lambda\Phi^i(t_k) = \rho^i(t_k) + c[dT(t_k) - dt^i(t_k)] + T^i(t_k) - I^i(t_k) + E^i(t_k) - \lambda N^i + \eta^i(t_k)$$

where

- t_k : GPS time at epoch k.
- P : code observation (m).
- i : satellite number.
- ρ : distance between the moving object and the satellite position (m).
- c : speed of light (m/s).
- dT : receiver clock bias (s).
- dt : satellite clock bias including Selective Availability (SA) clock error (s).
- E : effect of ephemeris error including SA orbit error (m).
- I : ionospheric delay (m).
- T : tropospheric delay (m).
- ϵ: code observation noise (m).
- λ : carrier wavelength (m).
- Φ: carrier phase observation $(cycles)$.

- N : integer ambiguity (*cycles*).
- η : carrier observation noise (m).

Access to the above observations depends on the type of user and the quality of the receiver.

2.2 Single and Double Differencing in GPS

State of the art receivers can have access to code and carrier phase measurement of 12 satellites in 2 frequencies. In this kind of receivers a big portion of the ionospheric delay can be corrected and removed [28, 33]. Since the receiver clock bias is the same for the observation from all satellites, the error due to the receiver clock bias can be completely removed by *single differencing*. In single differencing, the receiver subtracts code and/or phase measurement of one satellite from the others [53]. Single differencing eliminates a major source of error. If it is possible to mount a GPS receiver in a known location (i.e. base), one can use the *double differencing* method to eliminate other sources of error. Within short distances, ionospheric and tropospheric errors are highly correlated, and can be eliminated by making a difference between the code and the carrier phase measurement of the base and the moving receiver. The appropriate length scale for this is not very clear and it depends on sunspot activity [33]. When the activity is low, the short distance could cover larger areas and conversely. It can be shown that double differencing reduces ephemeris error by a factor of d/r [53], where d and r are the distances from the moving object to the base and to the satellite, respectively. Using the operator $(\cdot)_{i,j}^{k,l} = [(\cdot)_i^k - (\cdot)_j^k] - [(\cdot)_i^l - (\cdot)_j^l]$, where i and j are indices for the receivers and l and k are indices for satellites. Then double differencing can

be written as follows:

$$P_{i,j}^{k,l} = \rho_{i,j}^{k,l} + \epsilon_{i,j}^{k,l}, \tag{2.1}$$

and,

$$\lambda\Phi_{i,j}^{k,l} = \rho_{i,j}^{k,l} + \lambda N_{i,j}^{k,l} + \eta_{i,j}^{k,l}. \tag{2.2}$$

In the above formula the time index is not shown for simplicity. If the short baseline assumption is not applied, the canceled term will show up in the double difference observer and should be estimated [53]. Sometimes there are not enough observations to estimate all of these terms. In this case,we are forced to consider these terms as noise terms. Multi-path is another source of error that cannot be removed from the observation (2.1) and (2.2) [7].

Unlike other terms, if no cycle slip occurs, the integer ambiguity is constant with respect to time. The fact that the integer ambiguity is constant in time is very important, in fact, all integer ambiguity resolution methods rely on this property. Once this integer number is known, the phase measurement can be used for positioning. We should remember that although equation (2.1) does not have integer ambiguity in it, still the energy of the noise, $\epsilon_{i,j}^{k,l}$, is an order of magnitude higher than the energy of the noise in (2.2) [44]. Therefore, the integer ambiguity problem remains intact.

Although double differencing eliminates many sources of error, it is not necessarily the best way of handling the measurement. Double differencing reduces the number of observation equations which may not be the optimum choice for certain applications. Therefore, if a good model for a specific error exists, we can use this model to estimate the error instead of eliminating it through the double differencing operation. As we mentioned earlier single differencing eliminates the

error due to receiver clock bias by subtracting the phase/code measurement of one satellite from the others, but this reduces the number of measurement equations. We can remove this part from the double differencing operation, i.e. we can only subtract the measurements of one receiver from the base (the receiver in a known location). In this case, we can eliminate a good portion of the error due to ionospheric, tropospheric, ephemeris, and satellite clock bias. The receiver clock bias, dT, can be modeled by a second order system driven by a Brownian motion process. In Chapter 5 we use this model for estimating the position of the receiver as well as the clock bias for an integrated INS/GPS.

2.3 Cycle Slip in Carrier Phase Measurement

Carrier phase measurement enables a GPS receiver to reach centimeter level accuracy. This is true only if the exact phase is measured. In addition to this, the receiver should track the phase at all times to be able to measure the exact phase. This task is done by the PLL built in the receiver.

When the receiver is turned on, the fraction of the phase (i.e. the difference between the satellite transmitted carrier and a receiver generated replica signal) is observed and an integer counter initialized. During tracking, the counter is incremented by one whenever the fractional phase changes from 2π to 0. Thus, at a given time the observed accumulated phase is 2π times the sum of the fractional phase and the integer count. The initial integer number of full cycles between the satellite and the receiver is unknown. This phase ambiguity remains constant as long as no loss of signal lock occurs. If a loss occurs, the integer counter is reinitialized which causes a jump in the instantaneous accumulated phase by an integer number of cycles. This jump is called a cycle slip which, of course, is

restricted to phase measurements [28].

Three sources of cycle slips can be distinguished. First, cycle slips are caused by obstruction of the satellite signal due to trees, buildings, bridges, etc. This is the most frequent source of cycle slip. The second source for cycle slips is a low signal to noise ratio due to bad ionospheric conditions, multipath, high receiver dynamics, or low satellite elevation. A third source is a failure in the receiver software which leads to incorrect signal processing. Cycle slips could also be caused by malfunctioning satellite oscillators, but these are rare [28].

Cycle slip detection is a very important part of a navigation system that is based on carrier phase GPS. If a cycle slip is not detected correctly the position given by the navigation system is not reliable. In Chapter 8 we propose a new method that has the potential ability of cycle slip detection even under conditions when the number of satellites is below a critical number.

Chapter 3

Nonlinear Filtering: An Introduction

Filtering problems consist of "estimating" a process $\{\mathbf{x}_t\}$ (or a function of it) given the related process, $\{\mathbf{y}_t\}$, which can be observed [18]. The observation is available in an interval, i.e., $\{\mathbf{y}_s, 0 \leq s < t\}$ and the function of the state is estimated at time t. Except for the linear Guassian system and very special cases in nonlinear settings, estimating the state given the observations results in an infinite dimensional filter [46]. Therefore, approximation methods of finite dimension are very appealing.

The most widely used approximate filtering method is the extended Kalman filter, which is a heuristic approach based on linearization of the state dynamics and the observation near the nominal path [46]. EKF is computationally simple but, the convergence of the estimated conditional density to the actual conditional density is not guaranteed.

Projection filtering is another approximation method [9, 11, 12, 13]. In projection filtering it is assumed that the conditional density of the state of the system

can be approximated by a member of a parametric family of densities. In this case, estimating the conditional density is equivalent to estimating the parameter of the family. In [11] the exponential family of densities is chosen as the parametric family. In contrast, the approach in [9] employs a Galerkin approximation to solve the Fokker-Planck equation [46], between measurement epochs.

Particle filtering is an approximation method for nonlinear filtering and it is based on the Monte Carlo method; in this method, the particles at time t_i are i.i.d. random vectors that are distributed according to the empirical conditional distribution of the state, given the observations up to time t_i. These particle/state vectors are used in the state equation to find the values of particles at time t_{i+1}. Then at time t_{i+1}, the empirical distribution is evaluated according to the values of the particles. The new observation at time t_{i+1} is taken into account through Bayes' Rule to calculate the conditional empirical distribution, this process is then repeated. In [40] it is proved that by tracking a large enough number of particles, one can get an approximate conditional distribution that is arbitrarily close to the true conditional distribution.

3.1 Problem Setup

We assume that all stochastic processes are defined on a fixed probability space (Ω, F, P), and a finite time interval, $[0, T]$, on which there is defined an increasing family of σ-fields, $\{\mathcal{F}_t, 0 \leq t \leq T\}$. It is assumed that each process, $\{\mathbf{x}_t\}$, is adapted to $\mathcal{F}_t$, i.e., $\{\mathbf{x}_t\}$ is $\mathcal{F}_t$-measurable for all t. We assume that $\{\mathbf{x}_t\}$ is a vector diffusion process of the form

$$\mathbf{x}_t = \mathbf{x}_0 + \int_0^t \mathbf{f}_s(\mathbf{x}_s)ds + \int_0^t G_s(\mathbf{x}_s)d\mathbf{w}_s, \tag{3.1}$$

where $\mathbf{x}_t \in \mathcal{R}^n$, and $\mathbf{w}_t \in \mathcal{R}^q$ is a vector from an independent Brownian motion process; the second integral is in the Ito sense [49], and the function $\mathbf{f}_t(\cdot)$ and the matrix $G_t(\cdot)$ have the proper dimensions. The observation, $\mathbf{y}_t$, is a discrete time process given as follows:

$$\mathbf{y}_{n\tau} = \mathbf{h}_n(\mathbf{x}_{n\tau}) + \mathbf{v}_n, \tag{3.2}$$

where $\mathbf{y}_{n\tau} \in \mathcal{R}^d$, and $\mathbf{v}_n \in \mathcal{R}^d$ is a discrete time white Gaussian noise process with zero mean and known covariance matrix. The state dynamics and observation equations can be rewritten formally as follows:

$$\begin{aligned} d\mathbf{x}_t = &\quad \mathbf{f}_t(\mathbf{x}_t)dt + G_t(\mathbf{x}_t)d\mathbf{w}_t, \quad \text{given the distribution of } \mathbf{x}_0 \\ \mathbf{y}_{n\tau} = &\quad \mathbf{h}_n(\mathbf{x}_{n\tau}) + \mathbf{v}_n \end{aligned} \tag{3.3}$$

The noise processes $\{\mathbf{w}_t,\ t \geq 0\}$, and $\{\mathbf{v}_n,\ n = 0, 1, \cdots\}$, and the initial condition $\mathbf{x}_0$ are assumed to be independent. We use Q_t and R_n for the covariance matrices of the processes $\mathbf{w}_t$ and $\mathbf{v}_n$, respectively. We assume that R_n is invertible for all n's. We have the following additional assumptions [25]:

A 3.1.1 *[local Lipschitz continuity]* $\forall\ \mathbf{x},\ \mathbf{x}' \in B_r$ *and* $t \in [0, T]$*, where* B_r *is a ball of radius* r*, we have*

$$\begin{aligned} \|\mathbf{f}_t(\mathbf{x}) - \mathbf{f}_t(\mathbf{x}')\| &\leq k_r\|\mathbf{x} - \mathbf{x}'\|, \quad \textit{and} \\ \|G_t(\mathbf{x})Q_tG_t^T(\mathbf{x}) - G_t(\mathbf{x}')Q_tG_t^T(\mathbf{x}')\| &\leq k_r\|\mathbf{x} - \mathbf{x}'\|. \end{aligned} \tag{3.4}$$

A 3.1.2 *[Non-Explosion] There exists* $k > 0$ *such that*

$$\begin{aligned} \mathbf{x}^T\mathbf{f}_t(\mathbf{x}) &\leq k(1 + \|\mathbf{x}\|^2), \quad \textit{and} \\ \textit{trace}(G_t(\mathbf{x})Q_tG_t^T(\mathbf{x})) &\leq k(1 + \|\mathbf{x}\|^2). \end{aligned} \tag{3.5}$$

$\forall\ t \in [0, T]$ *and* $\forall\ \mathbf{x} \in \mathcal{R}^n$.

Under Assumptions (A3.1.1) and (A3.1.2), there exists a unique solution $\{\mathbf{x}_t, t \in [0,T]\}$ to the state equation, and $\mathbf{x}_t$ has finite moment of any order [25].

In addition to these, we assume that the probability distribution of the state $\mathbf{x}_t$, given the observation up to time t, $\pi_t(d\mathbf{x}) = P(\mathbf{x}_t \in d\mathbf{x}|\mathbf{y}^t)$, where $\mathbf{y}^t = \{\mathbf{y}_n,\ i = 1, \cdots, n,\ n\tau < t\}$, has a density p_t with respect to the Lebesgue measure on $\mathcal{R}^n$. Then $\{p_t,\ t > 0\}$ satisfies the following partial differential equation and updating equations [11]:

$$\begin{aligned} \tfrac{\partial}{\partial t} p_t = &\quad \mathcal{L}_t^* p_t \qquad\qquad n\tau \le t < (n+1)\tau, \quad \text{and} \\ p_{n\tau} = &\quad c_n \Psi_n p_{n\tau^-} \end{aligned} \tag{3.6}$$

where

$$\begin{aligned} \mathcal{L}_t^*(\Phi) &= -\textstyle\sum_{i=1}^n \frac{\partial}{\partial \mathbf{x}_i}[f_t^i \Phi] + \frac{1}{2}\sum_{i,j=1}^n \frac{\partial^2}{\partial \mathbf{x}_i \partial \mathbf{x}_j}[a_t^{ij}\Phi], \\ [a_t^{ij}] &= G_t Q_t G_t^T, \\ \Psi_n(\mathbf{x}) &\triangleq \exp\left(-\tfrac{1}{2}(\mathbf{y}_{n\tau} - \mathbf{h}_n(\mathbf{x}))^T R_n^{-1}(\mathbf{y}_{n\tau} - \mathbf{h}_n(\mathbf{x}))\right), \end{aligned}$$

and c_n is a normalizing factor.

Except for the linear Gaussian case, and some very special nonlinear cases, solving System (3.6) constitutes an infinite dimensional filter [46]. Therefore, for practical problems it is necessary to approximate the conditional density in (3.6). In the next section, we discuss one of these approximation methods.

3.2 Projection Filtering on Exponential Families of Densities

This section is mainly a review of the results we use from [11]. We start this section with the definition of the exponential family of densities.

Definition 3.2.1 *Let $\{c_1, \cdots, c_p\}$ be affinely independent* [1] *scalar functions defined on $\mathcal{R}^n$, and assume that the convex set*

$$\Theta_0 = \left\{ \theta \in \mathcal{R}^p : \ \Upsilon(\theta) = log \int exp\left(\theta^T \mathbf{c}(\mathbf{x})\right) d\mathbf{x} < \infty \right\},$$

has nonempty interior. Then,

$$\mathcal{S} = \{p(\cdot, \theta), \ \theta \in \Theta\}$$
$$p(\mathbf{x}, \theta) := exp\left[\theta^T \mathbf{c}(\mathbf{x}) - \Upsilon(\theta)\right],$$

where $\Theta \subseteq \Theta_0$ is open, is called an exponential family of probability densities.

We denote by $\mathcal{S}^{\frac{1}{2}}$ the space of square roots of the densities in $\mathcal{S}$, i.e., $\mathcal{S}^{\frac{1}{2}} = \{\sqrt{p(\cdot,\theta)};\ \theta \in \Theta\}$. If $p(\cdot,\theta) \in \mathcal{S}$, then $\sqrt{p(\cdot,\theta)} \in L_2$. The functions $\frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_i}$, $i = 1, \cdots, p$ form a basis for the tangent vector space at $\sqrt{p(\cdot,\theta)}$ to the space $\mathcal{S}^{\frac{1}{2}}$, i.e., the tangent space at $\sqrt{p(\cdot,\theta)}$ is given by [4]:

$$L_{\sqrt{p(\cdot,\theta)}}\mathcal{S}^{\frac{1}{2}} = span\left\{ \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_1}, \cdots, \ \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_p} \right\}. \tag{3.7}$$

The inner product of any two basis elements is defined as follows

$$\begin{aligned} \left\langle \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_i}, \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_j} \right\rangle &= \frac{1}{4} \int \frac{1}{p(\mathbf{x},\theta)} \frac{\partial p(\mathbf{x},\theta)}{\partial \theta_i} \frac{\partial p(\mathbf{x},\theta)}{\partial \theta_j} d\mathbf{x} \\ &= \frac{1}{4} g_{ij}(\theta) \end{aligned} \tag{3.8}$$

It can be easily seen that $g(\theta) = (g_{ij}(\theta)) = (E[c_i c_j] - E[c_i]E[c_j])$ is the Fisher information matrix of $p(\cdot,\theta)$.

Any member of L_2 can be projected to the tangent space $L_{\sqrt{p(\cdot,\theta)}}\mathcal{S}^{\frac{1}{2}}$ according to the following projection formula

$$\begin{aligned} \Pi_\theta : \ L_2 \supseteq V &\rightarrow L_{\sqrt{p(\cdot,\theta)}}\mathcal{S}^{\frac{1}{2}} \\ v &\rightarrow \sum_{i=1}^{p} \sum_{j=1}^{p} 4g^{ij}(\theta) \left\langle v, \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_j} \right\rangle \frac{1}{2\sqrt{p(\cdot,\theta)}} \frac{\partial p(\cdot,\theta)}{\partial \theta_i}. \end{aligned} \tag{3.9}$$

[1] $\{c_1, \cdots, c_p\}$ are affinely independent if for distinct points $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{p+1}$, $\sum_{i=1}^{p+1} \lambda_i \mathbf{c}(\mathbf{x}_i) = 0$ and $\sum_{i=1}^{p+1} \lambda_i = 0$ implies $\lambda_1 = \lambda_2 = \cdots = \lambda_{p+1} = 0$ [17].

Projection filtering seeks a solution p_t for (3.6) that lies in $\mathcal{S}$. Of course, this solution is only an exponential density, but we hope, by choosing the proper family, to keep the approximation error small (in the L_2 sense).

If we consider the square root of the density in (3.6), we get

$$\frac{\partial\sqrt{p_t}}{\partial t} = \frac{1}{2\sqrt{p_t}}\frac{\partial p_t}{\partial t} = \frac{1}{2\sqrt{p_t}}\mathcal{L}_t^* p_t. \tag{3.10}$$

Define $\alpha_{t,\theta} = \frac{\mathcal{L}_t^* p_t(\cdot,\theta)}{p_t(\cdot,\theta)}$. We assume that for all $\theta \in \Theta$ and all $t \geq 0$, $E_{p(\cdot,\theta)}\{|\alpha_{t,\theta}|^2\} < \infty$, which implies that $\frac{\mathcal{L}_t^* p_t(\cdot,\theta)}{\sqrt{p_t(\cdot,\theta)}}$ is a vector in L_2 for all $\theta \in \Theta$ and all $t \geq 0$ [11].

Now assume that in equation (3.10), for $\{\sqrt{p_t},\ t \geq t_0\}$, starting at time $n\tau$ from the initial condition, $\sqrt{p_{n\tau}} = \sqrt{p(\cdot,\theta_{n\tau})} \in \mathcal{S}^{\frac{1}{2}}$ for some $\theta_{n\tau} \in \Theta$. Under these assumptions, the right hand side of (3.10) is in L_2, which can be projected into the finite dimensional tangent vector space $L_{\sqrt{p(\cdot,\theta_{n\tau})}}\mathcal{S}^{\frac{1}{2}}$. The propagation part of the projection filter for the exponential family, $\mathcal{S}$, in the interval $[n\tau,(n+1)\tau)$, is defined as the solution to the following differential equation in the same interval:

$$\frac{\partial\sqrt{p_t(\cdot,\theta_t)}}{\partial t} = \Pi_{\theta_t}\frac{\mathcal{L}_t^* p_t(\cdot,\theta_t)}{2\sqrt{p_t(\cdot,\theta_t)}} \quad . \tag{3.11}$$

We also assume that $\mathbf{h}_n(\mathbf{x})$ in equation (3.2) is time invariant, i.e., $\mathbf{h}_n(\mathbf{x}) = \mathbf{h}(\mathbf{x})$, and the components of $\mathbf{h}(\mathbf{x})$, $h^i(\mathbf{x})$, and $\|\mathbf{h}(\mathbf{x})\|^2_{R^{-1}}$ are linear combinations of $c_i(\mathbf{x}),\ i = 1,\cdots,\ p$:

$$\frac{1}{2}\|\mathbf{h}(\mathbf{x})\|^2_{R^{-1}} = \sum_{i=1}^{p}\lambda_i^0 c_i(\mathbf{x}) \quad \text{and} \quad h^k(\mathbf{x}) = \sum_{i=1}^{p}\lambda_i^k c_i(\mathbf{x}),\quad k = 1,\cdots,\ d \tag{3.12}$$

where $\|\mathbf{x}\|_A = \sqrt{\mathbf{x}^T A\mathbf{x}}$. Then, if $\mathbf{v}_n$ is stationary with the covariance matrix $R_n = R$, the likelihood function $\Psi_n(n)$ can be written as follows:

$$\begin{aligned}\Psi_n(\mathbf{x}) &= \exp(-\tfrac{1}{2}(\mathbf{y}_{n\tau}^T R^{-1}\mathbf{y}_{n\tau}))\exp(-\tfrac{1}{2}(\mathbf{h}^T(\mathbf{x})R^{-1}\mathbf{h}(\mathbf{x})) + (\mathbf{y}_{n\tau}^T R^{-1}\mathbf{h}(\mathbf{x})))\\ &= A_n\ \exp\left(-\sum_{i=1}^{d}\lambda_i^0 c_i(\mathbf{x}) + \sum_{k=1}^{p}(\sum_{i=1}^{p}\lambda_i^k z_{n\tau}^k)c_i(\mathbf{x})\right),\end{aligned} \tag{3.13}$$

where $\mathbf{z}_{n\tau} = \mathbf{y}_{n\tau}^T R^{-1}$, and A_n is a constant depending on $\mathbf{y}_{n\tau}$. Therefore, the coefficient $\Psi_n(\mathbf{x})$ is a member of exponential family of densities. This family is closed under multiplication. Using all of these facts, we can present the following theorem [11]:

Theorem 3.2.2 [Brigo 1996] *For system (3.3), where $\mathbf{w}_t$ is a Brownian motion process with covariance Q_t and $\mathbf{v}_i$ is a white Gaussian noise with covariance R, we assume (A3.1.1) and (A3.1.2) to be true. We also assume that $\frac{1}{2}\|\mathbf{h}(\mathbf{x})\|^2_{R^{-1}} = \sum_{i=1}^{p} \lambda_i^0 c_i(\mathbf{x})$, $h^k(\mathbf{x}) = \sum_{i=1}^{p} \lambda_i^k c_i(\mathbf{x})$, for $k = 1, \cdots, d$, and $E_{p(\cdot,\theta)}\|\frac{\mathcal{L}_t^* p(\cdot,\theta)}{p(\cdot,\theta)}\|^2 < \infty$, $\forall \theta \in \Theta$, $\forall t \geq 0$. Then for all $\theta \in \Theta$, and all $t \geq 0$, $\Pi_\theta \frac{\mathcal{L}_t^* p(\cdot,\theta)}{\sqrt{p(\cdot,\theta)}}$ is a vector on the exponential manifold $\mathcal{S}^{\frac{1}{2}}$. The projection filter density, $p_t^\Pi = p_t(\cdot, \theta_t)$ is described by*

$$\begin{aligned} \frac{\partial \sqrt{p_t(\cdot,\theta_t)}}{\partial t} &= \Pi_{\theta_t} \frac{\mathcal{L}_t^* p_t(\cdot,\theta t)}{2\sqrt{p(\cdot,\theta_t)}} , \qquad n\tau \leq t < (n+1)\tau \\ p_{n\tau}(\cdot,\theta_{n\tau}) &= c_n \Psi_n(\mathbf{y}_{n\tau}) p_{n\tau^-}(\cdot,\theta_{n\tau^-}) , \end{aligned}$$

and the projection filter parameter satisfies the following combined differential and stochastic difference equations:

$$\begin{aligned} g(\theta_t)d\theta_t &= E_{\theta_t}\{\mathcal{L}_t \mathbf{c}\}dt, \qquad n\tau \leq t < (n+1)\tau, \\ \\ \theta_{n\tau} &= \theta_{n\tau^-} - \lambda_0^0 + \textstyle\sum_{k=1}^d \lambda_0^k z_n^k, \end{aligned}$$

where

$$\mathcal{L}_t = \sum_{i=1}^{n} f_t^i \frac{\partial}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^{n} a_t^{ij} \frac{\partial^2}{\partial x_i \partial x_j},$$

and $\lambda_0^i = [\lambda_1^i, \cdots, \lambda_p^i]^T$, $i = 0, \cdots, d$, and z_n^k is the kth component of $\mathbf{z}_{n\tau}^T = R^{-1}\mathbf{y}_{n\tau}$.

Henceforth, we shall use E_θ and $E_{p(\cdot,\theta)}$, $\theta_{n\tau}$ and θ_n, and $p_{n\tau}$ and p_n, interchangeably, respectively.

Remark: The differential equation for θ_t is an ordinary differential equation with the vector field $g(\theta_t)^{-1}E_{\theta_t}\{\mathcal{L}_t\mathbf{c}\}$. This vector field should be computed analytically. If the analytical computation of this vector field is not possible an off-line numerical computation should be carried.

As can be seen from the statement of the theorem, the calculation of the conditional probability density is reduced to the calculation of the parameter of an exponential family. But, solving the differential equation in the theorem is not an easy task. At each moment $g(\theta_t)$ and $E_{\theta_t}\{\mathcal{L}_t\mathbf{c}\}$ need to be calculated. This imposes a heavy computational load. In this dissertation, we introduce a Monte Carlo method to calculate the parameter of the exponential family with a more affordable computational load.

Although projection filtering gives a better solution than EKF, there is no known error bound with which we can compare the distance between the real density and the density given by the projection filter. In the next section we review particle filtering as an alternative to optimal nonlinear filtering.

Remark : The assumption on $\mathbf{h}_n(\cdot)$ and R_n in this are made only to ensure that $\Psi_n(\cdot)$ is in the family of exponential densities. These assumptions can be relaxed if $\Psi_n(\cdot)$ is guaranteed to stay in the family.

