PARAMETRIC NON-LINEAR FILTERING

by

Richard H. Lamb, Jr.

B.S.E.E., George Washington University (1978)

S.M., Massachusetts Institute of Technology (1981)

E.E., Massachusetts Institute of Technology (1981)

SUBMITTED TO THE DEPARTMENT OF
ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1987

Signature of Author ______________________________

Department of Electrical Engineering and Computer Science

May 26, 1987

Certified by ______________

Robert R. Tenney

Thesis Supervisor

Accepted by ______________________________

Arthur C. Smith

Chairman, Department Committee on Graduate Students

Parametric Non-linear Filtering

by

Richard H. Lamb, Jr.

Submitted to the Department of Electrical Engineering
and Computer Science on June 20, 1987 in partial
fulfillment of the requirements for the degree of
Doctor of Science

Abstract

In this work we present a new approach to designing a class of non-linear filters which store accumulated information on the state of a system with a parametric probability density function (pdf). We ensure that the density retains a selected parametric structure through each non-linear filtering operation by approximating the result of each operation with a parametric pdf of the same form. We solve the approximation problem after each filtering operation by projecting the pdf after the operation back onto the original class of parametric pdfs. This re-projection technique takes on a particularly appealing form requiring only functions of the moments of the pdf when we use exponential polynomial pdfs. In this way the non-linear filtering problem is reduced to one which operates only on the parameters of the pdf. As it turns out, the well known linear (Kalman) and pseudo-linear (Extended Kalman) filters are degenerate forms of this approach, in which the pdf is an exponential with a second order polynomial exponent (e.g. Gaussian). In addition many of the advanced non-linear filtering techniques can be considered as special cases of our general approach. Since the computations of our approach involve only the parameters for ANY function, the inherent efficiency of the Kalman filtering style approach is retained as we go to more complicated densities required by more non-linear problems. The results of this work show that our technique does include, as a subset, many of the existing filtering methods. In fact, as a byproduct of this work, we find that we have a technique to not only approximate pdf's but any function which is described by a PDE. The resultant approximation shall always be in the form of a parameterized function associated with an ODE describing the parameters.

Thesis Supervisor: Robert R. Tenney
Title: Assistant Professor of Electrical Engineering

Dedicated to the memory of Howard Paul Hayden.

....oh it seems the good they die young,

I just looked around and he's gone.

-Dion

Acknowledgements

Over the past six years a great number of people have contributed to my personal and professional development. In particular I would like to thank Robert R. Tenney for providing me with the continual incentive, financial support and unlimited freedom during my stay at the Laboratory for Decision and Information Sciences. Such a combination I will not find anywhere. His continual encouragement in the face of difficult problems is why this work has finally reached an end.

I am also greatful to the other members of my doctoral committee Professors Alan Wilsky and Sanjoy Mitter for their comments and help during this work.

Of the many people who have made my stay at MIT enjoyable, I would like to thank the late Howard Hayden for showing me how to have a good time, Krishna Nathan for being my partner in crime for the various liberties taken during my graduate career, Peter Garnavich for his healthy dose of pessimism, and Vivek Dhavan for his philosophy of life and education. Special thanks go to Tae Joo and Mark Colavita (my 'third' thesis reader) for always coming through in a pinch, technically and otherwise. Their friendship has made life here enjoyable. The many interesting discussions with Ahmed Tewfik, Robin Rohlicek, Jerry Prince, and Ken Chou have also been very useful as well as entertaining. Thanks also go to Dr. Peter Doerschuk who taught me the meaning of masochism.

My mother deserves more than what can be said here, without her lifetime of support and encouragement none of this would have been possible. Kathie Wilson also deserves a special thanks for her warmth and persistence in the final phase of this thesis even though it meant peculiar hours on my part.

Finally I would like to thank Tom Ballard of The Naval Surface Weapons Center for providing the financial support of this work in its early stages and Sanjoy Mitter for providing support in my last term.

(This page unintentionally left blank)

Table of Contents

Chapter 1

Introduction

1.1 Scope of Thesis

Applications

Estimating the state of a dynamical system using noisy measurement data is one of the most common applications of stochastic estimation theory. Examples exist in all engineering fields. In tracking a moving object (system) with radar returns (measurements), it is nessesary to estimate the location and velocity (states). In orbit determination, observations consist of range, range rate, and directional cosines from which we must estimate the orbit or state. In underwater applications we may passively sample data from hydrophone arrays to determine the location and characteristics of a submarine.