3.3 Particle Filtering

Consider either the continuous dynamics and discrete observation in (3.3) or the discrete case,

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{f}_n(\mathbf{x}_n) + G_n(\mathbf{x}_n)\mathbf{w}_n, \quad \text{given the distribution of} \quad \mathbf{x}_0 \\ \mathbf{y}_n &= \mathbf{h}_n(\mathbf{x}_n) + \mathbf{v}_n. \end{aligned} \tag{3.14}$$

We assume that in both cases, the initial distribution for $\mathbf{x}_0$ is given. The

propagation of the conditional density, at least conceptually, can be calculated as follows [46]:

- Step 1 . Initialization:

$$p_0(\mathbf{x}_0|\mathbf{y}_0) = p(\mathbf{x}_0).$$

- Step 2 . Diffusion:

$$p_{(n+1)^-}(\mathbf{x}_{n+1}|\mathcal{Y}_n) = \int p(\mathbf{x}_{n+1}|\mathbf{x}_n) p_n(\mathbf{x}_n|\mathcal{Y}_n) d\mathbf{x}_n,$$

  where $\mathcal{Y}_n = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$.

- Step 3 . Bayes' rule update:

$$p_{(n+1)}(\mathbf{x}_{n+1}|\mathcal{Y}_{n+1}) = \frac{p(\mathbf{y}_{n+1}|\mathbf{x}_{n+1}) p_{(n+1)^-}(\mathbf{x}_{n+1}|\mathcal{Y}_n)}{\int p(\mathbf{y}_{n+1}|\mathbf{x}_{n+1}) p_{(n+1)^-}(\mathbf{x}_{n+1}|\mathcal{Y}_n) d\mathbf{x}_{n+1}} \quad ,$$

- Step 4 . $n \leftarrow n+1$; go to Step (2).

The conditional density given by the above steps is exact, but in general it can be viewed as an infinite dimensional filter, thus, not implementable. Particle filtering, in brief, is an approximation method that mimics the above calculations with a finite number of operations using the Monte Carlo method. The procedure for particle filtering is as follows [24, 40]:

Algorithm 3.3.1 *Particle Filtering*

- *Step 1 . Initialization*

  - ◇ *Sample* $\mathbf{x}_0^1, \cdots, \mathbf{x}_0^N$, *N i.i.d. random vectors with the initial distribution* $P_0(\mathbf{x})$.

- *Step 2 . Diffusion*

◇ *Find* $\widehat{\mathbf{x}}_{n+1}^1, \cdots, \widehat{\mathbf{x}}_{n+1}^N$ *from the given* $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$, *using the dynamic rules:*

$$d\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_t)dt + G_t(\mathbf{x}_t)d\mathbf{w}_t, \quad n\tau \le t < (n+1)\tau$$

or

$$\mathbf{x}_{n+1} = \mathbf{f}_n(\mathbf{x}_n) + G_n(\mathbf{x}_n)\mathbf{v}_n.$$

- *Step 3 . Find the empirical distribution*

$$P^N_{(n+1)^-}(\mathbf{x}) = \frac{1}{N}\sum_{j=1}^{N} \delta_{\widehat{\mathbf{x}}^j_{n+1}}(\mathbf{x})$$

- *Step 4 . Use Bayes' Rule*

$$P^N_{(n+1)}(\mathbf{x}) = \frac{\frac{1}{N}\sum\limits_{j=1}^{N} \delta_{\widehat{\mathbf{x}}^j_{n+1}}(\mathbf{x}) \cdot \Psi_{n+1}(\mathbf{x})}{\frac{1}{N}\sum\limits_{j=1}^{N} \delta_{\widehat{\mathbf{x}}^j_{n+1}}(\widehat{\mathbf{x}}^j_{n+1}) \cdot \Psi_{n+1}(\widehat{\mathbf{x}}^j_{n+1})}$$

- *Step 5 . Resample*

  ◇ *Sample* $\mathbf{x}_{n+1}^1, \cdots, \mathbf{x}_{n+1}^N$ *according to* $P^N_{n+1|n+1}(\mathbf{x})$

- *Step 6 .* $n \leftarrow n+1$*; go to Step (2).*

where $\delta_{\mathbf{v}}(\mathbf{w}) = 1$ if $\mathbf{w} = \mathbf{v}$ and 0 otherwise, and $\Psi_n(\mathbf{x})$ is the conditional density of the observation $\mathbf{y}_n$ given the state $\mathbf{x}$.

It is customary to call $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$ particles. In the next few lines, we try to explain in words the evolution of these particles using the above algorithm.

Let $\widehat{\mathbf{x}}_n^1, \cdots, \widehat{\mathbf{x}}_n^N$ be the distinct particles at time n before incorporating the observation at time n. The probability of each particle is $\frac{1}{N}$, that is, is uniformly distributed. After using the observations, the conditional probability of each particle changes. Some will have small, and some large probabilities. Therefore, in

the process of resampling, it is very likely that some particles will never be used and instead some other particles (with high probabilities) will be sampled more than once. Therefore, after resampling, some particles have repeated versions, but in the diffusion phase they go through different paths and at the end of the diffusion phase, it is very likely, we would have N distinct particles. This automatically makes the approximation one of better resolution in the areas where the probability is higher.

In [40] it is proved under some conditions that

$$\lim_{N\to\infty} E\left(\left| \frac{1}{N}\sum_{i=1}^{N} f(\widehat{\mathbf{x}}_n^i) - E_{P_n}(f(\mathbf{x})) \right| \right) = 0 \tag{3.15}$$

for every bounded Borel test function, $f(\cdot)$.

One problem in using the particle filtering method is the computational cost. In particular, for a high dimensional system, getting reasonable accuracy means using a large N, which results in a heavy computational cost. In the next chapter, we propose a method that can reduce the number of particles for a certain class of problems.

Chapter 4

Projection Particle Filtering

In the previous chapter, we saw two approximation methods for nonlinear filtering. In the particle filtering method, we saw that the conditional distribution is approximated by the empirical distribution. Unlike the empirical distribution, in most cases, the actual conditional distribution is smooth. Intuition suggests that if we have prior knowledge of some properties of the distribution, we can improve on the quality of the estimates over just using the empirical distribution. In this chapter first, we assume that the conditional density lies in an exponential family of densities. We will see that with this assumption, we can show the convergence of the approximated density to the actual one. Later, we relax this assumption and we only require that the conditional density stay close to the exponential family of densities. We prove that the error of the estimate for the latter case is bounded.

4.1 Particle Filtering for Exponential Families of Densities

For System (3.3), we assume that the probability density of $\mathbf{x}_t$, given the observation, is in a family of exponential densities $\mathcal{S}$ [1].

With this assumption, the proposed algorithm is as follows:

Algorithm 4.1.1 *Particle Filtering for an Exponential Family of Densities.*

- *Step 1 . Initialization*
  - ◇ *Sample* $\mathbf{x}_0^1, \cdots, \mathbf{x}_0^N$, *N i.i.d. random vectors with the density,* $p_0(\mathbf{x})$.
- *Step 2 . Diffusion*
  - ◇ *Find* $\widehat{\mathbf{x}}_{n+1}^1, \cdots, \widehat{\mathbf{x}}_{n+1}^N$ *from the given* $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$, *using the dynamic rule:*

$$d\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_t)dt + G_t(\mathbf{x}_t)d\mathbf{w}_t, \quad i\tau \leq t < (i+1)\tau$$

- *Step 3 . Find the MLE of* $\widehat{\theta}_{(n+1)^-}$ *given* $\widehat{\mathbf{x}}_{n+1}^1, \cdots, \widehat{\mathbf{x}}_{n+1}^N$ *[36]*

$$\widehat{\theta}_{(n+1)^-} = \arg\max_{\theta} \prod_{i=1}^{N} \exp(\theta^T \mathbf{c}(\widehat{\mathbf{x}}_{n+1}^i) - \Upsilon(\theta))$$

- *Step 4 . Use Bayes' Rule*

$$p(\mathbf{x}, \widehat{\theta}_{(n+1)}) = \frac{\exp\left(\widehat{\theta}_{(n+1)^-}^T \mathbf{c}(\mathbf{x}) - \Upsilon(\widehat{\theta}_{(n+1)^-})\right) \Psi_{n+1}(\mathbf{x})}{\int \exp\left(\widehat{\theta}_{(n+1)^-}^T \mathbf{c}(\mathbf{x}) - \Upsilon(\widehat{\theta}_{(n+1)^-})\right) \Psi_{n+1}(\mathbf{x}) d\mathbf{x}}$$

[1] This assumption is rather strong. We will drop this assumption later, and we will only assume that there exists a known family of densities that approximates the real density well, i.e., with acceptable accuracy.

- *Step 5 . Resample*

  - ◇ *Sample* $\mathbf{x}_{n+1}^1, \cdots, \mathbf{x}_{n+1}^N$ *according to* $p(\mathbf{x}, \widehat{\theta}_{n+1})$.

- *Step 6 .* $n \leftarrow n+1$*; go to Step (2).*

To generate $\mathbf{x}_{n+1}^1, \cdots, \mathbf{x}_{n+1}^N$, a Gibbs sampler can be used [23]. This brings an extra computational cost, which should be taken into account when choosing Algorithm 4.1.1 over Algorithm 3.3.1.

It is instructive to discuss the structure of the ML estimator. We are going to use this structure for the proof of convergence.

Let $\widehat{\mathbf{x}}_n^1, \cdots, \widehat{\mathbf{x}}_n^N$ be the value of the particles right before the measurement at time n. The MLE of θ_n, $\widehat{\theta}_n$, satisfies the first order necessary condition

$$\sum_{i=1}^{N} c_j(\widehat{\mathbf{x}}_n^i) - N \frac{\int_{\mathbf{x}} c_j(\mathbf{x}) \exp(\widehat{\theta}_n^T \mathbf{c}(\mathbf{x})) d\mathbf{x}}{\int_{\mathbf{x}} \exp(\widehat{\theta}_n^T \mathbf{c}(\mathbf{x})) d\mathbf{x}} = 0.$$

Therefore, we get

$$\frac{1}{N} \sum_{i=1}^{N} c_j(\widehat{\mathbf{x}}_n^i) = E_{\widehat{\theta}_n}(c_j(\mathbf{x})), \quad \text{for} \quad j = 1, \cdots, p \ . \tag{4.1}$$

Equation (4.1) says that the sample average of $c_j(\mathbf{x})$ and its probabilistic average, evaluated at $\widehat{\theta}_n$, should be equal. The MLE of θ is the solution to the system of equations in (4.1). Let $F_j(\theta)$ be as follows:

$$F_j(\theta) = \frac{1}{N} \sum_{i=1}^{N} c_j(\widehat{\mathbf{x}}_n^i) - \frac{\int c_j(\mathbf{x}) \exp(\theta^T \mathbf{c}(\mathbf{x})) d\mathbf{x}}{\int \exp(\theta^T \mathbf{c}(\mathbf{x})) d\mathbf{x}} \quad , \quad j = 1, \cdots, p.$$

For simplicity we drop the index n from θ_n. It is easy to see that

$$-\frac{\partial F_i}{\partial \theta_j} = E_\theta(c_i(\mathbf{x}) c_j(\mathbf{x})) - E_\theta(c_i(\mathbf{x})) E_\theta(c_j(\mathbf{x})).$$

This shows that $(-\frac{\partial F_i}{\partial \theta_j})_{i,j} = g(\theta)$, where $g(\theta)$ is the Fisher information matrix of the exponential density at θ. Since $c_i(\mathbf{x}), i = 1, \cdots, p$ are affinely independent

$g(\theta) > 0, \forall \theta \in \Theta$. Therefore, (4.1) is the necessary and sufficient condition for optimality.

In the next few pages, we prove the convergence of the MLE of θ_n, $\widehat{\theta}_n$, to θ_n in the mean square sense.

In each iteration the proposed algorithm starts from the density $p_{\widehat{\theta}_t}(\mathbf{x}_t|\mathbf{y}^t)$, $t = \tau n$, where $\widehat{\theta}_t$ is the best estimate θ_t according to the algorithm. After a full iteration, the algorithm yields $\widehat{\theta}_{t+1}$ which is the best estimate of θ_{t+1}. The error in $\widehat{\theta}_{t+1}$ is a combination of the series of possible errors for which we want to find an upper bound. The first source of error is the error in $\widehat{\theta}_t$, which will propagate even if no other error is considered. The other source comes from the fact that in each iteration new particles are resampled based on the estimated density which is different from the actual density. Finally, the last source of error comes from the discretization of the stochastic dynamics of the system. We want to emphasize that here we assume $\Psi_n(\mathbf{x}) = \exp(-\frac{1}{2}(\mathbf{y}_{n\tau} - \mathbf{h}_n(\mathbf{x}_{n\tau}))^T R_n^{-1}(\mathbf{y}_{n\tau} - \mathbf{h}_n(\mathbf{x}_{n\tau})))$ lies in the chosen family of densities. Therefore, no other error is added to the estimate because of the Bayes' correction.

We recall the following fact [36]:

Fact 4.1.2 *For the family of densities $\mathcal{S}$ with probability density*

$$p(\mathbf{x}, \theta) = \exp(\theta^T \mathbf{c}(\mathbf{x}) - \Upsilon(\theta)),$$

the Fisher information matrix $g(\theta) = (E(c_i(\mathbf{x})c_j(\mathbf{x})) - E(c_i(\mathbf{x}))E(c_j(\mathbf{x})))_{i,j}$ is positive definite. Also the log likelihood function

$$l(\theta) = \theta^T \mathbf{C}(\mathbf{x}) - \Upsilon(\theta),$$

is strictly concave. Therefore, for

$$c_j(\mathbf{x}) = E_\theta[c_j(\mathbf{x})], \qquad j = 1, \cdots, p,$$

if a solution exists[2]*, it is unique. In addition if* $\mathbf{x}_1, \cdots, \mathbf{x}_N$ *are* N *i.i.d. random variables distributed according to* $p(\mathbf{x}, \theta)$*, then the MLE of* θ*,* $\widehat{\theta}_N$*, is asymptotically normal, i.e.*

$$\begin{aligned} \widehat{\theta}_N &= \arg\max_{\theta} \prod_{i=1}^{N} p(\mathbf{x}_i, \theta) \ , \\ \sqrt{N}(\widehat{\theta}_N - \theta) &\sim \mathcal{N}(0, g^{-1}(\theta)). \end{aligned}$$

Using this fact, it is easy to see that

$$E\left(\left\|\widehat{\theta}_N - \theta\right\|^2\right) = \frac{1}{N}\ trace(g^{-1}(\theta)),$$

therefore, when $N \longrightarrow \infty$, $\widehat{\theta}_N \longrightarrow \theta$ in the m.s. sense. On the other hand, $\widehat{\theta}_N$ is the solution to (4.1). Using the strong law of large numbers [10], when $N \to \infty$ the LHS in (4.1) goes to $E_\theta(c_j(\mathbf{x}))$, $j = 1, \cdots, p$, with probability one. In other words, the solution to (4.1) when the LHS is the exact $E_\theta(c_j(\mathbf{x}))$, $j = 1, \cdots, p$, gives the exact solution for θ. Using this argument, one can expect that by finding a good estimate of the left hand side of (4.1), a good estimate of θ is accessible. In each iteration of the algorithm presented in this section the estimate of the LHS of (4.1) is found by using the Monte Carlo method and the approximate solution for the stochastic differential equation (3.3).

To approximate the solution to the stochastic differential equation (3.3), we employ the method used in [39]. In the following, we review this method briefly. The stochastic differential equation in (3.3) can be rewritten as follows:

$$d\mathbf{x}_t = \mathbf{f}_t\left(\mathbf{x}_t\right) dt + \sum_{r=1}^{q} \mathbf{g}_t^r\left(\mathbf{x}_t\right) dw_t^r, \tag{4.2}$$

where $\mathbf{g}_t^r(\cdot)$ is the r^{th} column of the matrix $G_t(\cdot)$, and w_t^r is the r^{th} component of $\mathbf{w}_t$. We introduce the operators

[2] In [17] it is shown that if $N > p$, the solution exists almost surely.

$$\Lambda_r u = \left(\mathbf{g}_r \ , \ \frac{\partial}{\partial \mathbf{x}} \right) u,$$

$$Lu = \left(\frac{\partial}{\partial t} + \left(\mathbf{f} \ , \ \frac{\partial}{\partial \mathbf{x}} \right) + \frac{1}{2} \sum_{r=1}^{q} \sum_{i=1}^{n} \sum_{j=1}^{n} g_i^r g_j^r \frac{\partial^2}{\partial \mathbf{x}_i \partial \mathbf{x}_j} \right) u,$$

where $\left(\mathbf{a} \ , \ \frac{\partial}{\partial \mathbf{x}} \right) = \sum_{i=1}^{n} a_i \frac{\partial}{\partial \mathbf{x}_i}$. Then, the approximate solution for the stochastic differential equation can be written as follows:

$$\begin{aligned} \mathbf{x}_{k+1} \ &= \ \mathbf{x}_k + \sum_{r=1}^{q} g_{t_k}^r \xi_k^r h^{\frac{1}{2}} + \mathbf{f}_{t_k} h + \sum_{r=1}^{q} \sum_{i=1}^{q} \left(\Lambda_r \mathbf{g}^r \right)_{t_k} \xi_k^{ir} h + \\ & \quad \frac{1}{2} \sum_{r=1}^{q} \left(L\mathbf{g}^r + \Lambda_r \mathbf{f} \right)_{t_k} \xi_k^r h^{\frac{3}{2}} + \left(L\mathbf{f} \right)_{t_k} \frac{h^2}{2}, \end{aligned} \tag{4.3}$$

where h is the step size and the coefficients $\mathbf{g}_{t_k}^r$, $\mathbf{f}_{t_k}$, $\left(\Lambda_i \mathbf{g}^r \right)_{t_k}$, etc., are computed at the point $(t_k, \mathbf{x}_k)$. Also, the sets of random variables ξ_k^r, ξ_k^{ir} are independent for distinct k and can, for each k, be modeled as follows:

$$\xi^{ij} = \frac{1}{2} \xi^i \xi^j - \frac{1}{2} \gamma_{ij} \zeta^i \zeta^j, \qquad \gamma_{ij} = \begin{cases} -1 & , i < j \\ 1 & , i \geq j \end{cases} .$$

and ξ^i and ζ^j are independent random variables satisfying

$$\begin{aligned} & E\xi_i = E\xi_i^3 = E\xi_i^5 = 0, \quad E\xi_i^2 = 1, \qquad E\xi_i^4 = 3, \\ & E\zeta_j = E\zeta_j^3 = 0, \qquad E\zeta_j^2 = \zeta_j^4 = 1. \end{aligned}$$

In particular, ξ_i can be modeled by the law $P\left(\xi = 0\right) = \frac{2}{3}$, $P\left(\xi = \sqrt{3}\right) = \frac{1}{6}$, and $P\left(\xi = -\sqrt{3}\right) = \frac{1}{6}$, and ζ_j can be modeled by $P\left(\zeta = -1\right) = P\left(\zeta = 1\right) = \frac{1}{2}$.

Definition 4.1.3 *We say that a function $u(\cdot)$ belongs to the class $\mathcal{F}$, written as $u \in \mathcal{F}$, if we can find constants, $k > 0$, and $\kappa > 0$, such that for all $\mathbf{x} \in \mathcal{R}^n$, the following inequality holds:*

$$\| u(\mathbf{x}) \| \leq k \left(1 + \| \mathbf{x} \|^{\kappa} \right).$$

Before we present our results we need to specify the probability space in which the random variables are defined. As we mentioned before, the stochastic process associated to the dynamics and the observation equation are defined on a fixed probability space (Ω, F, P), the expectation associated to this probability space is denoted by E. In Algorithm 4.1.1 the generated particles form a Markov process. Similar to section 2.2 of [40] we denote the probability space associated to this process by $(\Omega', F', P'_{[y]})$. The subindex y is used to emphasize that the probability measure is conditioned on the observation y. Another set of random variables, ξ^i, ζ^i, are defined for the approximation of the stochastic differential equation (4.2). We denote the probability space associated to these random variables by (Ω'', F'', P''). The expectation associated to this process is denoted by E''. Finally we define $(\widetilde{\Omega}, \widetilde{F}, \widetilde{P})$, where $\widetilde{\Omega} = \Omega \times \Omega' \times \Omega''$ and $\widetilde{F} = F \times F' \times F''$. For every $\widetilde{\omega} \in \widetilde{\Omega}$ we define $\widetilde{\omega} = (\omega, \omega', \omega'')$, then for every $A \in F$, $B \in F'$, and $C \in F''$ we define the probability measure $\widetilde{P}(A \times B \times C) = \int_A \left(\int_C P'_{[Y]}(B) dP''(\omega'') \right) dP(\omega)$. The expectation with respect the probability measure $\widetilde{P}$ is denoted by $\widetilde{E}$.

The following theorem summarizes the weak approximation results for (4.3).

Theorem 4.1.4 [**Milstein [39]**] *Suppose (A3.1.1) from Section (3.1), and suppose that the functions* $\mathbf{f}(\cdot)$, $\mathbf{g}^r(\cdot)$, $r = 1, \cdots, q$ *together with the partial derivatives of sufficiently high order, belong to class* $\mathcal{F}$. *Also, suppose that the functions* $\Lambda_i \mathbf{g}^r$, $L\mathbf{g}^r$, $\Lambda_r \mathbf{f}$, *and* $L\mathbf{f}$ *grow at most as a linear function in* $\|\mathbf{x}\|$. *Then, if the function* $u(\cdot)$ *and all its derivatives up to order* 6 *belong to class* $\mathcal{F}$, *the approximation (4.3) has the order of accuracy* 2, *in the sense of weak approximation, i.e.,*

$$\| \widetilde{E} u \left(\mathbf{x}_{0,\mathbf{x}_0}(t_k) \right) - \widetilde{E} u \left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0}(t_k) \right) \| \leq K h^2, \quad t_k \in [0, T],$$

where K *is a constant and* $\mathbf{x}_{0,\mathbf{x}_0}(\cdot)$ *and* $\widehat{\mathbf{x}}_{0,\mathbf{x}_0}(\cdot)$ *are the exact and approximate*

solutions for the stochastic differential equation, respectively.

The Monte Carlo approximation of $\widetilde{E}u\left(\mathbf{x}_{0,\mathbf{x}_0}\left(t_k\right)\right)$ brings another error term. The combination of these errors can be expressed as follows:

$$\left\|\widetilde{E}u\left(\mathbf{x}_{0,\mathbf{x}_0}\left(t_k\right)\right)-\tfrac{1}{N}\sum_{i=1}^{N}u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0^i}\left(t_k\right)\right)\right\|\leq$$

$$\left\|\widetilde{E}u\left(\mathbf{x}_{0,\mathbf{x}_0}\left(t_k\right)\right)-\widetilde{E}u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0}\left(t_k\right)\right)\right\|+\left\|\widetilde{E}u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0}\left(t_k\right)\right)-\tfrac{1}{N}\sum_{i=1}^{N}u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0^i}\left(t_k\right)\right)\right\|.$$

If the variance of $u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0}\left(t_k\right)\right)$ is bounded, we have

$$\widetilde{E}\left\|\widetilde{E}u\left(\mathbf{x}_{0,\mathbf{x}_0}\left(t_k\right)\right)-\frac{1}{N}\sum_{i=1}^{N}u\left(\widehat{\mathbf{x}}_{0,\mathbf{x}_0^i}\left(t_k\right)\right)\right\|\leq Kh^2+\frac{k^{'}}{N^{1/2}},\tag{4.4}$$

where K and $k^{'}$ are constants, and h is the step size for the approximation of the solution of the stochastic differential equation.

The next lemma relates the approximate solution to the stochastic differential equation and the estimate of the parameter θ. This lemma is the main building block for our result in this section.

Lemma 4.1.5 *For the stochastic differential equation*

$$d\mathbf{x}_t=\mathbf{f}_t\left(\mathbf{x}_t\right)dt+G_t\left(\mathbf{x}_t\right)d\mathbf{w}_t,\quad \mathbf{x}_0,\quad t\in[0,t_f],$$

assume that $\mathbf{f}_t(\cdot)$, $G_t(\cdot)$ *are such that for the Brownian motion,* $\mathbf{w}_t$*, the probability density of the state* $\mathbf{x}_t$ *lies in the family* $\mathcal{S}$ *for* Θ *bounded, with* $g(\theta)\geq\vartheta I$ *for some* $\vartheta>0$*. We also assume the conditions in* `Fact 4.1.2` *and in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$*. Then, there exist* k_1 *and* k_2 *such that*

$$\widetilde{E}[\|\theta_t-\widehat{\theta}_t\|]\leq k_1h^2+\frac{k_2}{N^{1/2}},\quad t\in[0,t_f]\tag{4.5}$$

where $\widehat{\theta}_t$ *is the estimate of* θ_t*, and* N *and* h *are the number of particles and the time step, respectively.*

Proof: Let θ_0 be the initial condition for θ. At $t = 0$, N independent initial conditions are generated based on the density $p(\mathbf{x}, \theta_0)$, and the approximation method (4.3) is applied. From (4.4) we know that:

$$\widetilde{E}\|E_{\theta_t}\mathbf{c}(\mathbf{x}_t) - \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}\left(\widehat{\mathbf{x}}_t^i\right)\| \leq Kh^2 + \frac{k'}{N^{1/2}}.$$

On the other hand, from (4.1), we know that $\widehat{\theta}$ is a solution to the system of equations

$$\frac{1}{N}\sum_{i=1}^{N} c_j(\widehat{\mathbf{x}}_t^i) = E_{\widehat{\theta}_t}(c_j(\mathbf{x}_t)), \quad \text{for} \quad j = 1, \cdots, p.$$

From `Fact 4.1.2`, the solution is exact if we replace $\frac{1}{N}\sum_{i=1}^{N} c_j(\widehat{\mathbf{x}}_t^i)$ by $E_{\theta_t}(c_j(\mathbf{x}_t))$. Subtracting the term $E_{\theta_t}(c_j(\mathbf{x}))$ from both sides of the above equations and using the vector form for it, we get

$$\frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_t^i) - E_{\theta_t}(\mathbf{c}(\mathbf{x}_t)) = E_{\widehat{\theta}_t}(\mathbf{c}(\mathbf{x}_t)) - E_{\theta_t}(\mathbf{c}(\mathbf{x}_t)).$$

On the other hand, we know that $E_\theta(\mathbf{c}(\mathbf{x}))$ is a differentiable and one to one function of θ (see `Fact 4.1.2`). The derivative of this function, $g(\theta)$, is positive definite and by assumption $g(\theta) \geq \vartheta I$. Therefore, $\exists \alpha > 0$ such that

$$\begin{aligned} \|\theta_t - \widehat{\theta}_t\| &\leq \alpha\|E_{\theta_t}(\mathbf{c}(\mathbf{x}_t)) - E_{\widehat{\theta}_t}(\mathbf{c}(\mathbf{x}_t))\| \\ &= \alpha\|E_{\theta_t}(\mathbf{c}(\mathbf{x}_t)) - \tfrac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_t^i)\|. \end{aligned}$$

Taking the expectation on both sides of the inequality we have

$$\begin{aligned} \widetilde{E}\|\theta_t - \widehat{\theta}_t\| &\leq \alpha\ \widetilde{E}\|\tfrac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_t^i) - E_{\theta_t}(\mathbf{c}(\mathbf{x}_t))\| \\ &\leq \alpha\left(Kh^2 + \tfrac{k'}{N^{1/2}}\right) \\ &= k_1h^2 + \tfrac{k_2}{N^{1/2}} \end{aligned}$$

◇

Now we are ready to present the main result of this section.

Theorem 4.1.6 *For System (3.3) assume that* $\mathbf{f}_t(\cdot)$, $G_t(\cdot)$, *and* $\mathbf{h}(\cdot)$ *are such that for the Brownian motion* $\mathbf{w}_t$, *and the Gaussian noise* $\mathbf{v}_n$, *the conditional probability density of the state* $\mathbf{x}_t$, *conditioned on the observations, lies in the family* $\mathcal{S}$ *for* Θ *bounded and for* $t \in [0, T]$. *Also assume the conditions in* `Fact 4.1.2` *and in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$. *Then, if* $g^{-1}(\theta_t) E_{\theta_t}(\mathcal{L}_t \mathbf{c}(\mathbf{x}))$ *is Lipschitz with Lipschitz constant* L *and* $g(\theta) \geq \vartheta I$, *there exist* l_1 *and* l_2 *such that*

$$\widetilde{E}\|\theta_n - \widehat{\theta}_n\| \leq \sum_{i=0}^{n-1} \exp(Li\tau) \left(l_1 h^2 + \frac{l_2}{N^{1/2}} \right), \qquad n\tau \in [0, T],$$

where $\widehat{\theta}_n$ *is the estimate of* θ_n, *and* N *and* h *are the number of particles and the time step, respectively. This inequality implies convergence of the estimated parameter,* $\widehat{\theta_n}$, *to the true parameter,* θ_n, *as* $h \longrightarrow 0$ *and* $N \longrightarrow \infty$.

Proof: Let θ_t and $\widehat{\theta}_t$ be the actual and the estimated values of the parameter of the density at time $t = n\tau$, respectively. At time $t' = (n+1)\tau$ the error in the estimate of $\theta_{t'}$ is a combination of the error in the estimate in $\widehat{\theta}_t$ and the error added in the time interval $[t, t']$.

If the conditional density stays in the exponential family of densities, θ_t has to satisfy the following differential equation:

$$\dot{\theta} = g^{-1}(\theta) E_{\theta_t}(\mathcal{L}_t \mathbf{c}(\mathbf{x}))\, dt, \qquad n\tau \leq t < (n+1)\tau.$$

Let $\widetilde{\theta}_{t'}$ be the estimate of $\theta_{t'}$, if the error due to resampling and the approximation of the stochastic differential equation solution is not taken into account in the interval $[t, t']$ (i.e. $\widetilde{\theta}_{t'}$ is computed from the above ordinary differential equation starting at $\widehat{\theta}_t$), then

$$\|\theta_{t'} - \widehat{\theta}_{t'}\| \leq \|\theta_{t'} - \widetilde{\theta}_{t'}\| + \|\widetilde{\theta}_{t'} - \widehat{\theta}_{t'}\|.$$

By the assumption of the theorem, $g^{-1}(\theta) E_{\theta_t}(\mathcal{L}_t \mathbf{c}(\mathbf{x}))$ is Lipschitz with Lipschitz constant L, then by continuity of the solution of the differential equation with respect to the initial condition [31], we know that

$$\left\|\theta_{t'} - \widetilde{\theta}_{t'}\right\| \leq \left\|\theta_t - \widehat{\theta}_t\right\| e^{L(t'-t)},$$

therefore,

$$\widetilde{E}\left\|\theta_{t'} - \widetilde{\theta}_{t'}\right\| \leq \widetilde{E}\left\|\theta_t - \widehat{\theta}_t\right\| e^{L(t'-t)}.$$

Also from the `Lemma 4.1.5`, $\exists k_1(t^{'})$ and $k_2(t^{'})$ such that

$$\widetilde{E}[\|\widetilde{\theta}_{t'} - \widehat{\theta}_{t'}\|] \leq k_1(t^{'})h^2 + \frac{k_2(t^{'})}{N^{1/2}},$$

therefore,

$$\widetilde{E}\|\theta_{t'} - \widehat{\theta}_{t'}\| \leq \widetilde{E}\left\|\theta_t - \widehat{\theta}_t\right\| e^{L(t'-t)} + k_1(t^{'})h^2 + \frac{k_2(t^{'})}{N^{1/2}}.$$

The observation noise $\mathbf{v}_n$ and the function $\mathbf{h}(\cdot)$ are such that Bayes' Rule does not introduce any further error in the estimate of $\widehat{\theta}_{t'}$. More precisely, $\Psi_n(\mathbf{x})$ is assumed to be a member of $\mathcal{S}$. This implies that after applying Bayes' Rule to $p(\mathbf{x}, \theta_{t'})$ and $p(\mathbf{x}, \widehat{\theta}_{t'})$ parameters $\theta_{t'}$ and $\widehat{\theta}_{t'}$ are shifted with the same vector and therefore, $\|\theta_{t^{+'}} - \widehat{\theta}_{t^{+'}}\| = \|\theta_{t'} - \widehat{\theta}_{t'}\|$. Here $t^{+'}$ represents the time right after Bayes' correction. Therefore, starting from the initial condition θ_0 we get

$$\widetilde{E}\|\theta_n - \widehat{\theta}_n\| \leq \sum_{i=0}^{n-1} \exp(Li\tau)\left(l_1 h^2 + \frac{l_2}{N^{1/2}}\right), \quad n\tau \in [0, T]$$

where

$$l_i = \max_n k_i(n\tau), \quad n\tau \in [0, T], \quad i = 1, 2.$$

$\diamond$

Here, we would like to make a few remarks:

- The result of Theorem 4.1.6 can be easily extended to convergence in the mean square sense.

- The assumption that the probability density stays in the family of densities, $\mathcal{S}$, does not seem very realistic. But with our approach, we should be able to get the result in [11]. In fact, in [11] the evolution of the density is forced to stay in the family at every single moment. In our method, we only force the density to be in the family at the end of each full iteration, i.e. observation epoch. This allows the estimated density to be closer to the actual density.

- In [11] the observation equation is considered to be time invariant. Here, the time-varying nature of $\mathbf{h}_n(\mathbf{x})$ does not complicate the algorithm. It surely affects the assumption that the density stays in the family, but as we explained earlier, this assumption is not realistic to begin with, and it will be dropped.

- If $u(\cdot)$ is in $\mathcal{F}$, then

$$\lim_{\substack{N\longrightarrow\infty \\ h\longrightarrow 0}} \widetilde{E}\,\|E_\theta u(\mathbf{x}) - E_{\theta^*} u(\mathbf{x})\| = 0.$$

  This is a criterion similar to the one used in [40].

4.2 Projection Particle Filtering for Exponential Families of Densities

In this section, we drop the assumption that the conditional density of the state given the observation (3.6) lies in the exponential family of densities, $\mathcal{S}$. Also, we do not require that $\Psi_n(\mathbf{x})$ is a member of $\mathcal{S}$. Instead we make other assumptions. First we need the following definition:

Definition 4.2.1 *We say that a function $u(\cdot)$ belongs to the class $\mathcal{F}_{k\kappa}$, written as $u \in \mathcal{F}_{k\kappa}$, for fixed $k > 0$ and $\kappa > 0$, such that for all $\mathbf{x} \in \mathcal{R}^n$, the following inequality holds:*

$$\|u(\mathbf{x})\| \leq k\left(1 + \|\mathbf{x}\|^{\kappa}\right).$$

The next two assumptions are to guarantee the existence of an exponential density close to the true conditional density.

A 4.2.2 *For the density in (3.6) there exists an exponential family of densities $\mathcal{S}$ such that $\forall t \in [0,T]$, $\forall u \in \mathcal{F}_{k\kappa}$ $\exists \theta_t^* \in \Theta^*$ and $\epsilon > 0$ such that*

$$\widetilde{E}\|E_{p_t}(u(\mathbf{x})) - E_{\theta_t^*}(u(\mathbf{x}))\| \leq \epsilon\ , \tag{4.6}$$

where Θ^* is convex [3] *and compact.*

[3] It is easy to see that the assumption of convexity is very natural. Assume $\theta_1, \theta_2 \in \Theta^*$ then $\int \exp(\theta_i^T \mathbf{c}(\mathbf{x}))d\mathbf{x} \leq \infty$ for $i = 1, 2$. Therefore, using the Holder inequality we have

$$\begin{aligned}
\int \exp((\alpha\theta_1^T + (1-\alpha)\theta_2^T)\mathbf{c}(\mathbf{x}))d\mathbf{x} &= \int (\exp(\theta_1^T \mathbf{c}(\mathbf{x})))^{\alpha} (\exp(\theta_2^T \mathbf{c}(\mathbf{x})))^{(1-\alpha)} d\mathbf{x} \\
&\leq \left(\int \left(\left(\exp(\theta_1^T \mathbf{c}(\mathbf{x}))\right)^{\alpha}\right)^{1/\alpha} d\mathbf{x}\right)^{\alpha} \left(\int \left(\left(\exp(\theta_2^T \mathbf{c}(\mathbf{x}))\right)^{1-\alpha}\right)^{1/1-\alpha} d\mathbf{x}\right)^{1-\alpha} \\
&= \left(\int \exp(\theta_1^T \mathbf{c}(\mathbf{x}))d\mathbf{x}\right)^{\alpha} \left(\int \exp(\theta_2^T \mathbf{c}(\mathbf{x}))d\mathbf{x}\right)^{1-\alpha} \\
&\leq \infty
\end{aligned}$$

where $0 < \alpha < 1$.

A 4.2.3 *For $\theta^*_{n^-}$ in (A4.2.2) and $\Psi_n(\mathbf{x})$, $\exists \Psi^*_n(\mathbf{x})$ such that*

$$p(\mathbf{x},\theta) = \frac{p(\mathbf{x},\theta^*_{n^-})\Psi^*_n(\mathbf{x})}{\int p(\mathbf{x},\theta^*_{n^-})\Psi^*_n(\mathbf{x})d\mathbf{x}}$$

is in the family $\mathcal{S}$ for some $\theta \in \Theta^*$ and we have:

- *$\forall \theta \in \Theta^*$ and $\forall u(\cdot) \in \mathcal{F}_{k\kappa}$, $\exists \epsilon > 0$ such that*

$$\widetilde{E}\| \frac{E_\theta \Psi_n(\mathbf{x})u(\mathbf{x})}{E_\theta \Psi_n(\mathbf{x})} - \frac{E_\theta \Psi^*_n(\mathbf{x})u(\mathbf{x})}{E_\theta \Psi^*_n(\mathbf{x})}\| \leq \epsilon.$$

- *$\forall u(\cdot) \in \mathcal{F}_{k\kappa}$, $\exists \epsilon > 0$ such that*

$$\widetilde{E}\| \frac{E_{\theta^*_{n^-}} \Psi^*_n(\mathbf{x})u(\mathbf{x})}{E_{\theta^*_{n^-}} \Psi^*_n(\mathbf{x})} - \frac{E_{p_{n^-}} \Psi_n(\mathbf{x})u(\mathbf{x})}{E_{p_{n^-}} \Psi_n(\mathbf{x})}\| \leq \epsilon.$$

From Assumption (A4.2.3) it is clear that if $\Psi^*_n(\cdot)$ satisfies the requirements of the assumption then $c\Psi^*_n(\cdot)$ satisfies the same requirements, where c is a positive constant. Therefore, without loss of generality we assume that $\Psi^*_n(\cdot) = \exp(\overline{\alpha}^T \mathbf{c}(\cdot))$ for some $\overline{\alpha} \in \mathcal{R}^p$. Using Assumption (A4.2.2), we can state the following fact.

Fact 4.2.4 *$\forall \theta_1, \theta_2 \in \Theta^*$ and $\forall u \in \mathcal{F}_{k\kappa}$, $\exists K_1, K_2$ positive such that*

$$\|E_{\theta_1} u(\mathbf{x}) - E_{\theta_2} u(\mathbf{x})\| \quad \leq \quad K_1 \|\theta_1 - \theta_2\| \tag{4.7}$$

$$\|\theta_1 - \theta_2\| \quad \leq \quad K_2 \|E_{\theta_1} \mathbf{c}(\mathbf{x}) - E_{\theta_2} \mathbf{c}(\mathbf{x})\| \ . \tag{4.8}$$

Proof: To prove (4.7), define $f_u(\theta) = E_\theta u(\mathbf{x})$ for $u(\cdot) \in \mathcal{F}_{k\kappa}$. Then

$$\frac{d}{d\theta_i} f_u(\theta) = E_\theta c_i(\mathbf{x})u(\mathbf{x}) - E_\theta c_i(\mathbf{x}) E_\theta u(\mathbf{x}).$$

Since $\|u(\mathbf{x})\| \leq k(1 + \|\mathbf{x}\|^\kappa)$ and $\theta \in \Theta^*$, where Θ^* is compact, then there exists a constant A such that

$$\| \frac{df_u(\theta)}{d\theta} \| \leq A \qquad \forall u(\cdot) \in \mathcal{F}_{k\kappa} \quad \text{and} \quad \forall \theta \in \Theta^*.$$

Since Θ^* is convex and compact, it is clear that $\exists K_1$ independent of $u(\cdot)$ such that $f_u(\mathbf{x})$ is Lipschitz over Θ^* with the Lipschitz constant K_1 [31].

Inequality (4.8) follows from the fact that Θ^* is compact and the Fisher information matrix $g(\theta) > \vartheta I$ over Θ^*.

$\diamond$

Denote the interior of the set Θ^* by Θ^*_{int}. For Θ^*_{int} we can state the following fact.

Fact 4.2.5 *The set*

$$\mathcal{A} = \left\{ \alpha : \int \exp(\alpha^T \mathbf{c}(\mathbf{x})) \exp(\theta^T \mathbf{c}(\mathbf{x})) d\mathbf{x} < \infty, \forall \theta \in \Theta^*_{int} \;\; and \;\; \alpha \in \mathcal{R}^p \right\}$$

is closed.

Proof: Assume $\mathcal{A}$ is not closed. Therefore, there exists a converging sequence $\{\alpha_i\} \subset \mathcal{A}$ with converging point $\overline{\alpha} \notin \mathcal{A}$, then $\exists \overline{\theta} \in \Theta^*_{\text{int}}$ such that

$$\int \exp(\overline{\alpha}^T \mathbf{c}(\mathbf{x})) \exp(\overline{\theta}^T \mathbf{c}(\mathbf{x})) d\mathbf{x} > M, \qquad \forall M \in \mathcal{R}.$$

Since Θ^*_{int} is an open set, $\exists \epsilon > 0$ such that $\mathcal{N}_\epsilon(\overline{\theta}) \in \Theta^*_{\text{int}}$. Also, since $\{\alpha_i\}$ is a converging sequence, $\exists k > 0$ such that $\alpha_k \in \mathcal{N}_\epsilon(\overline{\alpha})$. This implies that $\theta_1 \in \Theta^*_{\text{int}}$ where $\theta_1 = \overline{\theta} + \overline{\alpha} - \alpha_k$. Therefore,

$$\int \exp(\alpha_k^T \mathbf{c}(\mathbf{x})) \exp(\theta_1^T \mathbf{c}(\mathbf{x})) d\mathbf{x} < \infty.$$

On the other hand, we know that

$$\exp(\alpha_k^T \mathbf{c}(\mathbf{x})) \exp(\theta_1^T \mathbf{c}(\mathbf{x})) = \exp(\overline{\alpha}^T \mathbf{c}(\mathbf{x})) \exp(\overline{\theta}^T \mathbf{c}(\mathbf{x})) \; .$$

This is a contradiction, therefore, $\mathcal{A}$ is closed.

$\diamond$

The following lemma is one of the building blocks of the results of this section.

Lemma 4.2.6 *For $\theta^*_{n^-}$ and $\Psi^*_n(\mathbf{x})$ defined in (A4.2.3), and $\forall u(\cdot) \in \mathcal{F}_{k\kappa}$, $\exists$ positive numbers k_1, k_2, k_3, k_4 independent of $\theta^*_{n^-}$ and $\Psi^*_n(\mathbf{x})$, such that $\forall \theta_1, \theta_2 \in \Theta^*$ the following are true.*

(a) $\quad k_1 \leq \|E_\theta \Psi^*_n(\mathbf{x})\| \leq k_2 \quad \forall \theta \in \Theta^*.$

(b) $\quad \|E_\theta \Psi^*_n(\mathbf{x}) u(\mathbf{x})\| \leq k_3 \quad \forall \theta \in \Theta^*.$

(c) $\quad \|E_{\theta_1} \Psi^*_n(\mathbf{x}) u(\mathbf{x}) - E_{\theta_2} \Psi^*_n(\mathbf{x}) u(\mathbf{x})\| \leq k_4 \|\theta_1 - \theta_2\|.$

Proof: Let $\mathcal{M}$ be a set defined as follows

$$\mathcal{M} = \{\mathbf{m} : \mathbf{m} = \theta_1 - \theta_2, \quad \forall \theta_1, \theta_2 \in \Theta^*\}.$$

We claim that $\mathcal{M}$ is compact. To prove this claim, assume $\{\mathbf{m}_i\}$ to be a sequence in $\mathcal{M}$, i.e $\mathbf{m}_i \in \mathcal{M}$. Also we assume that $\lim_{i \longrightarrow \infty} \mathbf{m}_i = \overline{\mathbf{m}}$. We know that there exist sequences $\{\theta_{1,i}\}$ and $\{\theta_{2,i}\}$ such that $\mathbf{m}_i = \theta_{1,i} - \theta_{2,i}$ and $\theta_{1,i}, \theta_{2,i} \in \Theta^*$. Since Θ^* is compact there exist converging subsequences $\{\overline{\theta}_{1,i}\}$ and $\{\overline{\theta}_{2,i}\}$ in Θ^*. This implies that $\overline{\mathbf{m}} = \overline{\theta}_1 - \overline{\theta}_2$, where $\overline{\theta}_1$ and $\overline{\theta}_2$ are the limits of the subsequences $\{\overline{\theta}_{1,i}\}$ and $\{\overline{\theta}_{2,i}\}$. Since $\overline{\theta}_1$ and $\overline{\theta}_2 \in \Theta^*$, then $\overline{\mathbf{m}} \in \mathcal{M}$, therefore $\mathcal{M}$ is closed. Since Θ^* is bounded, $\mathcal{M}$ is bounded and therefore, it is compact.

We define set $\mathcal{A}_1$ as follows:

$$\mathcal{A}_1 = \left\{ \alpha : \int \exp(\alpha^T \mathbf{c}(\mathbf{x})) \exp(\theta^T \mathbf{c}(\mathbf{x})) d\mathbf{x} < \infty, \forall \theta \in \Theta^* \text{ and } \alpha \in \mathcal{R}^p \right\}.$$

It is clear that $\mathcal{A}_1 \subset \mathcal{A}$. As we mentioned before, without loss of generality we can assume $\Psi^*_n(\mathbf{x}) = \exp(\overline{\alpha}^T \mathbf{c}(\mathbf{x}))$ and from Assumption (A4.2.3) it is clear that $\overline{\alpha} \in \mathcal{A} \bigcap \mathcal{M}$. Since $\mathcal{A} \bigcap \mathcal{M}$ and Θ^* are compact we have

$$\min_{\theta \in \Theta^*} \min_{\alpha \in \mathcal{A} \bigcap \mathcal{M}} \|E_\theta \Psi^*_n(\mathbf{x})\| \leq \|E_\theta \Psi^*_n(\mathbf{x})\| \leq \max_{\theta \in \Theta^*} \max_{\alpha \in \mathcal{A} \bigcap \mathcal{M}} \|E_\theta \Psi^*_n(\mathbf{x})\|.$$

In other words (a) is true with $k_1 = \min_{\theta\in\Theta^*} \min_{\alpha\in\mathcal{A}\bigcap\mathcal{M}} \|E_\theta \Psi_n^*(\mathbf{x})\|$ and $k_2 = \max_{\theta\in\Theta^*} \max_{\alpha\in\mathcal{A}\bigcap\mathcal{M}} \|E_\theta \Psi_n^*(\mathbf{x})\|$. Similarly, since $u(\cdot) \in \mathcal{F}_{k\kappa}$, (b) is true.

Using the above argument and the argument in Fact 4.2.4, we can show that $\|\frac{d}{d\theta} E_\theta \Psi_n^*(\mathbf{x}) u(\mathbf{x})\|$ is uniformly bounded and since Θ^* is convex and compact, then (c) is true [31].

$\diamond$

In the following we go through the proof of the theorem that we state later precisely. Assume $\widehat{\theta}_n$ is calculated according to Algorithm 4.1.1 and assume $p(\mathbf{x}, \widehat{\theta}_n)$ is such that $\forall u \in \mathcal{F}_{k\kappa}$

$$\widetilde{E}\|E_{\widehat{\theta}_n} u(\mathbf{x}) - E_{\theta_n^*} u(\mathbf{x})\| \;\leq\; \delta \;, \tag{4.9}$$

where θ_n^* (see Assumption (A4.2.2)) satisfies

$$\widetilde{E}\|E_{p_n} u(\mathbf{x}) - E_{\theta_n^*} u(\mathbf{x})\| \;\leq\; \epsilon. \tag{4.10}$$

Using the density $p(\mathbf{x}, \widehat{\theta}_n)$, new particles $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$ are generated. The approximate solution for the stochastic differential equation at time $(n+1)\tau$ maps these particles to $\widehat{\mathbf{x}}_{n+1}^1, \cdots, \widehat{\mathbf{x}}_{n+1}^N$. From these new particles $\widehat{\theta}_{n+1}$ is calculated. From (4.9) and (4.10) we have

$$\widetilde{E}\|E_{p_n} u(\mathbf{x}) - E_{\widehat{\theta}_n} u(\mathbf{x})\| \;\leq\; \delta + \epsilon. \tag{4.11}$$

We define the function $\mathbf{r}(\mathbf{x})$ as follows:

$$\mathbf{r}(\mathbf{x}) = E'' \mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)))$$

where $\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau)$ is the approximate solution of stochastic differential equation (4.2) at time $(n+1)\tau$ with the given initial condition $\mathbf{x}$ at time $n\tau$ using the

method in (4.3). Since according to our assumption $\mathbf{c} \in \mathcal{F}_{k\kappa}$, then by using lemma 9.1 in [39], we have

$$\|\mathbf{r}(\mathbf{x})\| \leq K_3(1 + \|\mathbf{x}\|^{\mu}),$$

where K_3 and μ only depend on the function $\mathbf{c}(\cdot)$ and the dimension of $\mathbf{x}$. We assume that $\mathbf{r} \in \mathcal{F}_{k\kappa}$. If the argument of $\mathbf{r}(\cdot)$ is a random variable, then using (4.11) we have

$$\widetilde{E}\|E_{p_n}\mathbf{r}(\mathbf{x}) - E_{\widehat{\theta}_n}\mathbf{r}(\mathbf{x})\| \quad \leq \quad \delta + \epsilon. \tag{4.12}$$

More explicitly,

$$\widetilde{E}\|E_{p_n}E''[\mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau))] - E_{\widehat{\theta}_n}E''[\mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau)]\| \quad \leq \quad \delta + \epsilon. \tag{4.13}$$

From Theorem 4.1.4 we have

$$\widetilde{E}\|E_{p_n}\mathbf{c}(\mathbf{x}_{n,\mathbf{x}}((n+1)\tau)) - E_{p_n}E''\mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau))\| \quad \leq \quad K_4h^2, \tag{4.14}$$

for some $K_4 > 0$.