In each case, we must start with a model for the state dynamics and the measurement process. This characterizes mathematically the internal and external behavior of the system. The model may not describe the system exactly and the observations may be noisy, distorted and incomplete. Given whatever information we have, the goal is to try to estimate the internal state of the system in the best fashion possible. This procedure of processing measurements to get an estimate of the state is referred to as 'filtering' and has been an active area of research for many years. In what follows, we shall review that small part of this research which is most relevant to the technical approach taken herein.

When the model and measurements are linear functions of the state we wish to estimate, closed form and efficient algorithms exist for estimating the state. Most notable among these is the Kalman Filter (KF). This solution to the filtering problem consists of a recursive pair of vector/matrix equations describing the state estimate and its error variance. Not only does this algorithm yield a solution to the (linear) filtering problem, but it also provides a structure that is very appealing from the point of view of implementation. This latter issue is why many filtering problems are cast into the framework of the Kalman Filter, even though the final results may be sub-optimal.

In the general filtering problem both model and measurements may be nonlinear functions of the state. For this case few algorithms exist for estimating the state, let alone efficient ones. Due to the generality of such models, solutions to the nonlinear filtering problem are typically presented in the form of probability density functions describing the state. This approach yields a probabalistic description of the state in its most general form. Although solutions 'exist' [2],[1], they cannot, in general, be implemented. This is due to the complex nature of the equation describing the pdf, which involves the evaluation of many multidimensional integrals over the whole state space.

Motivation

Since the majority of real world models involve nonlinearities of some sort, we seek a solution to the filtering problem which results in a probabilistic description of the state, for generality, which from an implementation standpoint can be calculated from a finite number of recursive (vector) equations. In this work we define an implementation as an algorithm or set of equations which operate on a finite set of real numbers. This definition is driven by the desire to take advantage of the existence of high speed digital computers which can only handle finite sets of real numbers (ignoring truncation). A trivial example of an application of this definition is the common approximation of functions by a finite grid of points. In this work, we shall seek a more sophisticated implementation which shall allow the designer to choose the form of the approximation and then optimize its evolution over time (the finite grid approximation is just a special case of our general scheme).

The initial motivation for this work came from the desire to estimate processes which are multi-modal in nature. Since most algorithms presently available are only capable of modeling single modal (Gaussian) processes, while many physical proceses are inherently multi-modal and non-Gaussian, we found it essential to derive an algorithm which could deal with such processes. As a result of our efforts, we found a procedure which turns out to be useful in a large class of real world filtering problems.

Directions of Past Research

Past research in the area of nonlinear filtering has concerned itself primarily with approximating the original state model in such a way that it may be considered linear about some operating point. One of the key results of this work is the Extended Kalman Filter (EKF) [4](p 190). This degenerate form of the nonlinear filter is basically a compromise in order to make the filter computationally feasible. Unfortunately, as would be expected, a great many applications cannot be cast into this framework, as they give rise to a filter which is highly unstable.

Many other approaches to the nonlinear filtering problem have simply been extensions of this theme. The EKF is the result of approximating the various components of the state model with their two-term Taylor series expansions. Other filters which have been developed are the second order and higher order filters. These simply consider more terms in the Taylor series approximations to the state model components.

The two general results of past research have been those by Kushner [7] and Zakai [6]. Kushner's work resulted in a partial differential equation (PDE) describing the evolution of the true underlying probability density function (pdf) describing the state, based on the state model, while the work by Zakai simplified this result. Since the pdf is a complete statistical description of the state, these results can be used to determine the best estimate of the state over a large class of optimality criteria. The key problem with this approach is the difficulty in solving the PDE's describing the pdf. Only a few closed form solutions to special cases exist (with the KF being one of them), and since the PDE is infinite dimensional, any other solution involves an approximation. In addition, due to the dimension of the state, most approximation techniques become inadequate. Hence, while an equation describing a solution to the nonlinear filtering problem exists, it can not be practically implemented.

Directions of Current Research

Current research in the area of nonlinear filtering consists of searching for closed form solutions for specific models [3], or the existence thereof [5]. The key to much of the activity is due to the simplifications afforded by the Zakai equation. The work has contributed a great deal to the understanding of the difficulties involved with the implementation of the nonlinear filter. However, thus far, very few practical filter implementations have arisen from these studies.

Other recent work has involved the use of a combination of linear filtering results to approximate the probabalistic description afforded by the general nonlinear filters [8], [9], [10], [11]. Derivations of these filters have used a 'bottom-up' approach (unlike the above work), starting with the linear (or pseudo linear) filtering solutions and forming weighted sums of their corresponding pdf's to account for the unrestricted, probably multi-modal, shape of the pdf's associated with the actual nonlinear filter. Since the individual linear filters are easily constructed, a reasonable combination of such filters still yields a result which can be implemented.