Using the Monte Carlo method to calculate the $E_{p_n}\mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau))$ brings another error term that is due to the finite number of particles as the initial conditions for method (4.3). The expectation of this error is bounded, i.e. $\exists K_5 > 0$ s.t.

$$\widetilde{E}\|E_{\widehat{\theta}_n}E''\mathbf{c}(\widehat{\mathbf{x}}_{n,\mathbf{x}}((n+1)\tau)) - \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n,\widehat{\mathbf{x}}_i^i}((n+1)\tau))\| \quad \leq \quad \frac{K_5}{N^{\frac{1}{2}}}, \tag{4.15}$$

where $\widehat{\mathbf{x}}^i$ are distributed according to $p(\mathbf{x}, \widehat{\theta}_n)$. Combining (4.13), (4.14), and (4.15) we get

$$\begin{aligned}\widetilde{E}\|E_{p_n}\mathbf{c}(\mathbf{x}_{n,\mathbf{x}}((n+1)\tau)) - \tfrac{1}{N}\textstyle\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n,\widehat{\mathbf{x}}^i}((n+1)\tau))\| \leq \\ \delta + \epsilon + K_4h^2 + \tfrac{K_5}{N^{\frac{1}{2}}}.\end{aligned} \tag{4.16}$$

Based on (A4.2.2), we know that $\exists \theta^*_{(n+1)^-}$ such that

$$\widetilde{E}\|E_{p_{(n+1)^-}}\mathbf{c}(\mathbf{x}) - E_{\theta^*_{(n+1)^-}}\mathbf{c}(\mathbf{x})\| \leq \epsilon. \tag{4.17}$$

We know that, if $\mathbf{x}$ (initial condition at time $n\tau$) is distributed according to $p_n(\mathbf{x})$, then $E_{p_{(n+1)^-}}\mathbf{c}(\mathbf{x}) = E_{p_n}\mathbf{c}(\mathbf{x}_{n,\mathbf{x}}((n+1)\tau))$, therefore, from (4.16) and (4.17) we get

$$\widetilde{E}\|E_{\theta^*_{(n+1)^-}}\mathbf{c}(\mathbf{x}) - \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n,\widehat{\mathbf{x}}^i}((n+1)\tau))\| \leq \delta + 2\epsilon + K_4h^2 + \frac{K_5}{N^{\frac{1}{2}}}. \tag{4.18}$$

Then $\widehat{\theta}_{(n+1)^-}$ given by Algorithm 4.1.1 satisfies the following inequality

$$\widetilde{E}\|E_{\theta^*_{(n+1)^-}}\mathbf{c}(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)^-}}\mathbf{c}(\mathbf{x})\| \leq \delta + 2\epsilon + K_4h^2 + \frac{K_5}{N^{\frac{1}{2}}}. \tag{4.19}$$

From (A4.2.3) we know that $\exists \theta \in \Theta^*$ such that

$$\begin{aligned}\widetilde{E}\left\|\frac{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})} - \frac{E_{p_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{p_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})}\right\| &= \widetilde{E}\left\|E_\theta u(\mathbf{x}) - E_{p_{(n+1)}}u(\mathbf{x})\right\| \\ &\leq \epsilon.\end{aligned}$$

Since θ satisfies the inequality in (A4.2.2), we can choose $\theta^*_{(n+1)}$ to be θ, i.e. $\theta^*_{(n+1)} = \theta$.

On the other hand we have

$$\begin{aligned}\left\|E_{\theta^*_{(n+1)}}u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)}}u(\mathbf{x})\right\| &= \left\|\frac{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})} - \frac{E_{\widehat{\theta}_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})}\right\| \\ &\leq \left\|\frac{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})} - \frac{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})}\right\| + \\ &\quad \left\|\frac{E_{\theta^*_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})} - \frac{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})}\right\| + \\ &\quad \left\|\frac{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi^*_{n+1}(\mathbf{x})} - \frac{E_{\widehat{\theta}_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})}\right\|,\end{aligned}$$

therefore,

$$\left\| E_{\theta^*_{(n+1)}} u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)}} u(\mathbf{x}) \right\| \leq$$

$$\frac{\|E_{\theta^*_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) u(\mathbf{x})\|}{\|E_{\theta^*_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x})\| \|E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x})\|} \left\| E_{\theta^*_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) \right\| +$$

$$\frac{1}{\|E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x})\|} \left\| E_{\theta^*_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) u(\mathbf{x}) \right\| +$$

$$\left\| \frac{E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x}) u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}} \Psi^*_{n+1}(\mathbf{x})} - \frac{E_{\widehat{\theta}_{(n+1)^-}} \Psi_{n+1}(\mathbf{x}) u(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} \right\|.$$

Using Lemma 4.2.6 and (A4.2.3) we get

$$\widetilde{E} \| E_{\theta^*_{(n+1)}} u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)}} u(\mathbf{x}) \| \leq \frac{k_3 k_4 + k_1 k_4}{k_1^2} \widetilde{E} \| \theta^*_{(n+1)^-} - \widehat{\theta}_{(n+1)^-} \| + \epsilon.$$

Therefore, from (4.19) and Fact 4.2.4 we get

$$\widetilde{E} \| \theta^*_{(n+1)^-} - \widehat{\theta}_{(n+1)^-} \| \leq K_2 \left(\delta + 2\epsilon + K_4 h^2 + \frac{K_5}{N^{\frac{1}{2}}} \right).$$

This implies that, $\exists \iota_1, \iota_2, \iota_3, \iota_4 > 0$ such that

$$\widetilde{E} \| E_{\theta^*_{(n+1)}} u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)}} u(\mathbf{x}) \| \leq \iota_1 \delta + \iota_2 \epsilon + \iota_3 h^2 + \iota_4 N^{-\frac{1}{2}}.$$

The next theorem summarizes our result in this section.

Theorem 4.2.7 *For the system (3.3) assume (A3.1.1), (A3.1.2), (A4.2.2), and (A4.2.3). We also assume the conditions in* `Fact 4.1.2` *and in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$*, and we assume* $\mathbf{r} \in \mathcal{F}_{k\kappa}$*. Then in* `Algorithm 4.1.1` *with approximation (4.3), if* $\forall u(\cdot) \in \mathcal{F}_{k\kappa}$

$$\widetilde{E} \| E_{\widehat{\theta}_n} u(\mathbf{x}) - E_{\theta^*_n} u(\mathbf{x}) \| \quad \leq \quad \delta$$

then

$$\widetilde{E}\|E_{\theta^*_{(n+1)}}u(\mathbf{x}) - E_{\widehat{\theta}_{(n+1)}}u(\mathbf{x})\| \leq \iota_1\delta + \iota_2\epsilon + \iota_3 h^2 + \iota_4 N^{-\frac{1}{2}},$$

for some $\iota_1, \iota_2, \iota_3, \iota_4 > 0$.

In Theorem 4.2.7 only one step of Algorithm 4.1.1 is considered; it is straightforward to then use Theorem 4.2.7 repeatedly for the time interval $[0, T]$, where $T = M\tau$. In that case, $\|E_{\widehat{\theta}_0}u(\mathbf{x}) - E_{\theta^*_0}u(\mathbf{x})\| \leq \delta_0$, then $\exists \alpha_1, \alpha_2, \alpha_3$, and α_4 positive such that

$$\widetilde{E}\|E_{\theta^*_n}u(\mathbf{x}((n)\tau)) - E_{\widehat{\theta}_n}u(\mathbf{x}((n)\tau))\| \leq \alpha_1\delta_0 + \alpha_2\epsilon + \alpha_3 h^2 + \alpha_4 N^{-1/2},$$

for $0 \leq n \leq M$.

Chapter 5

Application of Projection Particle Filtering in Navigation

In this chapter we use the approximation methods for nonlinear filters introduced in the previous chapters in position estimation for systems with nonlinear dynamics and observation. We are particularly interested in the situations where methods based on linearization such as EKF fail to provide reasonable estimates.

In the first part of this chapter we address the problem of positioning in the presence of integer uncertainty. Such uncertainties arise in navigation problems where carrier phase differential GPS is part of the observations. In these cases resolving the integer ambiguity is essential for the navigation system.

In the second part of this chapter we apply projection particle filtering to an Integrated INS/GPS. We show that when the number of visible satellites is below a critical number nonlinear filtering can provide an accurate estimate of the position while EKF fails to converge.

5.1 Particle Filtering for Nonlinear Systems with Constant Integer Uncertainty

Consider the following nonlinear dynamics and observation

$$\begin{aligned} d\mathbf{x}_t &= \mathbf{f}_t(\mathbf{x}_t)dt + G_t(\mathbf{x}_t)d\mathbf{w}_t \\ \mathbf{y}_{n\tau} &= \mathbf{h}_n(\mathbf{x}(n\tau)) + J_n\mathbf{z} + \mathbf{v}_n \end{aligned}$$

where the assumptions and the dimensions for $\mathbf{x}_t$, $\mathbf{y}_{n\tau}$, $\mathbf{w}_t$, and $\mathbf{v}_n$ are the same as in the previous sections. We assume that $\mathbf{z}$ is a random integer vector, i.e. $\mathbf{z} \in \mathcal{Z}^m$ and J_n has the proper dimension. Vector $\mathbf{z}$ is assumed to be constant in time. This problem can be set up in discrete time as well. In this case,the system dynamics and the observation can be written as follows:

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{f}_n(\mathbf{x}_n) + G_n(\mathbf{x}_n)\mathbf{w}_n \\ \mathbf{y}_n &= \mathbf{h}_n(\mathbf{x}_n) + J_n\mathbf{z} + \mathbf{v}_n \end{aligned}$$

In both setups we assume that the integer uncertainty affects only some components of the observation, and other components are unaffected by $\mathbf{z}$. The affected components have associated noise components in $\mathbf{v}_n$ that have considerably lower energy. In other words, the uncertain components of $\mathbf{y}_{n\tau}$ (or equivalently $\mathbf{y}_n$) would be considerably more accurate than the other components, if the integer ambiguities were known. This suggests that an accurate estimation of $\mathbf{z}$ can increase the accuracy of the estimate of the state of the system significantly. With this explanation, our treatment of $\mathbf{z}$ is clear. From the state dynamics and the observation equation we first estimate $\mathbf{z}$ and then, with fixed $\mathbf{z}$, we use regular nonlinear filtering methods to estimate the state of the system $\mathbf{x}_t$.

We augment the state $\mathbf{x}_t$ with the integer ambiguity $\mathbf{z}$. Having done that, the state dynamics and the observation have the following form:

$$
\begin{aligned}
d\begin{bmatrix} \mathbf{x}_t \\ \mathbf{z}_t \end{bmatrix} &= \begin{bmatrix} \mathbf{f}_t(\mathbf{x}_t) \\ 0 \end{bmatrix} dt + \begin{bmatrix} G_t(\mathbf{x}_t) \\ 0 \end{bmatrix} d\mathbf{w}_t \\
\mathbf{y}_{n\tau} &= \mathbf{h}_n(\mathbf{x}(n\tau) + J_n\mathbf{z}(n\tau)) + \mathbf{v}_n.
\end{aligned} \tag{5.1}
$$

We assume that the initial distribution of $(\mathbf{x}_0^T, \mathbf{z}_0^T)^T$ is known. Now the state dynamics and the observation have the same form as was studied in Section (3.3). Therefore, we can apply particle filtering to find the conditional probability distribution of the augmented state. This setup is a special case of the setup in Section (3.3). In (5.1) there is no state transition for $\mathbf{z}_t$, therefore, using particle filtering in its original form may not be the best option. Recall that in particle filtering we start with N i.i.d. particles distributed according to the initial distribution. In the resampling part the low probability particles die and the high probability particles produce many particles identical to themselves. Since $\mathbf{z}_t$ does not change, the part of the particles associated to $\mathbf{z}_t$ tends to cover smaller and smaller portions of the state space. In fact, the state space of the integer vectors is defined by the particles at the initial time. This problem can be overcome by modifying the algorithm mentioned in Section (3.3). In the new algorithm, Step 5 is changed in such a way that the particles are the addition of the original particles found by Algorithm 3.3.1, with a random vector. The modification is very important for the integer values, since the integers do not have a dynamics that is driven by a random input. In [34], a similar modification has been used for the regular nonlinear filtering setup. It seems that the convergence results given in [34] can be applied to the current case as well.

Based on the modified algorithm, we simulated a nonlinear filtering problem similar to the problem involved in the GPS system.

In a two dimensional space, three transmitters (imagine three pseudo satel-

lites) are mounted on three known points (2000, 100000), (0, 100000), and (-2000, 100000). The moving object can measure its distance from these transmitters. For each pseudo satellite, two types of measurement are possible: One with high measurement noise and the other with low measurement noise. For the low measurement noise, though, there is an integer ambiguity. The dynamics of the moving object for this example is considered to be in discrete time and linear time invariant. The dynamics and observation equation is given as follows:

$$\begin{pmatrix} x_1 \\ v_1 \\ x_2 \\ v_2 \end{pmatrix}_{n+1} = \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ v_1 \\ x_2 \\ v_2 \end{pmatrix}_n + \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{pmatrix}_n ,$$

$$\begin{aligned} y_n^a &= \|\mathbf{x} - \mathbf{s}_i\| + v_n^a \qquad , i = 1, 2, 3 \\ y_n^b &= \|\mathbf{x} - \mathbf{s}_i\| + n_i + v_n^b \quad , i = 1, 2, 3, \end{aligned}$$

where $\mathbf{x} = (x_1, x_2)^T$, $\mathbf{s}_i$ is the position of pseudo satellite i in two dimensional space, $\Delta t = 0.1$ unit of time, n_i is the integer ambiguity of the pseudo satellite i, and $\mathbf{w} = (w_1, w_2, w_3, w_4)^T$ and $\mathbf{v} = \left(v_1^a, v_1^b, v_2^a, v_2^b, v_3^a, v_3^b\right)^T$ are zero mean white Gaussian noise process with covariance matrices $\Sigma_{\mathbf{w}} = diag\,(1, 0.5, 1, 0.5)$ and $\Sigma_{\mathbf{v}} = diag\,(5, 0.2, 5, 0.2, 5, 0.2)$, respectively. In the simulation, it is assumed that the initial condition for the position is distributed in a square of size 200×200 units squared, symmetric with respect to the origin.

In brief, the simulation can be separated into two parts, initialization and the full non-linear filtering. In the initialization part, we start with the initial probability distribution for (x_1, x_2) and from a series of observations, we find an estimate for the probability distribution of (v_1, v_2). In this part, we do not use the dynamics of the moving object. Using our estimate for the probability distribution

of (x_1, v_1, x_2, v_2) we find the distribution for the integer ambiguity. After this, the initialization is over, and the full non-linear filter is used. There are some minor numerical considerations that we would like to point out. In the Bayes step of the algorithm, the numbers are usually very small, and without proper scaling the original algorithm would not work. In the resampling part, one can use the law of large numbers and regenerate the particles based on their weight without generating random numbers that are time consuming. The result of the simulations are shown in Figures 5.1, 5.2, 5.3, 5.4, 5.5, and 5.6. To display the estimated integers, we simply used the mean value, which is not necessarily the best choice. Of course, since we have the distribution, we can use the MAP estimate of the integers. In this simulation we forced one of the integers to have a jump. Although our algorithm is not designed for these kinds of changes, we see that it can estimate the new integer values. In future, we use special treatment for the times when these kinds of jumps happen. As we can see, the estimates for the integers are reasonably good. The reliability of the estimate for the integers depends on the energy of the noise.

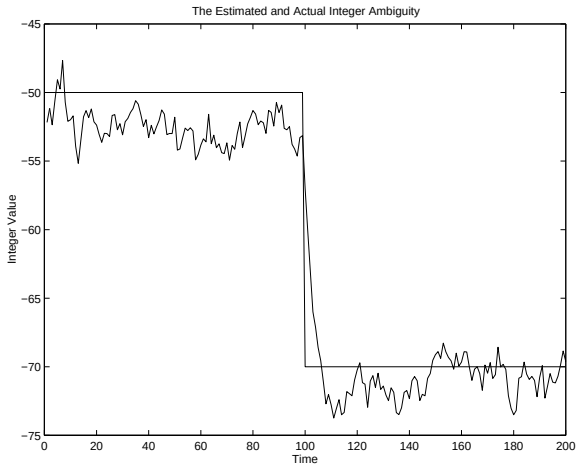

Figure 5.1: Estimated integer ambiguity versus the actual integer ambiguity of pseudo satellite (1). At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

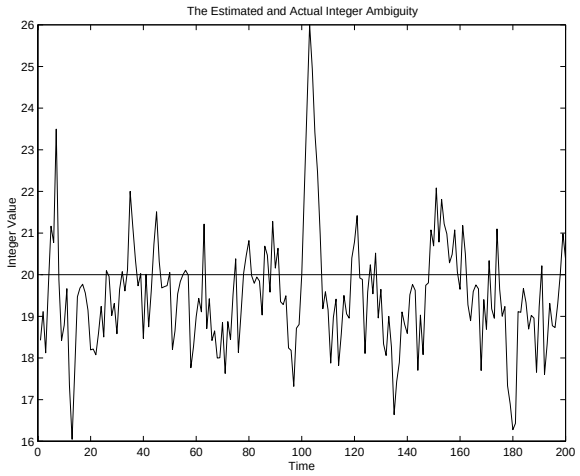

Figure 5.2: Estimated integer ambiguity versus the actual integer ambiguity of pseudo satellite (2). At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

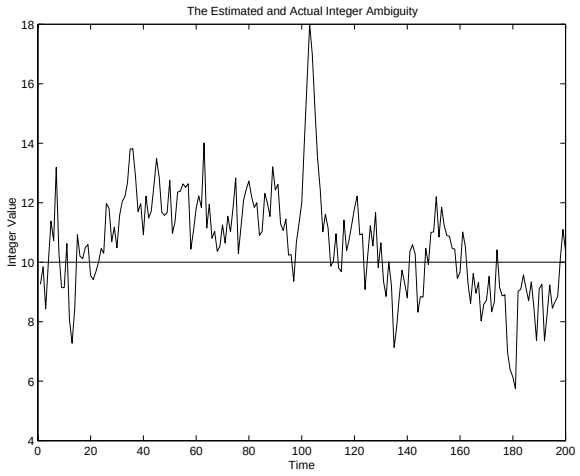

Figure 5.3: Estimated integer ambiguity versus the actual integer ambiguity of pseudo satellite (3). At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

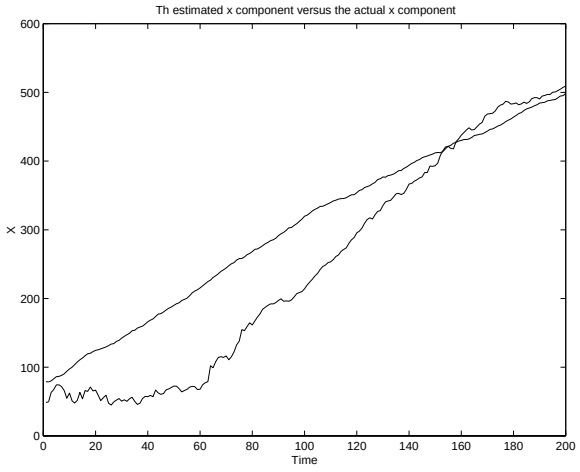

Figure 5.4: Estimated x_1 component versus the actual x_1 component of the position of the car. At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

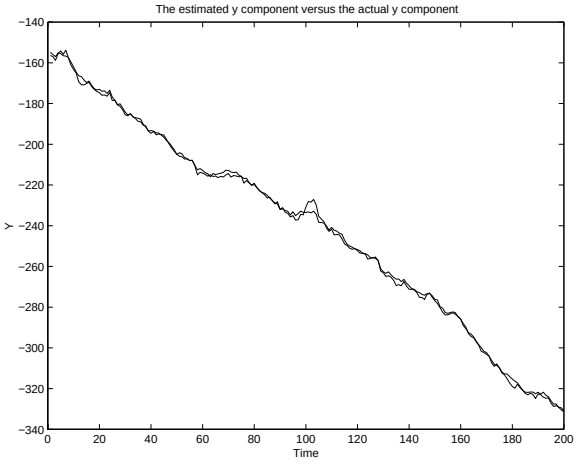

Figure 5.5: Estimated x_2 component versus the actual x_2 component of the position of the car. At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

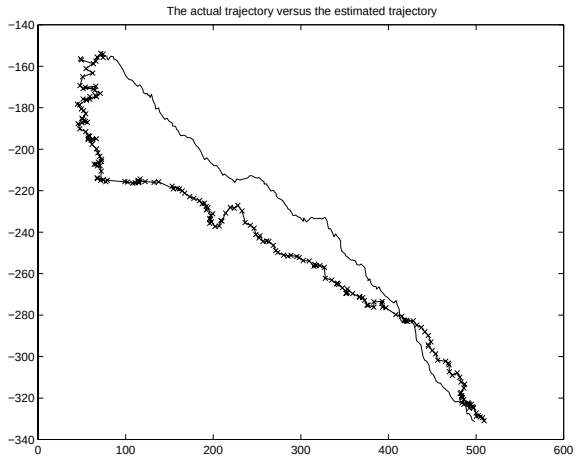

Figure 5.6: Estimated trajectory versus the actual trajectory of the car. At time 100 there is a cycle slip of strength -20 for the measured phase of the carrier from pseudo satellite (1).

5.2 Applications of Projection Particle Filtering for an Integrated INS/GPS

For the rest of this chapter we assume that the integer ambiguity resolution problem is resolved (see Chapter 7). Therefore, we consider the observation equation provided by the *ith* GPS satellite to have the following form:

$$y_i = \rho^i(r_x, r_y, r_z) - \rho^i(b_x, b_y, b_z) + c\delta + v^i \ , \tag{5.2}$$

where $[b_x, b_y, b_z]^T$ is the known base coordinate, δ is the combination of the receiver clock bias in the base and the rover, and v^i is the measurement noise for the *ith* satellite signal.

Here we would like to mention that the nonlinearity in measurement is not only due to the function ρ. As we explain later the integrated INS/GPS requires coordinate transformations between the INS parameters and the GPS parameters, which contributes to the nonlinearity of the measurement.

5.2.1 Coordinate Systems

Parameters of an integrated INS/GPS are expressed in different coordinate systems. In this subsection we intend to introduce these different coordinate systems and the transformation from one to another [21].

ECEF frame

The GPS measurements are given in an Earth Centered Earth Fixed (ECEF) frame. Two different coordinate systems are common for describing the location of a point in the ECEF frame.

parameter	value	Description
a	6378137.0 m	semi major axis
b	6356752.3142 m	semi minor axis
ω_{ie}	7.292115×10^{-5}	angular velocity of the Earth
f	$f = \frac{a-b}{a}$	flatness of the ellipsoid
e	$\sqrt{f(1-f)}$	eccentricity of the ellipsoid

Table 5.1: Definition of the parameters for WGS84 reference frame

The usual rectangular coordinate system $[p_x, p_y, p_z]^T$ for the point p, herein referred to as the ECEF coordinate system, has its x axis extended through the intersection of the prime meridian (0° longitude) and the equator (0° latitude). The z axis extends through the true north pole (i.e. parallel to the Earth's spin axis). The y axis completes the right-handed coordinate system.

The geodetic coordinate system is defined according to the familiar latitude, longitude, and hight coordinate system and is shown by $[p_\lambda, p_\phi, p_h]^T$. For this system of coordinates, the Earth's geoid is approximated by an ellipsoid. The defining parameters for the geoid according to the WGS84 reference frame are given in Table 5.1.

The transformation from the ECEF geodetic to the ECEF rectangular coordinate systems is given as follows

$$\begin{aligned} p_x &= (N + p_h)cos(p_\lambda)cos(p_\phi) \\ p_y &= (N + p_h)cos(p_\lambda)sin(p_\phi) \\ p_z &= (N(1-e^2) + p_h)sin(p_\lambda), \end{aligned} \tag{5.3}$$

where $N = \frac{a}{\sqrt{1-e^2 sin^2(p_\lambda)}}$. The inverse transformation can be derived from (5.3).

Local Geographical frame

It is convenient to express the navigation-frame velocity in the local coordinate system. This coordinate system is rectangular, and it has the x axis, y axis, and the z axis extended through the north, the east, and the down direction, respectively. With this definition for the local geographic coordinate system, the navigation-frame velocity, $[v_N, v_E, v_D]^T$, is related to the geodetic rate vector according to

$$\begin{pmatrix} v_N \\ v_E \\ v_D \end{pmatrix} = \begin{pmatrix} R_\lambda + p_h & 0 & 0 \\ 0 & (R_\phi + p_h)\cos(p_\lambda) & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} \dot{p_\lambda} \\ \dot{p_\phi} \\ \dot{p_h} \end{pmatrix}, \tag{5.4}$$

where $R_\lambda = \frac{a(1-e^2)}{(1-e^2\sin^2(p_\lambda))^{\frac{3}{2}}}$, and $R_\phi = \frac{a}{(1-e^2\sin^2(p_\lambda))^{\frac{1}{2}}}$.

Platform and Body frames

The measurements by accelerometers and gyros are expressed in the platform frame. For simplicity we assume that the axis of the gyros and the axis of the accelerometers are aligned with the axis of the platform frame. Also, we assume that the body frame and the platform frame are aligned, and the center of the coordinate system is the same for both frames. The transformation from body frame to local geographical frame is calculated at every moment, and it depends on the angular rate change measured by the gyros, the rotation of the Earth, and the rotation of the local frame with respect to an inertial frame, all expressed in the body frame. The transform matrix from the platform frame to the local frame is expressed as follows

$$\frac{d}{dt}R_{b2g} = R_{b2g}\Omega^b_{gb}, \tag{5.5}$$

where

$$\Omega_{gb}^{b} = \begin{pmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{pmatrix}, \tag{5.6}$$

and $\omega_{gb}^{b} = [p, q, r]^T$ is the inertial angular rate expressed in the body frame. ω_{gb}^{b} can be expressed as follows

$$\begin{aligned} \begin{pmatrix} p \\ q \\ r \end{pmatrix} &= \begin{pmatrix} \widetilde{p} \\ \widetilde{q} \\ \widetilde{r} \end{pmatrix} + \begin{pmatrix} b_p \\ b_q \\ b_r \end{pmatrix} \\ &\quad -R_{g2b} \left(w_{ie} \begin{pmatrix} \cos(p_\lambda) \\ 0 \\ -\sin(p_\lambda) \end{pmatrix} + \begin{pmatrix} v_E/(R_\phi + p_h) \\ -v_N/(R_\lambda + p_h) \\ v_E \tan(p_\lambda)/(R_\phi + p_h) \end{pmatrix} \right), \end{aligned} \tag{5.7}$$

where $[\widetilde{p}, \widetilde{q}, \widetilde{r}]^T$ is the measured angular rate, and $[b_p, b_q, b_r]^T$ is the bias in the angular rate measurement.

If we assume that in the time interval $[t, t + \delta t]$, Ω_{gb}^{b} is a constant matrix then we have

$$R_{g2b}(t + \delta t) = \exp(-\Omega_{gb}^{b}(t)\delta t) R_{g2b}(t).$$

Since Ω_{gb}^{b} is a skew symmetric matrix, then $\exp(-\Omega_{gb}^{b}(t)\delta t)$ has a simple form:

$$\exp(-\Omega_{gb}^{b}\delta t) = [I + \frac{\sin(\|\omega_{gb}^{b}(t)\delta t\|)}{\|\omega_{gb}^{b}(t)\|}\Omega_{gb}^{b} + \frac{1 - \cos(\|\omega_{gb}^{b}(t)\delta t\|)}{\|\omega_{gb}^{b}(t)\|^2}(\Omega_{gb}^{b})^2].$$

The transformation from the body frame to the local frame, R_{b2g}, is simply the transpose of R_{g2b}, i.e. $R_{b2g} = R_{g2b}^{T}$.

5.2.2 GPS Clock Drift and INS Dynamics

The GPS clock drift and the INS equations are the sources that contribute to the dynamic equation for the integrated INS/GPS.

The INS dynamic equation can be expressed as follows.

$$
\begin{aligned}
d\begin{pmatrix} p_\lambda \\ p_\phi \\ p_h \end{pmatrix} &= \begin{pmatrix} \frac{1}{R_\lambda + p_h} & 0 & 0 \\ 0 & \frac{1}{(R_\phi + p_h)\cos(p_\lambda)} & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} v_N \\ v_E \\ v_D \end{pmatrix} dt \\
d\begin{pmatrix} v_N \\ v_E \\ v_D \end{pmatrix} &= \left[\begin{pmatrix} -\frac{v_E^2}{R_\phi + p_h}\tan(p_\lambda) - 2\omega_{ie}\sin(p_\lambda)v_E + \frac{v_N v_D}{R_\lambda + p_h} \\ \frac{v_E v_N}{R_\lambda + p_h}\tan(\lambda) + \omega_{ie}\sin(p_\lambda)v_N + \frac{v_E v_D}{R_\phi + p_h} + 2\omega_{ie}\cos(p_\lambda)v_D \\ -\frac{v_N^2}{R_\lambda + p_h} - \frac{v_E^2}{R_\phi + p_h} - 2\omega_{ie}\cos(p_\lambda)v_E \end{pmatrix} \right. \\
&\quad \left. + R_{b2g}\left(\begin{pmatrix} \widetilde{a}_u \\ \widetilde{a}_v \\ \widetilde{a}_w \end{pmatrix} + \begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} \right) + \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} \right] dt + d\mathbf{w}_t^v,
\end{aligned}
\tag{5.8}
$$

where $g = 9.780327 m/s^2$ is the gravitational acceleration, $[\widetilde{a}_u, \widetilde{a}_v, \widetilde{a}_w]^T$ is the accelerometer measurement expressed in the body frame, $[b_u, b_v, b_w]^T$ is the accelerometer measurement bias again expressed in the body frame, and $\mathbf{w}^v$ is a vector valued Brownian motion process with zero mean and known covariance matrix. The measurement bias is assumed to have the following dynamics

$$
d\begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} = -a_b \begin{pmatrix} b_u \\ b_v \\ b_w \end{pmatrix} dt + d\mathbf{w}_t^b, \tag{5.9}
$$

where $\mathbf{w}_t^b$ is a vector valued Brownian motion with zero mean and known covariance matrix, and a_b is a small positive constant.

The receiver clock drift, δ_t, is represented by the integration of an exponentially correlated random process ϱ_t [16]

$$
\begin{aligned}
d\varrho_t &= -a_\varrho \varrho_t dt + dw_t^\varrho \\
d\delta_t &= \varrho_t dt,
\end{aligned}
\tag{5.10}
$$

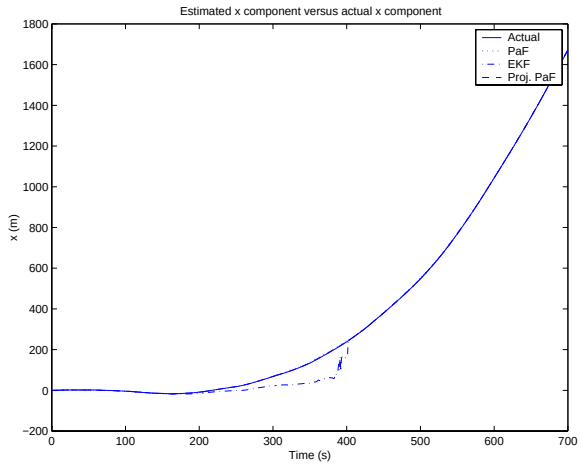

Figure 5.7: Comparison of the estimated and actual x component for three different methods, EKF, particle filtering, and projection particle filtering. For $t < 100$, the number of satellites is 6, for $100 \leq t \leq 400$, the number of satellites is 3, and for $t > 400$, the number of satellites is 4.

with $a_\varrho = 1/500$ and w_t^ϱ is a process of Brownian motion with zero mean and variance $\sigma_\varrho^2 = 10^{-24}$. This dynamic model is typical for a quartz TCXO with frequency drift rate of $10^{-9}s/s$ [16].

5.2.3 Simulation and Results

In this section we present the simulation results for an integrated INS/GPS. Here we apply three different filtering methods, EKF, particle filtering, and projection particle filtering for a specified exponential density. We assumed that R_{g2b} is perfectly known, i.e. the estimation problem regarding the gyro measurements is solved. Therefore, the dimension of the dynamical system in this simulation is

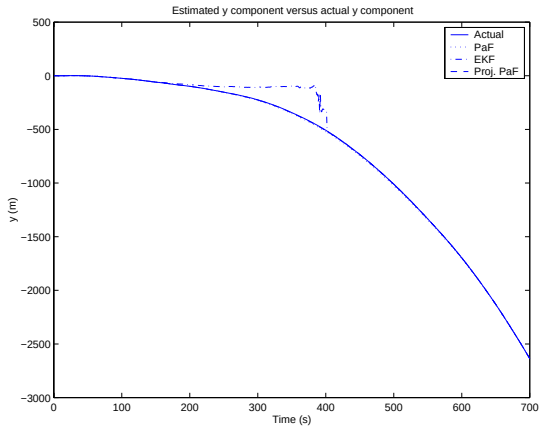

Figure 5.8: Comparison of the estimated and actual y component for three different methods, EKF, particle filtering, and projection particle filtering. For $t < 100$, the number of satellites is 6, for $100 \leq t \leq 400$, the number of satellites is 3, and for $t > 400$, the number of satellites is 4.

eleven. The state of the dynamical system, $\mathbf{x}$, is given as follows

$$\mathbf{x} = [p_\lambda, p_\phi, p_h, v_N, v_E, v_D, b_u, b_v, b_w, \varrho, \delta]^T.$$

The differential equation describing the dynamics of the system is the combination of the differential equation in (5.8), (5.9), and (5.10). Here, we assume that $a_b = 0.001$, and that the covariance matrices for the Brownian motions in the INS dynamic equations, Σ_b and Σ_v, are diagonal. To be more specific, $\Sigma_b = 10^{-6}I$ and $\Sigma_v = 10^{-4}I$, where I is the identity matrix of the right size. The time step we chose for the approximation of the stochastic differential equation is $h = 50\ ms$ and the Gaussian random vector generated in each step has the covariance matrix $\Sigma_h = h\Sigma$, where Σ is the covariance matrix of the combination of all Brownian motions in the dynamics. The observation equation is given in (5.2), where y_i is one component of the observation vector. The dimension of the observation vector

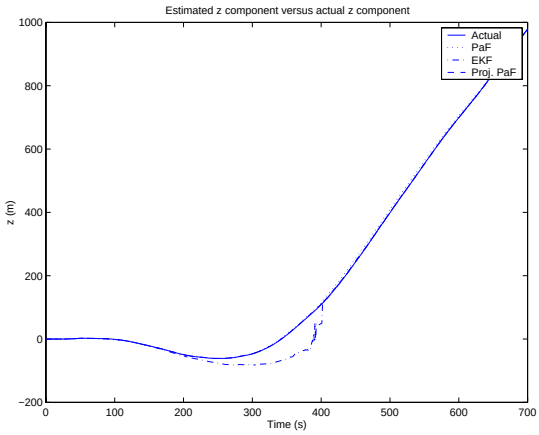

Figure 5.9: Comparison of the estimated and actual z component for three different methods, EKF, particle filtering, and projection particle filtering. For $t < 100$, the number of satellites is 6, for $100 \leq t \leq 400$, the number of satellites is 3, and for $t > 400$, the number of satellites is 4.

is the same as the number of available satellites. In (5.2) the observation is given as a function of the position in the ECEF rectangular coordinate system. Therefore, to be able to write down the observation equation as a function of the state of the system, one needs to use the transform in (5.3).

For this simulation we simply chose an 11 dimensional Gaussian density for the projection particle filtering. This choice of density makes the random vector generation easy and computationally affordable. To be able to use the projection particle filtering, we used maximum likelihood estimation of the parameters of the Guassian density before and after Bayes' correction.

In this simulation, we used two NovAtel RT-2[1] GPS receivers to collect the navigation data on April 2, 2000. From the collected data, we extracted the position

[1]RT-2 is the trademark of NovAtel Incorporated.

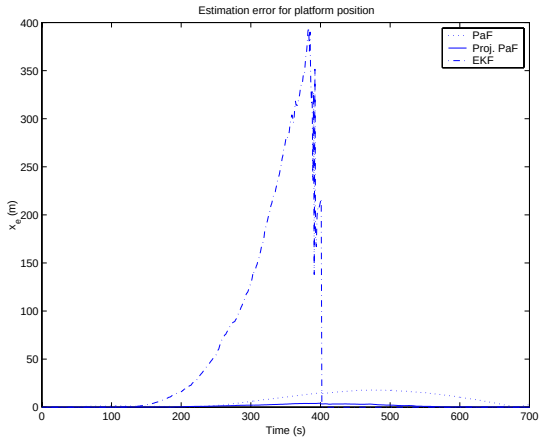

Figure 5.10: The estimation error for the platform position for three different methods, EKF, particle filtering, and projection particle filtering. For $t < 100$, the number of satellites is 6, for $100 \leq t \leq 400$, the number of satellites is 3, and for $t > 400$, the number of satellites is 4.

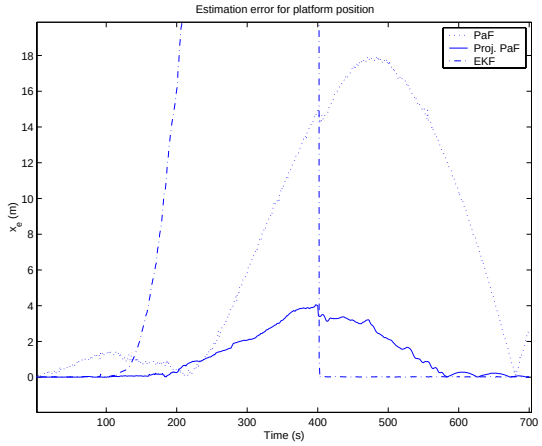

Figure 5.11: Detail of Figure 5.10, where the difference between the projection particle filtering method and the particle filtering method is clear.

information of the satellites, the pseudo range, and the carrier phase measurement noise powers for the L1 frequency. Using the collected information we generated the pseudo range and the carrier phase data for one static and one moving receiver (base and rover, respectively). Here we assume for the carrier phase measurement the integer ambiguity problem is already solved. The movement of the INS/GPS platform was simulation based and the measurement data measured by the accelerometers, the gyros, the GPS pseudo range, and the GPS carrier phase data were generated according to that movement.

In the simulation the GPS receiver starts with 6 satellites. At time $t = 100$, the receiver looses 3 satellites, and it gains one satellite at $t = 400$. We want to emphasize that for instantaneous stand alone positioning GPS requires at least 4 satellites. Figures 5.7-5.9 show the actual and estimated x, y, and z components of the position of the platform in ECEF rectangular system of coordinate. The estimates are given by three methods, EKF, particle filtering, and projection particle filtering. The error of these three methods are plotted in Figure 5.10. From this figure, it can easily be seen that EKF fails to give an acceptable estimate of the position when the number of satellites in view is below four. Unlike EKF, particle filtering and projection particle filtering are successful in providing a reasonable estimate of position. Figure 5.11 is the repeated version of Figure 5.10 with an emphasis on the comparison of the errors between particle filtering and projection particle filtering. For the same number of particles, here 500, the error of the estimate given by the projection particle filtering is smaller than the error for particle filtering. Finally we should mention that whenever the number of visible satellites is more than four, EKF can provide a very good estimate of the position.

Chapter 6

Particle Filtering for a Family of Mixture Densities

In Chapter 4 we introduced a new projection particle filtering method for an exponential family of densities. We proved that if a family of densities exists that is close to the true conditional density, then the error of the estimate given by projection particle filtering can be bounded and this bound depends on the choice of the specific exponential family. Finding such a family is not an easy task. This fact was a motivation for us to study particle filters for a family of mixture densities. Here we assume that the true conditional density is approximated by a linear combination of a finite number of density functions. We can extend the result of this chapter to the approximation of the true conditional density by a set of basis functions. Using this assumption, we replace the empirical distribution in [40] with an estimate that lies in the family. In Theorem 6.1.8 we show results similar to the ones presented in Chapter 4.

6.1 Projection Particle Filtering for a Family of Mixture Densities

We start this section with the definition of a family of mixture densities.

Definition 6.1.1 *Let $\{c_1, \cdots, c_p\}$ be a set of densities defined on $\mathcal{R}^n$, and $\theta \in \mathcal{R}^p$. Then*

$$\mathcal{S}_l = \{p(\cdot, \theta) = \theta^T \mathbf{c}(\cdot), s.t. \sum_{i=1}^{p} \theta_i = 1, \theta_i \geq 0, i = 1, \cdots, p\}$$

is called a family of mixture densities, where $\theta = (\theta_1, \cdots, \theta_p)^T$ and $\mathbf{c} = (c_1, \cdots, c_p)^T$.

For family $\mathcal{S}_l$, function $u(\cdot)$, and random vector $\mathbf{x}$ distributed according to $\sum_{i=1}^{p} \theta_i c_i(\mathbf{x})$ we have

$$Eu(\mathbf{x}) = \sum_{i=1}^{p} \theta_i E_i u(\mathbf{x}),$$

where $E_i(\cdot)$ is the expectation with respect to the density c_i. In particular if $u(\cdot) = \mathbf{c}(\cdot)$, we have

$$E\mathbf{c}(\mathbf{x}) = \beta\theta,$$

where β is a $p \times p$ matrix, and its ij element, $\beta_{ij} = \int c_i(\mathbf{x}) c_j(\mathbf{x}) d\mathbf{x}$. Here, we assume that β^{-1} exists. Therefore, β is positive definite.

If $\mathbf{x}^1, \cdots, \mathbf{x}^N$, are i.i.d. random vectors distributed according to $p(\cdot, \theta)$, then we define the estimate of θ, $\widehat{\theta}$, as follows:

$$\widehat{\theta} = \beta^{-1}(\frac{1}{N} \sum_{i=1}^{N} \mathbf{c}(\mathbf{x}^i)).$$

We know that

$$\theta = \beta^{-1} E\mathbf{c}(\mathbf{x}),$$

therefore, $\widehat{\theta}$ is an unbiased estimate of θ. From the strong law of large numbers it is clear that $\widehat{\theta} \longrightarrow \theta$ as $N \longrightarrow \infty$ with probability one.

We have

$$\theta - \widehat{\theta} = \beta^{-1} \left(E\mathbf{c}(\mathbf{x}) - \frac{1}{N} \sum_{i=1}^{N} \mathbf{c}(\mathbf{x}^i) \right) \ ,$$

therefore, the error estimate for the parameter θ can be bounded as follows:

$$\frac{1}{\lambda_{\max}} E\| E\mathbf{c}(\mathbf{x}) - \frac{1}{N} \sum_{i=1}^{N} \mathbf{c}(\mathbf{x}^i) \| \leq E\|\theta - \widehat{\theta}\| \leq \frac{1}{\lambda_{\min}} E\| E\mathbf{c}(\mathbf{x}) - \frac{1}{N} \sum_{i=1}^{N} \mathbf{c}(\mathbf{x}^i) \|,$$

where $\lambda_{\max}$ and λ_{min} are the maximum and minimum eigenvalues of the matrix β, respectively. It is very reasonable to assume that the variance of $\mathbf{c}(\mathbf{x})$ under $p(\cdot, \theta)$ is finite, then $\exists A > 0$ s.t. $E\|\mathbf{c}(\mathbf{x}) - \frac{1}{N} \sum_{i=1}^{N} \mathbf{c}(\mathbf{x}^i)\| \leq \frac{A}{\sqrt{N}}$.

If θ_1 and θ_2 are two different parameters satisfying the condition in Definition 6.1.1, we have

$$\begin{aligned} E_{\theta_1}\mathbf{c}(\mathbf{x}) - E_{\theta_2}\mathbf{c}(\mathbf{x}) &= \int \left(\theta_1^T \mathbf{c}(\mathbf{x}) \right) \mathbf{c}(\mathbf{x}) d\mathbf{x} - \int \left(\theta_2^T \mathbf{c}(\mathbf{x}) \right) \mathbf{c}(\mathbf{x}) d\mathbf{x} \\ &= \beta(\theta_1 - \theta_2) \ , \end{aligned}$$

therefore,

$$\lambda_{\min} \|\theta_1 - \theta_2\| \leq \|E_{\theta_1}\mathbf{c}(\mathbf{x}) - E_{\theta_2}\mathbf{c}(\mathbf{x})\| \leq \lambda_{\max} \|\theta_1 - \theta_2\|, \tag{6.1}$$

where $E_\theta u(\mathbf{x}) = \int u(\mathbf{x}) \theta^T \mathbf{c}(\mathbf{x}) d\mathbf{x}$. With this introductory explanation, we are ready to introduce particle filtering for a family of mixture densities.

Algorithm 6.1.2 *Particle Filtering for a Family of Mixture Densities.*

- *Step 1 . Initialization*

  - ◇ *Sample* $\mathbf{x}_0^1, \cdots, \mathbf{x}_0^N$, *N i.i.d. random vectors with the density,* $p_0(\mathbf{x})$.

- *Step 2 . Diffusion*

◇ *Find* $\widehat{\mathbf{x}}_{n+1}^1, \cdots, \widehat{\mathbf{x}}_{n+1}^N$ *from the given* $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$*, using the dynamic rule:*

$$d\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_t)dt + G_t(\mathbf{x}_t)d\mathbf{w}_t, \quad i\tau \leq t < (i+1)\tau$$

- *Step 3 . Projection*

$$\widehat{\theta}_{(n+1)^-} = \mathcal{G}(\beta^{-1}(\frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n+1}^i)))$$

where $\mathcal{G}$ *is to make sure that* $\widehat{\theta}$ *satisfies the conditions in Definition 6.1.1, in particular this function can be chosen as follows:*

$$\mathcal{G}(\mathbf{x}) \quad = \quad \begin{cases} \frac{(\mathbf{x})^+}{\|(\mathbf{x})^+\|_1} & \text{if} \quad \|(\mathbf{x})^+\| \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

where $\|\cdot\|_1$ *is the regular norm one, and* $(\cdot)^+ = \max(\cdot, 0)$*.*

- *Step 4 . Use Bayes' Rule*

$$\widehat{\theta}_{n+1} = \mathcal{G}\left(\int \frac{\widehat{\theta}_{(n+1)^-}^T \mathbf{c}(\mathbf{x})\Psi_{n+1}(\mathbf{x})}{\int \widehat{\theta}_{(n+1)^-}^T \mathbf{c}(\mathbf{x})\Psi_{n+1}(\mathbf{x})d\mathbf{x}}\mathbf{c}(\mathbf{x})d\mathbf{x}\right)$$

- *Step 5 . Resample*

◇ *Sample* $\mathbf{x}_{n+1}^1, \cdots, \mathbf{x}_{n+1}^N$ *according to* $p(\mathbf{x}, \widehat{\theta}_{n+1})$*.*

- *Step 6 .* $n \leftarrow n+1$*; go to Step (2).*

In the rest of this section we will prove that under certain conditions, stated later, the error of the estimate associated to the conditional density given by Algorithm 6.1.2 can be bounded. In this chapter we use the same notion for the probability spaces that where used in Chapter 4.

To show our main results in this chapter we need the following assumptions.

A 6.1.3 *For the density in (3.6) there exists a family of densities $\mathcal{S}_l$ such that* $\forall t \in [0,T]$, $\forall u \in \mathcal{F}_{k\kappa}$ $\exists \theta_t^*$ *where* $\|\theta_t^*\|_1 = 1$ *and* $\epsilon > 0$ *such that*

$$\widetilde{E}\|E_{p_t}(u(\mathbf{x})) - E_{\theta_t^*}(u(\mathbf{x}))\| \leq \epsilon \ . \tag{6.3}$$

A 6.1.4 *For $\theta_{n^-}^*$ in (A6.1.3) and $\Psi_n(\mathbf{x})$, $\exists \Psi_n^*(\mathbf{x}) = \overline{\alpha}^T \mathbf{c}(\mathbf{x})$ and $\|\alpha\|_1 = 1$ such that*

- $\forall \theta$ *where* $\|\theta\|_1 = 1$ *and* $\forall u(\cdot) \in \mathcal{F}_{k\kappa}$, $\exists \epsilon > 0$ *such that*

$$\widetilde{E}\|\frac{E_\theta \Psi_n(\mathbf{x})u(\mathbf{x})}{E_\theta \Psi_n(\mathbf{x})} - \frac{E_\theta \Psi_n^*(\mathbf{x})u(\mathbf{x})}{E_\theta \Psi_n^*(\mathbf{x})}\| \leq \epsilon.$$

- $\forall u(\cdot) \in \mathcal{F}_{k\kappa}$, $\exists \epsilon > 0$ *such that*

$$\widetilde{E}\|\frac{E_{\theta_{n^-}^*} \Psi_n^*(\mathbf{x})u(\mathbf{x})}{E_{\theta_{n^-}^*} \Psi_n^*(\mathbf{x})} - \frac{E_{p_{n^-}} \Psi_n(\mathbf{x})u(\mathbf{x})}{E_{p_{n^-}} \Psi_n(\mathbf{x})}\| \leq \epsilon.$$

Fact 6.1.5 $\forall \theta_1, \theta_2$ *where* $\|\theta_1\|_1 = 1$ *and* $\|\theta_2\|_1 = 1$ *we have*

$$\left\| \frac{E_{\theta_1} \Psi_n^*(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta_1} \Psi_n^*(\mathbf{x})} - \frac{E_{\theta_2} \Psi_n^*(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta_2} \Psi_n^*(\mathbf{x})} \right\| \leq M\|\theta_1 - \theta_2\|$$

for some $M > 0$.

Proof: Let matrix A_α and vector b_α be such that $E_\theta \Psi_n^*(\mathbf{x})\mathbf{c}(\mathbf{x}) = \int \alpha^T \mathbf{c}(\mathbf{x})\theta^T \mathbf{c}(\mathbf{x})\mathbf{c}(\mathbf{x})d\mathbf{x} = A_{_\alpha}\theta$, and similarly $E_{\theta_1} \Psi_n^*(\mathbf{x}) = b_{_\alpha}\theta$. Then,

$$\begin{aligned}
\left\| \frac{E_{\theta_1} \Psi_n^*(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta_1} \Psi_n^*(\mathbf{x})} - \frac{E_{\theta_2} \Psi_n^*(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta_2} \Psi_n^*(\mathbf{x})} \right\| &= \left\| \frac{A_\alpha \theta_1}{b_\alpha \theta_1} - \frac{A_\alpha \theta_2}{b_\alpha \theta_2} \right\| \\
&\leq \left\| \frac{A_\alpha \theta_1}{b_\alpha \theta_1} - \frac{A_\alpha \theta_2}{b_\alpha \theta_1} \right\| + \left\| \frac{A_\alpha \theta_2}{b_\alpha \theta_1} - \frac{A_\alpha \theta_2}{b_\alpha \theta_2} \right\| \\
&\leq \frac{\|A_\alpha\|}{b_\alpha \theta_1}\|\theta_1 - \theta_2\| + \frac{\|A_\alpha \theta_2\|\|b_\alpha\|}{b_\alpha \theta_1 b_\alpha \theta_2}\|\theta_1 - \theta_2\| \\
&\leq M\|\theta_1 - \theta_2\| \ ,
\end{aligned}$$

where $M = \max_{\alpha,\theta_1,\theta_2} \frac{\|A_\alpha\|}{b_\alpha\theta_1} + \frac{\|A_\alpha\theta_2\|\|b_\alpha\|}{b_\alpha\theta_1 b_\alpha\theta_2}$ is a finite constant.