Another approach which has led to a practical way of approximating pdf's is the work by Lo [12]. By restricting the model to linear dynamics and nonlinear measurements, he uses an exponential Fourier series to approximate the underlying pdf for periodic processes. The key property of this form of approximation is that the form of the exponential Fourier series can be maintained during pdf evolution by simply modifying the Fourier coefficents. This scheme will be considered in more detail later.

Direction of this Thesis

The direction of this thesis shall be to study the filtering problem from the viewpoint of pdf's describing the state, as in the general nonlinear filtering solution, but parameterized by a finite set of parameters, as in the EKF and weighted sum approaches. In this way we may consider both linear and nonlinear filtering solutions, from both past and current research, under one framework. This will in turn allow us to develop algorithms which derive characteristics from both methodologies, leading to general, yet efficient, filtering techniques for nonlinear models. In the following section, we will outline some of the contributions made by this thesis.

1.2 Contributions of this Thesis

Theory

To sumarize, the work of this thesis has led to two main contributions:

- A technique for approximating solutions to a partial differential equation with the solution to a set of related ordinary differential equations.
- A systematic approach to approximating the pdf associated with the optimal nonlinear filter with a finitely parameterized function (resulting from the application of the above).

The first contribution is a technique for approximating partial differential equations (PDE) of the form

$$\frac{\partial y(x,t)}{\partial t} = f(x,t,y(x,t)) \quad y(x,0) = y_0(x) \tag{1.2.1}$$

with a set of ordinary differential equations (ODE) of the form

$$F(\underline{\alpha}(t))\frac{\partial \underline{\alpha}(t)}{\partial t} = \underline{g}(\underline{\alpha}(t)) \quad \underline{\alpha}(0) = \underline{\alpha}_0 \tag{1.2.2}$$

where $\hat{y}(x;\underline{\alpha}(t))$ is the function approximating the solution to Equation 1.2.1, e.g., $y(x,t) \approx \hat{y}(x;\underline{\alpha}(t))$, and $\underline{\alpha}(t)$ is the vector of parameters which describes the functional approximation. The parametric function $\hat{y}(x;\underline{\alpha}(t))$ is chosen by the filter designer based on any a priori information about the true function's $(y(x,t))$ behavior (derived from the state model), and/or on implementation issues. Although the precise form is not critical to the theory behind our approach, a parameterization which cannot capture the key characteristics of $y(x,t)$ will lead to a poor approximation.

After the form of $\hat{y}(x;\underline{\alpha}(t))$ has been determined, our technique determines the 'optimal' F and $\underline{g}$ terms from this parameterization and the original PDE. As it turns out, the F matrix is strictly a function of the parameterization, and corresponds to an orthogonalization step onto a set of basis vectors $\frac{\partial \hat{g}(x;\underline{\alpha}(t))}{\partial \alpha_i}$. The vector $\underline{g}$, on the other hand, depends on the parameterization and on $f(,,)$, and corresponds to the projection of the function described by Equation 1.2.1, $f(x,t,y)$, onto the set of basis vectors $\frac{\partial \hat{g}(x;\underline{\alpha}(t))}{\partial \alpha_i}$.

Although this result actually came about as a byproduct of our work toward finding an efficient nonlinear filtering algorithm, it has applications in many areas outside of filtering theory.

Applications

The second contribution of this work comes from the application of our approximation technique to nonlinear filtering problems. Here the partial differential equation we wish to approximate is the Fokker-Planck (F-P) equation describing the evolution of the conditional density function, $p(x,t)$, for a specific stochastic model:

$$\frac{\partial p(x,t)}{\partial t} = \mathcal{L}^*(p(x,t)) \quad p(x,0) = p_0(x) \tag{1.2.3}$$

where $\mathcal{L}^*(\cdot)$ is the F-P operator[1]. The result of our approximation scheme is

$$p(x,t) \approx \hat{p}(x;\underline{\alpha}(t)) \tag{1.2.4}$$

$$F(\underline{\alpha}(t))\frac{\partial \underline{\alpha}(t)}{\partial t} = \underline{g}(\underline{\alpha}(t)) \quad \underline{\alpha}(0) = \underline{\alpha}_0. \tag{1.2.5}$$

Using the properties associated with pdf's, the forms of F and g can be simplified, and can usually be interpreted as functions of moments. This eliminates the need for any complicated integral evaluations, numerical or otherwise. Hence, the result is a filter which can be applied to an arbitrary state model (at least in theory) and which retains the finite implementation aspects of the Kalman filter.