$\diamond$

Assume that $\mathbf{x}_n^1, \cdots, \mathbf{x}_n^N$ in Step 2 of Algorithm 6.1.2 are distributed according to $p(\cdot, \widehat{\theta}_n)$. We also assume that

$$\widetilde{E}\|E_{\widehat{\theta}_n}\mathbf{c}(\mathbf{x}) - E_{\theta_n^*}\mathbf{c}(\mathbf{x})\| \leq \delta \quad . \tag{6.4}$$

At the end of the time interval $[n\tau, (n+1)\tau]$ we have

$$\begin{aligned}
&\widetilde{E}\|EE_{\widehat{\theta}_n}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau)) - EE_{\theta_n^*}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau))\| \\
&\qquad = \widetilde{E}\| \int \mathbf{c}(\mathbf{x})p(\mathbf{x},(n+1)\tau,\mathbf{s},n\tau)(p(\mathbf{s},\widehat{\theta}_n) - p(\mathbf{s},\theta_n^*))d\mathbf{x}d\mathbf{s}\| \\
&\qquad = \widetilde{E}\| \int \mathbf{c}(\mathbf{x})p(\mathbf{x},(n+1)\tau,\mathbf{s},n\tau)\mathbf{c}^T(\mathbf{s})(\widehat{\theta}_n - \theta_n^*)d\mathbf{x}d\mathbf{s}\| \\
&\qquad \leq \widetilde{E}\|\widehat{\theta}_n - \theta_n^*\| \int \|\mathbf{c}(\mathbf{x})p(\mathbf{x},(n+1)\tau,\mathbf{s},n\tau)\mathbf{c}^T(\mathbf{s})\|d\mathbf{x}d\mathbf{s} \\
&\qquad \leq L_1\widetilde{E}\|\widehat{\theta}_n - \theta_n^*\| \ ,
\end{aligned}$$

where $p(\mathbf{x},(n+1)\tau,\mathbf{s},n\tau)$ is the transition probability from state $\mathbf{s}$ to state $\mathbf{x}$ in the time interval $[n\tau,(n+1)\tau]$, and $L_1 = \int\int \|\mathbf{c}(\mathbf{x})p(\mathbf{x},(n+1)\tau,\mathbf{s},n\tau)\mathbf{c}^T(\mathbf{s})\|d\mathbf{x}d\mathbf{s} > 0$ is a constant, possibly depending on n. Therefore, using (6.1) $\exists K_1 > 0$ s.t.

$$\widetilde{E}\|EE_{\widehat{\theta}_n}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau)) - EE_{\theta_n^*}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau))\| \leq K_1\delta. \tag{6.5}$$

On the other hand, from (4.4) we have

$$\widetilde{E}\|EE_{\widehat{\theta}_n}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau)) - \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n\tau,\mathbf{s}^i}((n+1)\tau))\| \leq Kh^2 + \frac{k'}{N^{1/2}} \quad , \tag{6.6}$$

where $\mathbf{s}^i = \mathbf{x}_n^i$.

From Assumption (A6.1.3) we have

$$\begin{aligned}
&\widetilde{E}\|EE_{\theta_n^*}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau)) - E_{p_n}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau))\| \\
&\qquad = E\| \int \mathbf{q}(\mathbf{s})(p(\mathbf{s},\theta_n^*) - p_n(\mathbf{s}))d\mathbf{s}\| \\
&\qquad \leq \epsilon \ ,
\end{aligned} \tag{6.7}$$

where $\mathbf{q}(\mathbf{s}) = \int \mathbf{c}(\mathbf{x}) p(\mathbf{x}, (n+1)\tau, \mathbf{s}, n\tau) d\mathbf{x}$ is assumed to be in $\mathcal{F}_{k\kappa}$ [1] .

We have

$$EE_{p_n}\mathbf{c}(\mathbf{x}_{n\tau,\mathbf{s}}((n+1)\tau)) = E_{p_{(n+1)^-}}\mathbf{c}(\mathbf{x}).$$

Also from Assumption (A6.1.3) we know that $\exists \theta^*_{(n+1)^-}$ s.t

$$\widetilde{E}\|E_{\theta^*_{(n+1)^-}}\mathbf{c}(\mathbf{x}) - E_{p_{(n+1)^-}}\mathbf{c}(\mathbf{x})\| \leq \epsilon. \tag{6.8}$$

Therefore, from (6.5), (6.6), (6.7), and (6.8) we get

$$\widetilde{E}\|E_{\theta^*_{(n+1)^-}}\mathbf{c}(\mathbf{x}) - \frac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n\tau,\mathbf{s}^i}((n+1)\tau))\| \leq K_1\delta + 2\epsilon + Kh^2 + \frac{k'}{N^{1/2}}. \tag{6.9}$$

This implies that

$$\begin{aligned}\widetilde{E}\|\theta^*_{(n+1)^-} - \beta^{-1}(\tfrac{1}{N}\sum_{i=1}^{N}\mathbf{c}(\widehat{\mathbf{x}}_{n\tau,\mathbf{s}^i}((n+1)\tau)))\| \\ \leq 1/\lambda_{\min}(K_1\delta + 2\epsilon + Kh^2 + \tfrac{k'}{N^{1/2}}).\end{aligned} \tag{6.10}$$

The following fact is needed for proof of the theorem that will be presented later.

Fact 6.1.6 *For positive random vector $\alpha \in \mathcal{R}^p$, assume $\|\alpha\|_1 = 1$. Also assume that $\beta \in \mathcal{R}^p$ is a random vector such that $(\beta)^+ \neq 0$. Then, if*

$$E\|\alpha - \beta\| \leq \epsilon,$$

then

$$E\|\alpha - \mathcal{G}(\beta)\| \leq k\epsilon,$$

where $k > 0$ possibly depending on p.

Proof: α is a positive vector, therefore, $E\|\alpha - \beta\| \leq \epsilon$ implies $E\|\alpha - (\beta)^+\| \leq \epsilon$. Since all norms are equivalent in finite spaces, $\exists L > 0$ such that

$$E\|\alpha - (\beta)^+\|_1 \leq L\epsilon.$$

[1] This is a very mild condition.

Therefore,

$$E|\ \|(\beta)^+\|_1 - 1| \leq L\epsilon.$$

On the other hand, we have

$$\begin{aligned} \|\alpha - \tfrac{(\beta)^+}{\|(\beta)^+\|_1}\| &\leq \|\alpha - (\beta)^+\| + \|(\beta)^+ - \tfrac{(\beta)^+}{\|(\beta)^+\|_1}\| \\ &\leq \|\alpha - (\beta)^+\| + \tfrac{\|(\beta)^+\|}{\|(\beta)^+\|_1} \ \mid \|(\beta)^+\|_1 - 1 \mid. \end{aligned}$$

Therefore,

$$E\|\alpha - \frac{(\beta)^+}{\|(\beta)^+\|_1}\| \leq k\epsilon,$$

where $k = L + 1$.

$\diamond$

In Algorithm 6.1.2 we know $\widehat{\theta}_{(n+1)^-} = \mathcal{G}(\beta^{-1}(\frac{1}{N}\sum_{i=1}^{N} \mathbf{c}(\widehat{\mathbf{x}}_{n\tau,\mathbf{s}^i}((n+1)\tau))))$, therefore, by using Fact 6.1.6 and (6.10), we can conclude that $\exists L_2 > 0$ such that

$$E\|\theta^*_{(n+1)^-} - \widehat{\theta}_{(n+1)^-}\| \leq L_2(K_1\delta + 2\epsilon + Kh^2 + \frac{k'}{N^{1/2}}). \tag{6.11}$$

We assume that $\mathbf{c}(\cdot) \in \mathcal{F}_{k\kappa}$, therefore, from Assumption (A6.1.4), Fact 6.1.5, and (6.11) we have

$$\begin{aligned} \widetilde{E}\left\| \frac{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} - \frac{E_{\widehat{\theta}_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} \right\| \\ \leq \zeta_1\delta + \zeta_2\epsilon + \zeta_3 h^2 + \zeta_4 N^{-1/2} \end{aligned} \tag{6.12}$$

for some $\zeta_1, \zeta_2, \zeta_3, \zeta_4 > 0$.

On the other hand, from Assumption (A6.1.4) we have

$$\begin{aligned} &\widetilde{E}\left\| \frac{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} - \frac{E_{p_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})\mathbf{c}(\mathbf{x})}{E_{p_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} \right\| \\ &= \widetilde{E}\left\| \frac{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})u(\mathbf{x})}{E_{\theta^*_{(n+1)^-}} \Psi_{n+1}(\mathbf{x})} - E_{p_{(n+1)}}\mathbf{c}(\mathbf{x}) \right\| \\ &\leq 2\epsilon. \end{aligned} \tag{6.13}$$

In Step 4 of Algorithm 6.1.2 we have

$$\widehat{\theta}_{n+1} = \mathcal{G}\left(\frac{E_{\widehat{\theta}_{(n+1)^-}}\mathbf{c}(\mathbf{x})\Psi_{n+1}(\mathbf{x})}{E_{\widehat{\theta}_{(n+1)^-}}\Psi_{n+1}(\mathbf{x})}\right),$$

therefore, using (6.12), (6.13), and Fact 6.1.6 we can conclude that

$$E\|E_{\widehat{\theta}_{(n+1)}}\mathbf{c}(\mathbf{x}) - E_{\theta^*_{(n+1)}}\mathbf{c}(\mathbf{x})\| < \rho_1\delta + \rho_2\epsilon + \rho_3 h^2 + \rho_4 N^{-1/2},$$

for some positive $\rho_1, \rho_2, \rho_3, \rho_4$.

We summarize the results of this section in the following theorem.

Lemma 6.1.7 *For System (3.3) assume (A3.1.1), (A3.1.2), (A6.1.3), and (A6.1.4). We also assume* $\mathbf{c}(\cdot) \in \mathcal{F}_{k\kappa}$ *and the conditions in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$*. Then in* `Algorithm 6.1.2` *with approximation (4.3), if*

$$E\|E_{\widehat{\theta}_n}\mathbf{c}(\mathbf{x}) - E_{\theta^*_n}\mathbf{c}(\mathbf{x})\| \leq \delta$$

then $\exists \varrho_1^n, \varrho_2^n, \varrho_3^n, \varrho_4^n$ *positive such that*

$$E\|E_{\widehat{\theta}_{(n+1)}}\mathbf{c}(\mathbf{x}) - E_{\theta^*_{(n+1)}}\mathbf{c}(\mathbf{x})\| < \varrho_1^n\delta + \varrho_2^n\epsilon + \varrho_3^n h^2 + \varrho_4^n N^{-1/2}.$$

Lemma 6.1.7 is the building block for our main result in the next theorem.

Theorem 6.1.8 *For System (3.3) assume (A3.1.1), (A3.1.2), (A6.1.3), and (A6.1.4). We also assume* $\mathbf{c}(\cdot) \in \mathcal{F}_{k\kappa}$ *and the conditions in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$*. Then in* `Algorithm 6.1.2` *with approximation (4.3), if*

$$E\|E_{\widehat{\theta}_0}\mathbf{c}(\mathbf{x}) - E_{\theta^*_0}\mathbf{c}(\mathbf{x})\| \leq \delta$$

then for all $t \in [0, T]$, $\exists \varrho_1, \varrho_2, \varrho_3, \varrho_4$ *positive such that*

$$E\|E_{\widehat{\theta}_t}\mathbf{c}(\mathbf{x}) - E_{\theta^*_t}\mathbf{c}(\mathbf{x})\| < \varrho_1\delta + \varrho_2\epsilon + \varrho_3 h^2 + \varrho_4 N^{-1/2}.$$

Here we would like to make some remarks.

- In Theorem 6.1.8 four different factors can increase the accuracy of the estimation method:
  - N: the number of particles. When $N \longrightarrow \infty$ the error due to the limited number of particles disappears.
  - h: the step size in the solution of the stochastic differential equation. If instead of a differential equation the system dynamics is given by a difference equation this error disappears. Also, when $h \longrightarrow 0$ the error due to the approximate solution for the stochastic differential equation goes to zero.
  - ϵ: the closeness of the true conditional density to the family. A smaller ϵ means a more accurate family of densities.
  - δ: the initial estimate. It is clear that a better initial estimate of the density enhances the estimate of the density for the time $t \in [0, T]$
- An immediate result of Theorem 6.1.8 can be summarized as follows:

  Corollary 6.1.9 *For System (3.3) assume (A3.1.1), (A3.1.2), (A6.1.3), and (A6.1.4). We also assume the conditions in* `Theorem 4.1.4` *with* $\mathbf{c}(\mathbf{x})$ *replacing* $u(\mathbf{x})$*. Then in* `Algorithm 6.1.2` *with approximation (4.3), if*

  $$E\|\widehat{\theta}_0 - \theta_0^*\| \leq \delta$$

  then $\exists \iota_1, \iota_2, \iota_3, \iota_4$ *positive such that*

  $$E\|\widehat{\theta}_t - \theta_t^*\| < \iota_1\delta + \iota_2\epsilon + \iota_3 h^2 + \iota_4 N^{-1/2},$$

  for all $t \in [0, T]$.

6.2 Discussion

In Chapter 4 we used an exponential family of densities for approximating the conditional density. We have used this family (in the context of projection particle filtering) for position estimation in an integrated INS/GPS (Chapter 5), and also for integer ambiguity resolution for a carrier phase differential GPS (Chapter 7). Although in both cases we have been able to achieve very good results, applying projection particle filtering for an exponential family of densities does not seem to be a trivial task for general cases, where we don't have any idea about suitable exponential family. In fact, finding the proper exponential family for a specific problem is quite challenging [11].

In this Chapter, we have chosen a mixture of densities to approximate the conditional density. The components of this mixture may be viewed as a type of basis functions. In [15] an approach different but close to our approach was used for a tracking problem. In that approach, the components of the family of mixture densities are allowed to change. The new components are calculated according to the discrete time dynamics. Using the same method for nonlinear continuous time dynamics is not efficient, because the conditional density of the state given the initial condition should be calculated in order to find the new components of the family of mixture densities. This is equivalent to solving the forward Kolmogorov equation.

In our future work we intend to use the method introduced in this chapter for position estimation in an integrated INS/GPS. We expect that a performance similar to the one in Chapter 5 can be achieved, with lower computational burden.

Chapter 7

Integer Ambiguity Resolution Using Particle Filtering

Wherever possible, using differential GPS allows users to have a more accurate measurement. In fact, a good portion of the positioning error can be removed from the estimation using this method [53]. This is due to the fact that the error in GPS navigation data has a strong spatial correlation, and this error can be removed by comparison of measurements from two receivers that are relatively close to each other. A significant improvement in positioning accuracy is possible if one can measure the carrier phase of the GPS signal. With today's technology it is possible to measure the phase of the carrier within 10^{-3} modulo an integer number of full cycles [3]. Unfortunately, for positioning purposes this is not enough and one needs the exact phase difference between the transmitted and received signal to estimate the position. As mentioned in previous chapters, the difference between the measured and the actual phase, an unknown integer times 2π, is called integer ambiguity [28]. Resolving this ambiguity has been shown to be quite challenging.

The available integer ambiguity resolution methods are mostly based on a rough

estimation of the integer ambiguity and a search method to find the correct integer value [3]. In the LAMBDA method [52], using a least square estimation technique, first a float solution for the integer ambiguity is found. Then through a search method the integer vector that minimizes the variance of the error is estimated. If the covariance matrix associated to the integer solution is diagonal, the integer vector that minimizes the variance of the error is an integer vector whose elements are closest to the elements of the float solution vector. In practical problems that is not the case, therefore, a search for the integer solution is unavoidable. In Figure 7.1 a two dimensional integer least square problem is shown. In this figure the float solution is shown by $\times$ and the surfaces with the same error are shown by solid lines. It can be seen that the nearest integer vector is not the integer vector that minimizes the error. In a high dimensional problem finding the solution for the integer least square problem is quite challenging and the search space for the solution could be quite large.

The idea in the LAMBDA method is to find a transformation that maps integer vectors to integer vectors and at the same time maps the covariance matrix to a matrix that is diagonal or dominantly diagonal. Although finding this transformation has been shown to be NP complete, a suboptimal implementation of this method is proven to reduce the size of the search space effectively [52].

In our method, we first approximate the conditional probability density of the position of the rover (mobile GPS receiver) given the double difference measurement for the pseudo range observable. This density is used for the initialization that leads to the conditional pmf of the integer ambiguity given the double difference carrier phase measurements.

In the initialization part we use particle filtering as the tool for the approxima-

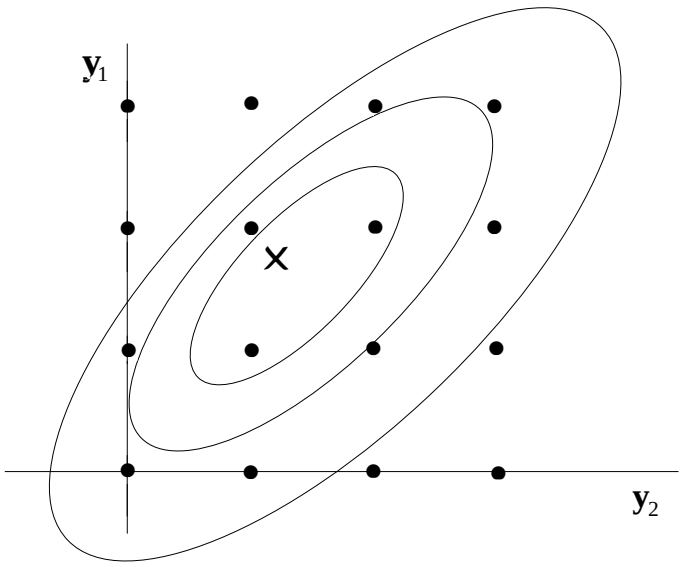

Figure 7.1: Example where the nearest integer vector and the integer vector that minimizes the error are far apart.

tion of the conditional distribution. To find the approximate conditional pmf of the integer ambiguities, given the double difference carrier phase measurements, we use a modified version of particle filtering. Since the set of the integer ambiguities is a discrete set and the size of this set/space is large, using regular particle filtering might not give the proper result. To overcome this problem, instead of the empirical pmf for the particles, we approximate the conditional pmf with an exponential form, in particular with a Gaussian shape. The filtering method here is very similar to the regular particle filtering method explained above. The only difference is that after applying Bayes' Rule, we use MLE to find the parameters of the exponential probability mass function. This pmf is used for generating new particles. This is analogous to the method presented in Chapter 4.

Since the noise power in the carrier phase measurement is very small compared

to the pseudo range measurements [53], and for practical purposes the number of particles is small compared to the size of the integer set, it is very likely that one integer vector attracts all the particles and ends up with probability equal to one. To avoid this problem we start the algorithm assuming high power noise for the carrier phase measurement, and as time increases we reduce the noise power. This technique is similar to *simulated annealing* [54] which is widely used in stochastic optimization techniques. Reducing the noise power is like the cooling process in the simulated annealing method.

In Chapter 5 we studied the problem of position estimation in the presence of an integer uncertainty by augmenting the state to include the unknown integer vector. We used the following system of equations for a moving object with nonlinear dynamics and observations similar to carrier phase differential GPS.

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{f}_n(\mathbf{x}_n) + G_n(\mathbf{x}_n)\mathbf{w}_n \\ \mathbf{y}_n &= \mathbf{h}_n(\mathbf{x}_n) + J_n\mathbf{z} + \mathbf{v}_n, \end{aligned}$$

where $\mathbf{z}$, the integer ambiguity, is a random integer vector, i.e. $\mathbf{z} \in \mathcal{Z}^m$ and J_n has the proper dimension. Vector $\mathbf{z}$ is assumed to be constant in time. One way of treating the integer ambiguity is augmenting the state $\mathbf{x}$ with the integer $\mathbf{z}$. In this case,we have

$$\begin{aligned} \begin{bmatrix} \mathbf{x}_{n+1} \\ \mathbf{z}_{n+1} \end{bmatrix} &= \begin{bmatrix} \mathbf{f}_n(\mathbf{x}_n) \\ \mathbf{z}_n \end{bmatrix} + \begin{bmatrix} G_n(\mathbf{x}_n) \\ 0 \end{bmatrix} \mathbf{w}_n \\ \mathbf{y}_n &= \mathbf{h}_n(\mathbf{x}_n) + J_n\mathbf{z}_n + \mathbf{v}_n \ . \end{aligned} \tag{7.1}$$

We used particle filtering to estimate the integer ambiguity as well as the position of a moving object in a two dimensional space. Although the results there were reasonably good, we had to use a very large number of particles for better estimation results. In this Chapter we use another approach. We use the results

of Theorems 4.1.6 and 4.2.7 and we apply a method similar to Algorithm 4.1.1. In this approach, first we estimate the integer uncertainty and then we use the estimated integer for accurate positioning. The details of this approach are given in the following sections.

7.1 Rationale

In Algorithm 4.1.1 we proposed a particle filtering method for exponential families of densities. Here we show that, that algorithm is applicable to integer ambiguity estimation. To support our claim we will give some simulation results with GPS data in the following sections, but we also want to justify why our claim is reasonable. For the sake of argument, we assume that the ambiguity is real, i.e. $\mathbf{n} \in \mathcal{R}^m$, so we can assign a probability density function to it. In the rest of this section we go through an approximate calculation of the probability density function of the real valued ambiguity given the observation and we show that the Gaussian density is a good candidate for the exponential family of densities.

Consider the measurement model in (2.1) and (2.2). The observation is the result of a double differencing from a possibly moving rover and a static base. We assume that during the observation no cycle slip happens. We seek to estimate the conditional probability density of the real valued ambiguity given the observations up to time $n+1$, i.e.

$$p(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1}) = \frac{p(\phi_{n+1}, p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n)p(\mathbf{n}|\Phi_1^n, P_1^n)}{\int_{\mathbf{n}} p(\phi_{n+1}, p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n)p(\mathbf{n}|\Phi_1^n, P_1^n)d\mathbf{n}} \quad , \tag{7.2}$$

where $\Phi_1^n = \{\phi_1, \phi_2, \cdots, \phi_n\}$ and $P_1^n = \{p_1, p_2, \cdots, p_n\}$ are the observation sets up to and including time n. We want to show that if $p(\mathbf{n}|\Phi_1^n, P_1^n)$ is Gaussian, then $p(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1})$ is approximately Gaussian. To show this, we only need to show

that $p(\phi_{n+1}, p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n)$ has the following form:

$$p(\phi_{n+1}, p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n) = \alpha \exp(-\frac{1}{2}(\mathbf{n}-\beta)^T\Gamma^{-1}(\mathbf{n}-\beta)),$$

for some α, β, and Γ that do not depend on $\mathbf{n}$.

For our integer ambiguity resolution method, we assume that no information about the dynamics of the receiver is available. Therefore,

$$\begin{aligned} p(\phi_{n+1}, p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n) &= p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n)p(p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n) \\ &= p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n)p(p_{n+1}). \end{aligned}$$

For $p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n)$, we have

$$\begin{aligned} p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n) &= \int p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n, \mathbf{x}_{n+1}) \\ &\quad p(\mathbf{x}_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n)d\mathbf{x}_{n+1} \\ &= \int p(\phi_{n+1}|\mathbf{n}, \mathbf{x}_{n+1})p(\mathbf{x}_{n+1}|p_{n+1})d\mathbf{x}_{n+1}. \end{aligned} \tag{7.3}$$

From (2.2), we get

$$\begin{aligned} p(\phi_{n+1}|\mathbf{n}, \mathbf{x}_{n+1}) &= \kappa_1 \exp\left(-\tfrac{1}{2}(\phi_{n+1} - \rho(\mathbf{x}_{n+1}) - \mathbf{n})^T\right. \\ &\quad \left.\Sigma_\phi^{-1}(\phi_{n+1} - \rho(\mathbf{x}_{n+1}) - \mathbf{n})\right), \end{aligned} \tag{7.4}$$

where Σ_ϕ is the covariance matrix of the double difference carrier phase observation noise, $\rho(\mathbf{x}_{n+1})$ is the vector of double difference true range, and κ_1 is a normalizing factor. Here we emphasize that the norm of Σ_ϕ is very small, therefore, in (7.3) the contribution of the region that is outside a small neighborhood of $\mathbf{x}^*$, the point that maximizes the argument of the exponent in (7.4), is negligible. This justifies the approximation of $p(\phi_{n+1}|\mathbf{n}, \mathbf{x}_{n+1})$ by linearization, i.e. we get

$$\begin{aligned} p(\phi_{n+1}|\mathbf{n}, \mathbf{x}_{n+1}) &\approx \kappa_1 \exp\left(-\tfrac{1}{2}(\phi_{n+1} - \rho(\mathbf{x}^*) - \mathbf{n} - A\Delta\mathbf{x}_{n+1})^T\right. \\ &\quad \left.\Sigma_\phi^{-1}(\phi_{n+1} - \rho(\mathbf{x}^*) - \mathbf{n} - A\Delta\mathbf{x}_{n+1})\right), \end{aligned} \tag{7.5}$$

where $\mathbf{x}_{n+1} = \mathbf{x}^* + \Delta\mathbf{x}_{n+1}$, and $A = \frac{d\rho(\mathbf{x})}{d\mathbf{x}}|_{\mathbf{x}=\mathbf{x}^*}$.

On the other hand, from (2.1), we have

$$p_{n+1} = \rho(\mathbf{x}_{n+1}) + \epsilon_{n+1}.$$

Therefore, after linearizing $\rho(\mathbf{x}_{n+1})$, and using the generalized inverse of A, we get

$$\Delta\mathbf{x}_{n+1} \approx (A^TA)^{-1}A^T(p_{n+1} - \rho(\mathbf{x}^*)) - (A^TA)^{-1}A^T\epsilon_{n+1}.$$

Here we assume that A is full rank, i.e. a sufficient number of satellites with acceptable geometry is available. Therefore,

$$p(\Delta\mathbf{x}_{n+1}|p_{n+1}) \approx \kappa_2 \exp(-\frac{1}{2}(D - \Delta\mathbf{x}_{n+1})^T\Upsilon^{-1}(D - \Delta\mathbf{x}_{n+1})), \tag{7.6}$$

where $D = (A^TA)^{-1}A^T(p_{n+1} - \rho(\mathbf{x}^*))$, $\Upsilon = (A^TA)^{-1}A^T\Sigma_pA(A^TA)^{-1}$, Σ_p is the covariance matrix of the double difference code measurement noise, and κ_2 is a normalizing factor.

From (7.3),(7.5) and (7.6), we get

$$\begin{aligned} p(\phi_{n+1}|p_{n+1}, \mathbf{n}, \Phi_1^n, P_1^n) &= \int p(\phi_{n+1}|\mathbf{n}, \Delta\mathbf{x}_{n+1})p(\Delta\mathbf{x}_{n+1}|p_{n+1})d\Delta\mathbf{x}_{n+1} \\ &\approx \alpha \exp(-\tfrac{1}{2}(\mathbf{n} - \beta)^T\Gamma^{-1}(\mathbf{n} - \beta)), \end{aligned}$$

where α, β, and Γ only depend on the matrices Σ_ϕ, Σ_p, and A, and the vectors p_{n+1}, $\rho(\mathbf{x}^*)$, and ϕ_{n+1}. Since $p(p_{n+1}|\mathbf{n}, \Phi_1^n, P_1^n) = p(p_{n+1})$ does not depend on $\mathbf{n}$, we can claim that the conditional density $p(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1})$ is approximately Gaussian.

In the above, we did not use the fact that $\mathbf{n}$ is a real valued vector. Therefore, by using a similar argument, we can claim that, if for an integer vector, $\mathbf{n}$, the pmf, $P(\mathbf{n}|\Phi_1^n, P_1^n)$, has a Gaussian shape, then the pmf, $P(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1})$, also has a Gaussian shape.

7.2 Particle Filtering for Gaussian Shaped Distributions

Using the justification in Section 7.1, we can replace the empirical distribution of the particle filtering by an exponential family with a Gaussian shape. Algorithm 7.2.1 takes this modification into account.

Algorithm 7.2.1 *Particle Filtering for a Gaussian Shaped Distribution.*

- *Step 1 . Initialization*

  ◇ *Sample* $\mathbf{n}_0^1, \cdots, \mathbf{n}_0^N$, *N i.i.d. random variable with the distribution,* $P_0(\mathbf{n})$.

- *Step 2 . New measurement and Bayes' Rule*

$$P^N(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1}) = \frac{\frac{1}{N}\sum_{j=1}^{N} \delta_{\mathbf{n}_n^j}(\mathbf{n}) \cdot p(\phi_{n+1}, p_{n+1}|\mathbf{n}_n^j)}{\frac{1}{N}\sum_{j=1}^{N} \delta_{\mathbf{n}_n^j}(\mathbf{n}_n^j) \cdot p(\phi_{n+1}, p_{n+1}|\mathbf{n}_n^j)}$$

- *Step 3 . Find the mean and the covariance estimates for* $P^N(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1})$.

$$\begin{aligned} \bar{\mathbf{n}}_{n+1} &= \sum_{i=1}^{N} P^N(\mathbf{n}_n^i|\Phi_1^{n+1}, P_1^{n+1})\mathbf{n}_n^i \\ \Sigma_{\mathbf{n}_{n+1}} &= \sum_{i=1}^{N} P^N(\mathbf{n}_n^i|\Phi_1^{n+1}, P_1^{n+1})(\mathbf{n}_n^i - \bar{\mathbf{n}}_n)(\mathbf{n}_n^i - \bar{\mathbf{n}}_n)^T \end{aligned}$$

- *Step 4 . Resample*

  ◇ *Sample real valued* $\widehat{\mathbf{n}}_{n+1}^1, \cdots, \widehat{\mathbf{n}}_{n+1}^N$ *according to* $p(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1})$.

  where

$$p(\mathbf{n}|\Phi_1^{n+1}, P_1^{n+1}) = \frac{\exp\left(\frac{-1}{2}(\mathbf{n} - \bar{\mathbf{n}}_{n+1})^T \ \Sigma_{\mathbf{n}_{n+1}}^{-1} \ (\mathbf{n} - \bar{\mathbf{n}}_{n+1})\right)}{\sqrt{(2\pi)^m \det(\Sigma_{\mathbf{n}_{n+1}})}}$$

- *Step 5 .* $\mathbf{n}_{n+1}^i = \mathbf{g}(\widehat{\mathbf{n}}_{n+1}^i)$*, for* $i = 1, \cdots, N$*.* $\mathbf{g}(\cdot)$ *is a rounding function.*

- *Step 6 .* $n \leftarrow n+1$*; go to Step (2).*

where $\delta_{\mathbf{v}}(\mathbf{w}) = 1$ if $\mathbf{w} = \mathbf{v}$, and 0 otherwise.

In Algorithm 7.2.1, if the number of particles is small, a bad initialization causes significant estimation error. To overcome this problem one can increase the number of particles and/or choose the initialization carefully. Increasing the number of particles increases the computational cost which is not desirable. Therefore, choosing a proper initialization is of great importance.

In the integer ambiguity resolution problem, we first initialize the conditional pmf of the integer ambiguity using the pseudo range measurement. Since the noise power of the pseudo range measurement is significantly larger than the noise power of the carrier phase measurement, it is very likely that one of the integer vectors, that has probability greater than the others, ends up with probability one and the rest of integer vectors end up with zero probability. To avoid this, we alter the covariance matrix for the carrier phase measurement and the covariance matrix for $p(\mathbf{n}|\Phi_1^n)$as follows:

$$\begin{aligned} \widehat{\Sigma}_n &= \Sigma_n + n^{-2} I \\ \widehat{\Sigma}_{\mathbf{n}_n} &= (1 + \tfrac{\alpha}{n}) \Sigma_{\mathbf{n}_n}, \end{aligned} \tag{7.7}$$

where α is a constant coefficient. The idea of changing the covariance matrices is borrowed from the simulated annealing technique. Similar to the temperature decrease in that technique, here we decrease the additional uncertainty added to the data as time grows. The change of the covariance matrix in (7.7) is not unique, but the general form of (7.7) should be kept.

Once we have the conditional pmf of the integer ambiguity given the observations, the estimate for integer ambiguity can be obtained by finding the point

where the conditional pmf is maximum. This is equivalent to finding the MAP estimate of the integer ambiguity.

7.3 Simulations and Results

Using the data described in Section 5.2.3, we generated the pseudo range and carrier phase data for one static and one moving receiver in the same way as was performed in Section 5.2.3. To be able to check our method we added an artificial integer ambiguity to the simulated data. We chose a three dimensional random walk dynamics with nonzero mean speed, $(x, y, z)_{n+1}^T = (x, y, z)_n^T + (2, 1, 1) + 2\epsilon_n$, where ϵ_n is a three dimensional zero mean Gaussian random vector with unit power. The spatial and temporal units are assumed to be meter and second, respectively.

We applied the method discussed in Section 7.2, for different numbers of measured epochs. For each of these cases we simulated 1000 different trials and we counted the number of times that the algorithm doesn't find the correct integer vector. We call these incorrect outcomes error of the estimate. The results of these experiments are summarized in Table 7.1. For the case where 20 epochs are used, the error is equal to %2.9.

Given the double difference pseudo range and double difference carrier phase measurements, the integer ambiguity estimate given by Algorithm 7.2.1 is a random variable. Therefore one can run the algorithm several times for the same data to confirm the estimate. Using this idea, in a separate experiment we ran the algorithm for each set of observations three times. If at least two out of three of the estimated integer ambiguities were the same, we would accept the repeated estimate as the integer ambiguity estimate. If none of the estimates were the same we would reject all answers. The results of this experiment are summarized in

No. of Epochs	No. of Particles	No. of trials	Error %
2	5000	1000	% 40
3	5000	1000	% 28
5	5000	1000	%15.8
10	5000	1000	%5.8
20	5000	1000	%2.9

Table 7.1: The percentage of error for integer ambiguity estimation

No. of Epochs	No. of Particles	No. of trials	Rejection %	Error %
5	5000	3×1000	%2.1	%13.7
10	5000	3×1000	%0.7	%1.2
20	5000	3×1000	%0.6	%0.0

Table 7.2: The percentage of error for integer ambiguity estimation

Table 7.2. It can be seen that when a sufficient number of measured epochs is available, the error percentage can be reduced significantly. It is also seen that for the case with a small number of epochs, the repeated trials don't help. This is because the small number of epochs makes the algorithm adapt itself to the data.

There are a few points about our method that we emphasize on:

- This method can be applied to kinematic positioning as well as static positioning.

- In this method we do not linearize the observation equations, therefore the correlation between the noise of the different measurements is only due to the double differencing.

- The use of a conditional pmf reduces the need to make complicated searches to resolve integer ambiguities, e.g. integer least squares.

Chapter 8

Detection of Abrupt Changes in a Nonlinear Stochastic System

In many practical problems arising in quality control, fault detection, and integrity monitoring, the underlying system can be represented by a parametric model. The parameters of such models usually can be categorized into two different sets. The first set contains the parameters that change slowly with respect to time, for example the parameters that describe the conditional density of position-velocity-orientation in a navigation system are of this type. The second set contains the parameters that are subject to sudden changes. These sudden changes are the results of a failure in the system dynamic, malfunctioning of measuring instruments, or perhaps the result of a change in the state of the system. We refer to these changes as sudden or abrupt because the time frame in which these changes happen is much smaller than the response time of the system which is limited by the nominal bandwidth of the system.

The abrupt changes in the system do not need to be catastrophic. In fact, in this dissertation we are interested in studying the changes that degrade the

performance/accuracy/efficiency of the system, but do not stop the system from functioning. A monitoring system is responsible for detecting and isolating these changes.

Online detection of abrupt changes for linear dynamical systems have been studied extensively (qv. [8] and the references therein). Unlike the linear case, change detection for nonlinear dynamical stochastic systems has not been investigated in any depth. In the cases where a nonlinear system experiences a sudden change, linearization and change detection methods for linear systems are the main tools for solving the change detection problem (see [41] for example). The reason for this lack of interest is clear; even when there is no change, the estimation of the state of the system given the observations results in an infinite dimensional nonlinear filter; the change in the system can only make the estimation problem harder.

As we discussed in the previous chapters, the theoretical results regarding the convergence of the approximate conditional density given by particle filtering to the true conditional density, suggests that this method is a useful approximation to exact nonlinear filtering. We believe that particle filtering and its modifications are a starting point to study change detection for nonlinear stochastic systems. Here we use the results in Chapter 4 and we develop a new change detection method for nonlinear stochastic systems. We show that for nonlinear systems the computational complexity of the CUSUM algorithm grows with respect to time, therefore, it is inapplicable in many practical applications. We introduce a change detection method based on a likelihood ratio test and a new statistic. We show that this statistic can be calculated recursively with constant computational complexity.

In Chapter 5 we showed that when the number of satellites is below a critical

number, linearization methods such as EKF result in an unacceptable position error for an integrated INS/GPS. We also showed that the approximate nonlinear filtering methods, projection particle filter in particular, are capable of providing an acceptable estimate of the position in the same situation.

In an integrated INS/GPS, if the carrier phase of the GPS signal is used for positioning, sudden changes of the phase measurement due to the cycle slip should be detected to be able to keep the integrity of positioning method intact. A cycle slip happens when the phase of the received signal estimated by the phase lock loop in the receiver has a sudden jump. If the cycle slip is not detected and repaired the position given by an integrated INS/GPS with a carrier phase receiver is no longer reliable. Therefore, one important aspect of an integrated INS/GPS is to detect such sudden changes. Since, in critical conditions, linearization methods are not capable of providing the estimate of the position, in the same setup, corresponding change detection methods are not useful either. We used an integrated INS/GPS under critical conditions as an application of our method. Since the proposed change detection method assumes known parameters after change, this application should not be considered a cycle slip detection method.

In this chapter, first we briefly define the change detection problem and we review the CUSUM algorithm for linear systems with additive changes. Then we present a new change detection method for nonlinear stochastic systems. Finally, we present some simulation results and we summarize the results and future work.

8.1 Change Detection: Problem Definition

On-line detection of a change can be formulated as follows [8]. Let $\mathcal{Y}_1^n = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$ be a sequence of observed random variables with conditional density

$p_\theta(\mathbf{y}_k|\mathbf{y}_{k-1},\cdots,\mathbf{y}_1)$. Before the unknown change time, t_0, the parameter of the conditional density, θ, is constant and equal to θ_0. After the change, this parameter is equal to θ_1. In online change detection, one is interested in detecting the occurrence of such a change. The exact time and the estimation of the parameters before and after the change is not required. In case of multiple changes, we assume that the changes are detected fast enough so that in each time instance only one change has to be considered. Online change detection is performed by a stopping rule [8]

$$t_a = \inf\{n : g_n(\mathcal{Y}_1^n) \geq \lambda\}$$

where λ is a threshold, $(g_n)_{n\geq 1}$ is a family of functions, and t_a is the alarm time, i.e. the time when change is detected.

If $t_a < t_0$ then a false alarm has occurred. The criteria for choosing the parameter λ and the family of functions $(g_n)_{n\geq 1}$ is to minimize the detection delay for the fixed mean time between false alarms.

8.2 Additive Changes in Linear Dynamical Systems

Consider the following system:

$$\begin{array}{rcccccl} \mathbf{x}_{k+1} & = & F_k\mathbf{x}_k & + & G_k\mathbf{w}_k & + & \Gamma_k\Upsilon_{\mathbf{x}}(k,t_0) \\ \mathbf{y}_k & = & H_k\mathbf{x}_k & + & \mathbf{v}_k & + & \Xi_k\Upsilon_{\mathbf{y}}(k,t_0)\ , \end{array} \tag{8.1}$$

where F_k, G_k, H_K, Γ_k, and Ξ_k are matrices of proper dimension, and $\Upsilon_{\mathbf{x}}(k,t_0)$ and $\Upsilon_{\mathbf{y}}(k,t_0)$ are the dynamic profiles of the assumed changes, of dimension $\widetilde{n} \leq n$ and $\widetilde{d} \leq d$, respectively. $\mathbf{w}_k$ and $\mathbf{v}_k$ are white Gaussian noise, independent of the

initial condition $\mathbf{x}_0$. It is assumed that$\Upsilon_{\mathbf{x}}(k,t_0)=0$ and $\Upsilon_{\mathbf{y}}(k,t_0)=0$ for $k<t_0$, but we do not necessarily have the exact knowledge of the dynamic profile and the gain matrices, Γ_k and Ξ_k. The dynamic profile of change may be assumed known or unknown.

For the case of known parameters before and after change, the CUSUM [8] algorithm can be used, and it is well known that the change detection method has the following form

$$\begin{aligned} t_a &= \min\{k\geq 1 | g_k \geq \lambda\} \\ g_k &= \max_{1\leq j\leq k} S_j^k \\ S_j^k &= \ln \frac{\prod_{i=j}^k p_{\rho(i,j)}(\epsilon_i)}{\prod_{i=j}^k p_0(\epsilon_i)}\ , \end{aligned} \tag{8.2}$$

where ϵ_i is the innovation process calculated using Kalman filtering, assuming that no change occurred, and $\rho(i,j)$ is the mean of the innovation process at time j conditioned on the change occurred at time i. p_0 and $p_{\rho(\cdot,\cdot)}$ are Gaussian densities with means 0, and $\rho(\cdot,\cdot)$, respectively. The covariance matrix for these two densities is the same and is calculated using Kalman filtering.

When the parameter after change is not known, the algorithm that is used for the change detection is the GLR test [55]. In this case, g_k is calculated as follows

$$g_k = \max_{1\leq j\leq k} \sup_{\Upsilon_{\mathbf{x}},\Upsilon_{\mathbf{y}}} S_j^k. \tag{8.3}$$

The solution for (8.3) is well known and can be found in many references [8].

Similar to nonlinear filtering, change detection for nonlinear stochastic systems results in an algorithm that is infinite dimensional. Linearization techniques, whenever applicable, are the main approximation tool for studying the change detection problem for nonlinear systems. In this setup, a nonlinear filtering problem is transformed to it linearized form through EKF and then the same algorithms

that are used for the linear Gaussian case are used for the change detection problem. Although linearization techniques are computationally efficient, they are not always applicable. In the sections to come, we propose a new method based on nonlinear particle filtering that can be used for change detection for nonlinear stochastic systems.

8.3 Nonlinear Change Detection: Problem Setup

Consider the following nonlinear system

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}_k^{i_k}(\mathbf{x}_k) + G_k^{i_k}(\mathbf{x}_k)\mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{h}_k^{i_k}(\mathbf{x}_k) + \mathbf{v}_k \ , \end{aligned} \tag{8.4}$$

where $\mathbf{x}_k \in \mathcal{R}^n$, $\mathbf{y}_{n\tau} \in \mathcal{R}^d$, $\mathbf{w}_k \in \mathcal{R}^q$ and $\mathbf{v}_k \in \mathcal{R}^d$ are white noise processes with known statistics, and the functions $\mathbf{f}_k^{i_k}(\cdot)$ and $\mathbf{h}_k^{i_k}(\cdot)$ and the matrix $G_k^{i_k}(\cdot)$ have the proper dimensions. The noise processes $\mathbf{w}_k$, $\mathbf{v}_k$, $k = 0, 1, \cdots$, and the initial condition $\mathbf{x}_0$ are assumed independent. We assume that

$$i_k = \begin{cases} 0 & k < t_0 \\ i & k \geq t_0, \quad i \in I \end{cases} , \tag{8.5}$$

where I is a countable index set. The index 0 is used for the nominal system and the system after change belongs to a countable set of systems. Here, we assume that the set I has only one member, i.e. we assume that the parameters after the change are known.

In this setup S_j^k can be written as follows

$$S_j^k = \ln \frac{p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 = j)}{p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 > k)}. \tag{8.6}$$

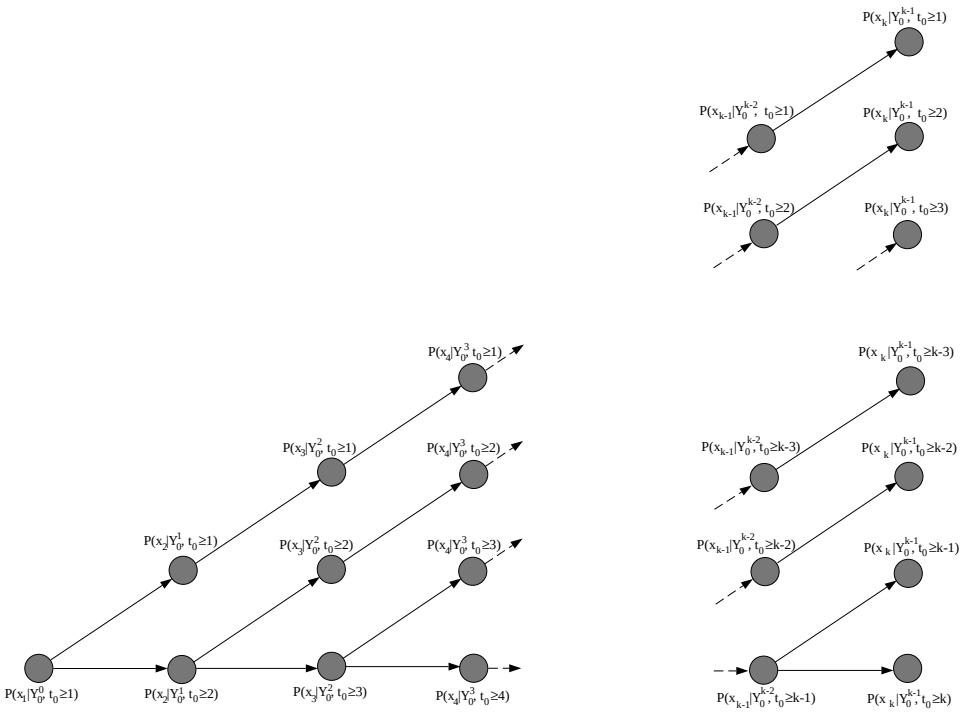

Figure 8.1: Combination of nonlinear filters used in the CUSUM change detection algorithm.

Writing (8.6) in a recursive form we get

$$p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 = j) \;\; = \;\; \prod_{i=j}^{k} p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 = j) \;, \tag{8.7}$$

where $p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 = j)$ can be written as follows

$$p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 = j) = \int_{\mathbf{x}_i} p(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 = j) d\mathbf{x}_i. \tag{8.8}$$

To find $p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 = j)$ in (8.8), one needs to find an approximation for the corresponding nonlinear filter. We assume that this approximation is done using either particle filtering or projection particle filtering (see Chapters 3 and 4).

To calculate the likelihood ratio in (8.6), we must calculate the conditional densities of the state given the observation for two hypothesis (changed occurred

at j and change occurred after k). This means that two nonlinear filters should be implemented just to compare these two hypothesis. Therefore, it is clear that to use an algorithm similar to (8.2), k parallel nonlinear filters should be implemented. In Figure 8.1, we see that the computational complexity of the CUSUM algorithm grows linearly with respect to time. In most applications this growth is not desirable. One possible way to approximate the CUSUM algorithm is to truncate the branches that are forked from the main branch in Figure 8.1. We will explain this truncation procedure and its technical difficulties in the next few lines.

Recall that the main branch (horizontal) and the branches forked from it in Figure 8.1 represent a series of nonlinear filters with specific assumptions on the change time. The dynamics and the observation equation for all forked branches are the same and the only difference is the initial density. If the conditional density of the state, given the observation, for a nonlinear system with the wrong initial density converges (in some meaningful way) to the true conditional density (initialized by the true initial density), we say that the corresponding nonlinear filter is *asymptotically stable* [14].

For asymptotically stable nonlinear filters, the forked branches in Figure 8.1 converge to a single branch, therefore there is no need to implement several parallel nonlinear filters. In other words, after each branching the independent nonlinear filter is used for a period of time and then this branch converges to the branches that have forked earlier, i.e. joins them. The time needed for the branch of the independent nonlinear filter to join the other forked branches depends on the convergence rate and the target accuracy of the approximation.

Although the procedure mentioned above can be used for asymptotically stable

nonlinear filters, there are several problems associated to this method. The known theoretical results for identifying asymptotically stable filters is limited to either requiring ergodicity and the compactness of the state space [5, 6, 35] or very special cases of the observation equation [14]. The rate of convergence of the filters in different branches is another potential shortcoming of the mentioned procedure. If the convergence rate is low in comparison with the rate of parameter change in the system, then the algorithm cannot take advantage of this convergence.

8.4 Nonlinear Change Detection: Non Growing Computational Complexity

In this section we introduce a new statistic to overcome the problem of growing computational complexity for the change detection method. We show that this statistic can be calculated recursively.

Consider the following statistic

$$T_j^k = \ln \frac{p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 \in \{j, \cdots, k\})}{p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 > k)}. \tag{8.9}$$

For the rest of this chapter we assume that, conditioned on change, the change time, t_0, is distributed uniformly, i.e.

$$P(t_0 = i | t_0 \in \{j, \cdots, k\}) = \begin{cases} \frac{1}{k-j+1} & i \in \{j, \cdots, k\} \\ 0 & \text{otherwise} \end{cases}. \tag{8.10}$$

With this assumption we have

$$\begin{aligned} p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\}) &= p(\mathcal{Y}_j^k, t_0 \in \{j,\cdots,k\}|\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\}) \\ &= p(\mathcal{Y}_j^k, t_0 = j \quad |\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\})+ \\ &\quad p(\mathcal{Y}_j^k, t_0 = j+1|\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\})+ \\ &\quad \vdots \\ &\quad p(\mathcal{Y}_j^k, t_0 = k \quad |\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\}) \\ &= \tfrac{1}{k-j+1}\Big(p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 = j)+ \\ &\qquad\qquad p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 = j+1)+ \\ &\qquad\qquad \cdots + p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 = k)\Big), \end{aligned}$$

therefore,

$$\begin{aligned} T_j^k &= \ln \tfrac{p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 \in \{j,\cdots,k\})}{p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 > k)} \\ &= \ln\left(\tfrac{1}{k-j+1}\sum_{i=j}^{k} \tfrac{p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 = i)}{p(\mathcal{Y}_j^k|\mathcal{Y}_1^{j-1}, t_0 > k)}\right). \end{aligned}$$

In other words, T_j^k can be written as follows

$$T_j^k = \ln\left(\frac{1}{k-j+1}\sum_{i=j}^{k}\exp(S_i^k)\right). \tag{8.11}$$

The change detection algorithm based on statistic T_j^k can be presented as follows

$$t_a = \min\{k \geq j \mid T_j^k \geq \lambda \quad \text{or} \quad T_j^k \leq -\alpha\}, \tag{8.12}$$

where j is the last time that $g_k \geq \lambda$ or $g_k \leq -\alpha$, and $\lambda > 0$ and $\alpha > 0$ are chosen such that the detection delay is minimum for a fixed mean time between two false alarms. Using (8.2) and (8.11), we try to find a relation between the detection method (8.12) and the CUSUM algorithm. Assume two possible extreme cases. The first one is the case where $S_i^k = c, \forall i \in \{j,\cdots,k\}$. In this case it is

clear that $T_j^k = S_i^k \quad \forall i \in \{j, \cdots, k\}$, and therefore, the performance of the two methods with the same thresholds is the same. In the second case we assume that $\exists i, l \in \{j, \cdots, k\}$ such that $S_i^k \gg S_l^k, \quad l \neq i$. Therefore, it can be seen that $T_j^k \approx S_l^k - \ln(k - j + 1)$, i.e. T_j^k is degraded by $-\ln(k - j + 1)$. With this simple analysis we can conclude that

$$\max_{i \in \{j, \cdots, k\}} S_i^k - \ln(k - j + 1) \leq T_j^k \leq \max_{i \in \{j, \cdots, k\}} S_i^k. \tag{8.13}$$

Therefore, with the same thresholds for both detection methods, (8.13) can be used to find the bounds for the performance of the detection algorithm in (8.12) with respect to the CUSUM algorithm. We want to emphasize that the thresholds used for detection method (8.12) need not be the same as the thresholds in the CUSUM algorithm, in fact, they should be optimum according to the criteria for the mean detection delay for the detection method in (8.12).

The main advantage of using the statistic T_j^k over S_j^k is the fact that T_j^k can be calculated recursively without growth in the computational complexity of the method with respect to time. We can rewrite $p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 \in \{j, \cdots, k\})$ as follows

$$p(\mathcal{Y}_j^k | \mathcal{Y}_1^{j-1}, t_0 \in \{j, \cdots, k\}) = \prod_{i=j}^{k} p(\mathbf{y}_i | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, k\}).$$

Using (8.10) we have

$$\begin{aligned} p(\mathbf{y}_i | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, k\}) &= p(\mathbf{y}_i, t_0 \in \{j, \cdots, k\} | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, k\}) \\ &= p(\mathbf{y}_i, t_0 \in \{j, \cdots, i\} | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, k\}) \; + \\ &\quad p(\mathbf{y}_i, t_0 > i | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, k\}) \qquad (8.14) \\ &= \tfrac{i-j+1}{k-j+1} p(\mathbf{y}_i | \mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, i\}) \; + \\ &\quad \tfrac{k-i}{k-j+1} p(\mathbf{y}_i | \mathcal{Y}_1^{i-1}, t_0 > i) \; . \end{aligned}$$

From (8.14) it is clear that we need only calculate two types of functions. These two functions are $p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, i\})$ and $p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 > i)$. To calculate these two functions we can use the following

$$\begin{aligned} & p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, i\}) \\ & \qquad = \int p(\mathbf{y}_i|\mathbf{x}_i,\ t_0 \in \{j, \cdots, i\}) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, i\}) d\mathbf{x}_i \\ & \qquad = \tfrac{1}{i-j+1} \int p_1(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 = i) d\mathbf{x}_i + \\ & \qquad\quad \tfrac{i-j}{i-j+1} \int p_1(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 \in \{j, \cdots, i-1\}) d\mathbf{x}_i \\ & \qquad = \tfrac{1}{i-j+1} \int p_1(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 > i-1) d\mathbf{x}_i + \\ & \qquad\quad \tfrac{i-j}{i-j+1} \int p_1(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 \in \{j, \cdots, i-1\}) d\mathbf{x}_i \end{aligned} \tag{8.15}$$

and

$$\begin{aligned} p(\mathbf{y}_i|\mathcal{Y}_1^{i-1}, t_0 > i) &= \int p(\mathbf{y}_i|\mathbf{x}_i,\ t_0 > i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 > i) d\mathbf{x}_i \\ &= \int p_0(\mathbf{y}_i|\mathbf{x}_i) p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 > i-1) d\mathbf{x}_i\ , \end{aligned} \tag{8.16}$$

where $p_0(\mathbf{y}_i|\mathbf{x}_i)$ and $p_1(\mathbf{y}_i|\mathbf{x}_i)$ are the conditional densities of the observation given the state of the system before and after the change, respectively. To calculate these two functions, two conditional densities, $p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 > i-1)$ and $p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 \in \{j, \cdots, i-1\})$ should be found. These two conditional densities can be calculated recursively as follows

$$\begin{aligned} & p(\mathbf{x}_i|\mathcal{Y}_1^{i-1}, t_0 > i-1) \\ & \qquad = \int p(\mathbf{x}_i|\mathbf{x}_{i-1},\ t_0 > i-1) p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 > i-1) d\mathbf{x}_{i-1} \\ & \qquad = \int p_0(\mathbf{x}_i|\mathbf{x}_{i-1}) p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 > i-1) d\mathbf{x}_{i-1}\ , \end{aligned} \tag{8.17}$$

where $p_0(\mathbf{x}_i|\mathbf{x}_{i-1})$ is the conditional density of the state at time i given the state at time $i-1$ assuming that no change has happened up until time $i-1$. The

recursion is complete with

$$\begin{aligned} p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 > i-1) &= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-1)p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0>i-1)}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-1)p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0>i-1)d\mathbf{x}_{i-1}} \\ &= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-2)p_0(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-2)p_0(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})d\mathbf{x}_{i-1}}, \end{aligned} \tag{8.18}$$

and it is assumed that the initial density of the state is known. (8.17) and (8.18) are, in fact, the equations for the nonlinear filter assuming that no change has happened. For the other conditional density we have

$$\begin{aligned} &p(\mathbf{x}_i|\mathcal{Y}_1^{i-1},\ t_0 \in \{j,\cdots,i-1\}) \\ &= \int p(\mathbf{x}_i|\mathbf{x}_{i-1},\ t_0 \in \{j,\cdots,i-1\})p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 \in \{j,\cdots,i-1\})d\mathbf{x}_{i-1} \\ &= \tfrac{1}{j-i}\int p_1(\mathbf{x}_i|\mathbf{x}_{i-1})p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 = i-1)d\mathbf{x}_{i-1} + \\ &\quad \tfrac{j-i-1}{j-i}\int p_1(\mathbf{x}_i|\mathbf{x}_{i-1})p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 \in \{j,\cdots,i-2\})d\mathbf{x}_{i-1}, \end{aligned} \tag{8.19}$$

where $p_1(\mathbf{x}_i|\mathbf{x}_{i-1})$ is the conditional density of the state at time i given the state at time $i-1$ assuming that a change has occurred. To complete the recursion formula we have

$$\begin{aligned} p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 = i-1) &= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0=i-1)p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0=i-1)}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0=i-1)p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0=i-1)d\mathbf{x}_{i-1}} \\ &= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-2)p_1(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0>i-2)p_1(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})d\mathbf{x}_{i-1}}, \end{aligned} \tag{8.20}$$

and

$$p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-1},\ t_0 \in \{j,\cdots,i-2\})$$

$$= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0\in\{j,\cdots,i-2\})p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0\in\{j,\cdots,i-2\})}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0\in\{j,\cdots,i-2\})p(\mathbf{y}_{i-1}|\mathbf{x}_{i-1},\ t_0\in\{j,\cdots,i-2\})d\mathbf{x}_{i-1}} \tag{8.21}$$

$$= \frac{p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0\in\{j,\cdots,i-2\})p_1(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})}{\int p(\mathbf{x}_{i-1}|\mathcal{Y}_1^{i-2},\ t_0\in\{j,\cdots,i-2\})p_1(\mathbf{y}_{i-1}|\mathbf{x}_{i-1})d\mathbf{x}_{i-1}}.$$

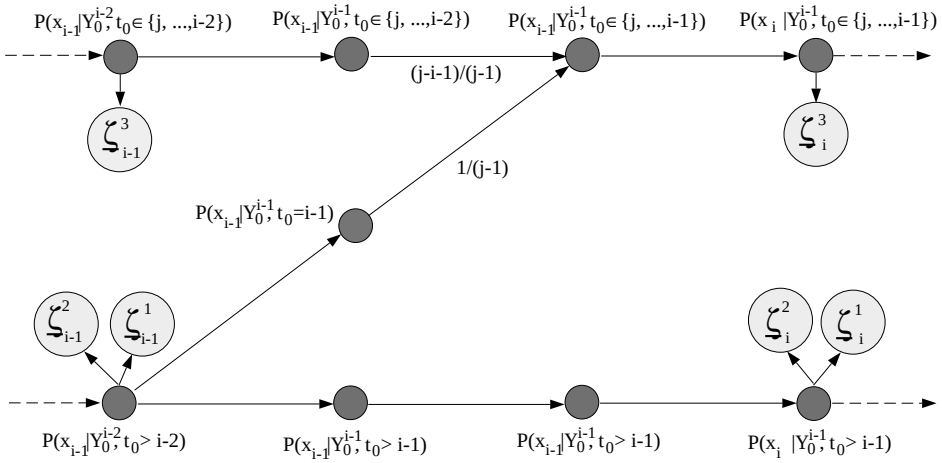

Figure 8.2: Implementation of the nonlinear filters used in the change detection algorithm in (8.12).

Figure 8.2 shows the implementation of equations (8.17) through (8.21); it can be seen that the complexity of the implemented nonlinear filter does not grow with time. Using Figure 8.2 and the definition of T_j^k we have

$$T_j^k = \sum_{i=j}^{k} \ln\left(\frac{1}{k-j+1}\left((k-i)+\frac{\varsigma_i^2}{\varsigma_i^1}+(i-j)\frac{\varsigma_i^3}{\varsigma_i^1}\right)\right), \tag{8.22}$$

where

$$\varsigma_i^1 = p(\mathbf{y}_i|\mathcal{Y}_1^{i-1},\ t_0 > i)$$

$$\varsigma_i^2 = p(\mathbf{y}_i|\mathcal{Y}_1^{i-1},\ t_0 = i)$$

$$\varsigma_i^3 = p(\mathbf{y}_i|\mathcal{Y}_1^{i-1},\ t_0 \in \{j,\cdots,i-1)\}.$$

8.5 Simulations and Results

In Chapter 5 we showed that for an integrated INS/GPS when the number of satellites is less than a critical number, projection particle filtering provides a very accurate estimate of the position while the position solution given by EKF is unacceptable. In this section we use the same example to apply the change detection method in (8.12). Similarly to Chapter 5, for a critical situation (low number of observable satellites) the linearization methods do not work, particularly we cannot use EKF. On the other hand, the CUSUM algorithm leads to a growth in computational complexity with respect to time, therefore, at this point a natural selection for a change detection algorithm is the method in (8.12). We wish to emphasize that in the example given in this section we assume that the parameter of change, before and after change, is known and the only unknown parameter is the change time. In future, we will address the more general problem of unknown change parameters.

The dynamics of an integrated INS/GPS and the observation equation for differential GPS are given in Chapter 5. The only difference is that we assume that the signal associated to one of the satellites experiences an abrupt change, i.e. we assume a known cycle slip in one of the channels.

For this simulation we simply chose an 11 dimensional Gaussian density for the projection particle filtering. This choice of density makes the random vector generation easy and computationally affordable. To be able to use the projection particle filtering, we used maximum likelihood to estimate the parameters of the Guassian density before and after Bayes' correction.

Using the data described in Section 5.2.3, we generated the pseudo range and carrier phase data for one static and one moving receiver in the same way as was

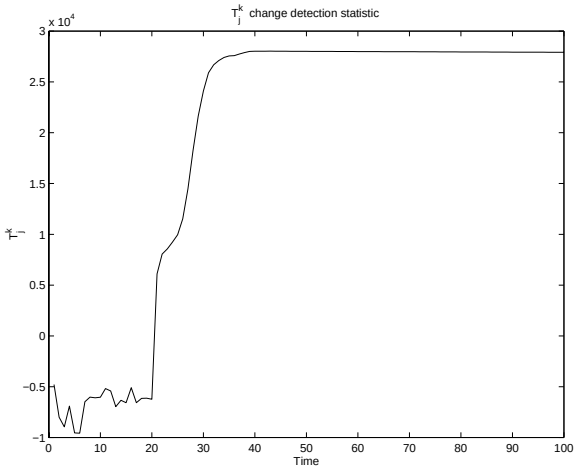

Figure 8.3: This figure shows the plot of T_j^k with respect to time. At time $t = 15$, the receiver loses 3 satellites. We assume that the cycle slip in channel one occurred at time $t = 20$.

performed in Section 5.2.3. Here we assume that for the carrier phase measurement the integer ambiguity problem is already solved. We also assumed that the phase lock loop associated to satellite 1 experiences a cycle slip and the phase suddenly changes. The size of the change is assumed to be one cycle. The movement of the INS/GPS platform was simulation based and the measurement data measured by the accelerometers, the gyros, the GPS pseudo range, and the GPS carrier phase data were generated according to that movement.

In the simulation we assumed that the GPS receiver starts with 6 satellites. At time $t = 15$, the receiver loses 3 satellites. We assume that the cycle slip in channel one occurred at time $t = 20$. In Figure 8.3 we have plotted T_j^k with respect to time. In the figure this sudden change is indicated by a sudden change in the value of T_j^K.

Chapter 9

Conclusions and Future Work

In this thesis we studied filtering, estimation, and detection for stochastic systems with nonlinear dynamics and nonlinear observations.

We presented a new approximate nonlinear filtering method for a class of systems whose conditional density lies in a certain family of exponential densities. We showed that under the conditions stated in Chapter 4 the approximate conditional density can be made arbitrarily close to the true conditional density in the sense described in the same Chapter. We also proved that when the true conditional density does not lie in an exponential family but it is close to it, the error of the estimate given by projection particle filtering is bounded. Using the results in Chapter 4, we presented a similar method for a family of mixture densities.

We showed that for an integrated INS/GPS when the number of visible satellites is below a critical number the extended Kalman filter fails to provide an acceptable estimate of the position. We showed that under the same conditions nonlinear filtering methods are capable of providing an accurate estimate of the position. Via numerical results, we also showed that the performance of the projection particle filter exceeds the regular particle filter.

We applied a method similar but different from projection particle filtering to the integer ambiguity resolution for carrier phase differential GPS. In this method, we assumed that the integer uncertainty in the carrier phase measurement is an integer random vector. We presented an algorithm that approximates the conditional density of the integer uncertainty given the observation. The numerical results reported in Chapter 7 indicate a very low percentage of error for this method.

Another problem that we addressed in this dissertation is the problem of detection of abrupt changes with known parameters after change for nonlinear stochastic systems. We showed that applying the CUSUM algorithm for such systems results in a growth in computational complexity with respect to time. To avoid this problem, we introduced a new statistic that can be used for change detection methods based on a likelihood ratio test. We showed that the calculation of this statistic can be done recursively with fixed computational complexity with respect to time.

The work we presented in this thesis may be extended towards several directions. In the following, we try to give a brief description of these directions.

Numerical results in Chapter 7 suggest that Algorithm 7.2.1 is a good candidate for an integer ambiguity resolution method. Although in that chapter we presented an argument to justify this fact, we were not able to prove that Algorithm 7.2.1 indeed provides an approximate solution for the conditional density of the integer uncertainty given the carrier phase observations. One direction for the extension of results of this dissertation would be the investigation of this matter.

From a practical point of view, we think that a very natural extension of this research would be to implement the projection particle filter for an integrated INS/GPS in real time. For this purpose one can use an affordable inertial navigation system and a hand held GPS receiver that can be connected to a computer.

We believe the next generations of cellular phones will have an inexpensive built-in integrated INS/GPS. One of the challenges in the next few years would be to design such systems with acceptable reliability and accuracy.

The abrupt change detection method that we studied in Chapter 8 is limited to change detection where the parameters after change are assumed to be known. In future, we intend to extend our results to the case where the parameters after change are unknown. The major obstacle in this extension is the complexity of the change detection method. Another subject that requires further investigations is the comparison of the presented method with other existing methods.

BIBLIOGRAPHY

[1] E. Abbott and D. Powell. Land-Vehicle Navigation Using GPS. *Proceedings of the IEEE*, 87(1):145–162, Jan. 1999.

[2] A.F.M. Smith and A.E. Gelfand. Bayesian Statistics Without Tears: A Sampling-Resampling Perspective. *The American Statistician*, 46:84–88, 1992.

[3] Y.M. Al-Haifi, S.J. Corbett, and P.A. Cross. Performance Evaluation of GPS Single-Epoch On-the-Fly Ambiguity Resolution. *Navigation*, 44(4):479–487, Winter 1998.

[4] S. Amari. *Differential-Geometrical Methods in Statistics*. Springer-Verlag, 1985.

[5] R. Atar. Exponential Stability for Nonlinear Filtering of Diffusion Processes in a Noncompact Domain. *The Annals of Probability*, 26(4):1552–1574, 1998.

[6] R. Atar and O. Zeitouni. Exponential Stability for Nonlinear Filtering. *Annales de l'Institut H. Poincare, Probabilities et Statistiques*, 33(6):697–725, 1997.

[7] P. Axelrad, C.J. Comp, and P.E. Macdoran. SNR-Based Multipath Error Correction for GPS Differential Phase. *IEEE Trans. on Aerospace and Electronic Systems*, 32(2):650–660, April 1996.

[8] M. Basseville and I.V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA, 1993.

[9] R. Beard, J. Gunther, J. Lawton, and W. Stirling. Nonlinear Projection Filter Based on Galerkin Approximation. *Journal of Guidance Control, and Dynamics*, 22(2):258–266, March–April 1999.

[10] P. Billingsley. *Probability and Measure*. John Wiley, New York, third edition, 1995.

[11] D. Brigo. *Filtering by Projection on the Manifold of Exponential Densities*. PhD thesis, Department of Economics and Econometrics, Vrije Universiteit, Amsterdam, October 1996.

[12] D. Brigo, B. Hanzon, and F. LeGland. A Differential Geometric Approach to Nonlinear Filtering : the Projection Filter. *Proceedings of the 34th IEEE Conference on Decision and Control*, pages 4006–4011, December 1995.

[13] D. Brigo and F. LeGland. A Finite Dimensional Filter with Exponential Conditional Density. *Proceedings of the 36th IEEE Conference on Decision and Control*, pages 1643–1644, December 10–12 1997.

[14] A. Budhiraja and D. Ocone. Exponential Stability in Discrete Time Filtering for Non-Ergodic Signals. *Stochastic Processes and their Applications*, 82:245–257, 1999.

[15] J. Carpenter, P. Clifford, and P. Fearnhead. An Improved Particle Filter for Non-linear Problems. *IEE Proceedings-Radar, Sonar and Navigation*, 146(1):2–7, 1999.

[16] H. Carvalho, P. Del Moral, A. Monin, and G. Salut. Optimal Nonlinear Filtering in GPS/INS Integration. *IEEE Trans. on Aerospace and Electronic Systems*, 33(3):835–850, July 1997.

[17] B.R. Crain. Exponential Models, Maximum Likelihood Estimation, and the Haar Condition. *Journal of the American Statistical Association*, 71(355):737–740, 1976.

[18] M.H.A. Davis and S.I. Marcus. An Introduction to Nonlinear Filtering. *In M. Hazewinkel and J.C. Willems edit. Stochastic Systems: The Mathematics of Filtering and Identification and Applications*, pages 53–75, 1981.

[19] N. El-Sheimy and K.P. Schwarz. Navigating Urban Areas by VISAT-A Mobile Mapping System Integrating GPS/INS/Digital Camera for GIS Applications. *Navigation*, 45(4):275–285, Winter 1999.

[20] G. Elkaim, M. O'Connor, T. Bell, and B. Parkinson. System Identification of a Farm Vehicle Using Carrier-Phase Differential GPS. *ION Conference*, pages 485–494, Sept. 1996.

[21] J.A. Farrell and M. Barth. *The Global Positioning System and Inertial Navigation*. Mc Graw Hill, New York, 1998.

[22] P. Fearnhead. *Sequential Monte Carlo Methods in Filter Theory*. Ph.D thesis, Merton College, University of Oxford, 1998.

[23] A.E. Gelfand and A.F.M. Smith. Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990.

[24] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel Approach to Nonlinear/NonGaussian Bayesian State Estimation. *IEE Proceedings-F (Radar and Signal Processing)*, 140(2):107–113, 1993.

[25] R.Z. Hasminskii. *Stochastic Stability of Differential Equations*. SIJTHOFF & NOORDHOFF Alphen aan den, The Netherlands, Rockville, Maryland, USA, 1980.

[26] A. Hassibi and S. Boyd. Integer Parametric Estimation in Linear Models with Applications to GPS. *IEEE Trans. on Signal Processing*, 46(11):2938–52, Nov. 1998.

[27] R. Hatch. Instantaneous Ambiguity Resolution. *IAG Symposium No. 107, Kinematic Systems in Geodesy Surveying, and Remote Sensing*, pages 299–308, September10-13 1990.

[28] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag, Second edition, 1993.

[29] M. Isard and A. Blake. Contour Tracking by Stochastic Propagation of Conditional Density. *Proceedings of European Conference on Computer Vision*, pages 343–356, 1996.

[30] X.X. Jin. Algorithm for Carrier-Adjusted DGPS Positioning and Some Numerical Results. *Journal of Geodecy*, pages 411–423, 1997.

[31] H.K. Khalil. *Nonlinear Systems*. Printice-Hall, second edition, 1996.

[32] G. Kitigawa. Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Models. *Journal of Computational and Graphical Statistics*, 5:1–25, 1996.

[33] J.A. Klobuchar. Ionospheric Effects on GPS. In B.W. Parkinson and J.J. Spilker Jr., editors, *Global Positioning System: Theory and Applications Volume I*, volume 163, pages 485–515. American Institute of Aeronautics and Astronautics, Inc., Washington D.C., 1996.

[34] F. LeGland, C. Musso, and N. Oudjane. An Analysis of Regularized Interacting Particle Methods in Nonlinear Filtering,. In J. Rojicek, M. Valeckova, M. Karry, and K. Warwick, editors, *Proceedings of the 3rd IEEE European Workshop on Computer-Intensive Methods in Control and Signal Processing*, volume 1, pages 167–174. Prague, September 7–9 1998.

[35] F. LeGland and N. Oudjane. Stability and Approximation of Nonlinear Filters in the Hilbert Metric, and Applications to Particle Filters. *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 1585–1590, December 2000.

[36] E.L. Lehmann and G. Casella. *Theory of Point Estimation*. Springer-Verlag, New York, second edition, 1998.

[37] G. Lorden. Procedures for Reacting to a Change in Distribution. *Annals of Mathematical Statistics*, 42:1897–1908, 1971.

[38] G.L. Mader. Kinematic GPS Phase Initialization Using the Ambiguity Function. *Proceedings of Sixth International Geodetic Symposium on Satellite Positioning*, pages 712–719, March 17-20 1990.

[39] G.N. Milstein. *Numerical Integration of Stochastic Differential Equations*. Kluwer Academic Publishers, New York, 1995.

[40] P. Del Moral. Non Linear Filtering: Interacting Particle Solution. *Markov Processes and Related Fields*, 2(4):555–580, 1996.

[41] I.V. Nikiforov. New Optimal Approach to Global Positioning System/Differential Global Positioning System Integrity Monitoring. *Journal of Guidance, Control and Dynamics*, 19(5):1023–1033, Sept–Oct 1996.

[42] M. O'Connor, G. Elkaim, T. Bell, and B. Parkinson. Automatic Steering of Farm Vehicles Using GPS. *3rd International Conference on Precision Agriculture*, pages 767–778, June 1996.

[43] E.S. Page. Continuous Inspection Schemes. *Biometrika*, 41:100–115, 1954.

[44] B.W. Parkinson. GPS Error Analysis. In B.W. Parkinson and J.J. Spilker Jr., editors, *Global Positioning System: Theory and Apllications Volume I*, volume 163, pages 469–483. American Institute of Aeronautics and Astronautics, Inc., Washington D.C., 1996.

[45] B.W. Parkinson and P.K. Enge. Differential GPS,. In B.W. Parkinson and J.J. Spilker Jr., editors, *Global Positioning System: Theory and Apllications Volume II*, volume 164, pages 3–50. American Institute of Aeronautics and Astronautics, Inc., Washington D.C., 1996.

[46] P.M. Maybeck. *Stochastic Models, Estimation, and Control*, volume 2. Academic Press, 1982.

[47] B.W. Remondi. Pseudo Kinematic GPS Results Using the Ambiguity Function Method. *Navigation, Journal of the Institute of Navigation*, 38(1):17–36, 1991.

[48] D.B. Rubin. Using the SIR Algorithm to Simulate Posterior Distributions. In J.M. Bernardo, M.H. DeGroot, D.V. Lindley, and A.F.M. Smith, editor, *Bayesian Statistics*, pages 395–402. Oxford University Press, Oxford, 1988.

[49] K. Sobczyk. *Stochastic Differential Equation With Application to Physics and Engineering*. Kluwer Academic Publishers, 1991.

[50] J.J. Spilker. GPS Signal Structure and Performance Characteristics. *Navigation, Journal of the Institute of Navigation*, 25(2):121–146, 1978.

[51] P.J.G. Teunissen. A New Method for Fast Carrier Phase Ambiguity Estimation. *Proc. IEEE Position, Location and Navigation Symp.*, pages 562–73, 1994.

[52] P.J.G. Teunissen. The Least Square Ambiguity Decorrelation Adjustment: a Method for Fast GPS Integer Ambiguity Estimation. *Journal of Geodesy*, 70(1-2):65–83, Nov. 1995.

[53] P.J.G. Teunissen and A. Kleusberg. GPS Observation Equations and Positioning Concepts. In A. Kleusberg and P.J.G. Teunissen, editors, *GPS for Geodesy*, pages 175–217. Springer-Verlag, New York, 1996.

[54] R.V.V. Vidal. *Applied Simulated Annealing*. Springer-Verlag, 1994.

[55] A.S. Willsky and H.L. Jones. A Generalized Likelihood Ratio Approach to the Detection and Estimation of Jumps in Linear Systems. *IEEE Transactions on Automatic Control*, 21(2):108–112, February 1976.

[56] R. Zickel and N. Nehemia. GPS Aided Dead Reckoning Navigation. *Proc. ION Nat. Tech. Meeting*, pages 577–586, 1994.