In order to demonstrate our approximation technique and to describe the form of F and g, as well as to illustrate the practical aspects of implementing such a filter, we shall study the nonlinear filter by way of example. We will apply Equation 1.2.5 to various parameterization/model combinations, and then analyze the results. Specifically, we shall consider the parameterization-model combinations listed in Table 1.1. In what follows we describe the filtering background which lead

[1]Since the difference between the F-P and Zakai equations in this context is slight, we state only the former.

Approximating pdf	Parameterization	Dynamical model	Chapter
Gaussian	$e^{-\frac{1}{2P}(x-m)^2+c}$	Linear scalar and vector	4
Gaussian	$e^{\alpha_0+\alpha_1 x+\alpha_2 x^2}$	Linear scalar and vector	4
Gaussian	$e^{-\frac{1}{2P}(x-m)^2+c}$	4th order polynomial scalar	5
Gaussian	$e^{\alpha_0+\alpha_1 x+\alpha_2 x^2}$	4th order polynomial scalar	5
Gaussian	$e^{-\frac{1}{2}(x-m)^T P^{-1}(x-m)+c}$	2nd order vector	5
Weighted sum of Gaussians	$\sum c_i N(x; m_i, P_i)$	4th order polynomial scalar	6
Weighted sum of Gaussians	$\sum c_i N(\underline{x}; \underline{m}_i, P_i)$	2nd order vector	6
Quartic	$e^{\alpha_0+\alpha_1 x+\alpha_2 x^2+\alpha_3 x^3+\alpha_4 x^4}$	4th order polynomial scalar	7

Table 1.1: Parametric pdf's and models considered

us to pursue the work in this thesis.

1.3 Engineering Background

Fundamentals

In this section we shall lay down a framework to study nonlinear filtering problems, and then outline some of the work done by the engineering community in filtering. In doing so we also provide the definitions necessary for later sections and chapters. We catagorize the material in this chapter as 'engineering' since it tends toward near term practical solutions to the filtering problem. In the next section we shall discuss work done by the 'mathematical' community.

We start by describing the state model that will be of concern in this thesis:

$$\begin{aligned} dx &= \mathbf{f}(x,t)dt + dw && \text{Dynamical Model} \\ dz &= \mathbf{h}(x,t)dt + dv && \text{Measurement Model} \end{aligned} \qquad (1.3.1)$$

where dw and dv are independent Brownian processes with intensities $Q(t)$ and $R(t)$ respectively. $x \epsilon R^n$ is the state of the system which we would like to estimate. $\mathbf{f}(x,t)$ is the model for the dynamics of the state and $\mathbf{h}(x,t)$ is the model for the relationship between measurement and state. When either of the functions $\mathbf{f}(x,t)$ or $\mathbf{h}(x,t)$ are other than linear functions of x, the problem is said to be nonlinear.

Using this framework, the nonlinear filtering problem can be stated as follows:

- Given the functions $\mathbf{f}(x,t)$ and $\mathbf{h}(x,t)$
- and a series of measurements, z, up to time t (presumably generated from a system described by the above model)
- then determine the 'best' estimate of the state x at time t

Diffusion Operator

The solution to the filtering problem associated with Equation 1.3.1 is given by many authors [2], [7], [1]. Given here is a solution from [1](Zakai equation):

$$\frac{\partial p}{\partial t} = \mathcal{L}^*(p) + h^T(x,t)R^{-1}(t)pz - \frac{1}{2}\mathbf{h}^T(x,t)R^{-1}(t)\mathbf{h}(x,t)p \tag{1.3.2}$$

$$\mathcal{L}^*(p) = -\sum_{i=1}^{n}\frac{\partial(f_i(x,t)p)}{\partial x_i} + Q(t)\sum_{i,j=1}^{n}\frac{\partial^2 p}{\partial x_i \partial x_j} \tag{1.3.3}$$

where p corresponds to the unnormalized conditional density function of the state x, given all the measurements, z, up to the present. From this equation we may derive any of the statistics of the state estimate (e.g. conditional mean). A solution of these equations requires storage of the complete function $p(x,t)$ for all x. This, unfortunately, usually makes a direct solution impractical. It is the goal of many researchers to find a suitable framework in order to implement the nonlinear filter. The following pargraphs will discuss some of the notable methods presently available.

Linear

Much work has been done on the application of existing methods to the solution of the nonlinear filtering problem, resulting in modifications to the Kalman filter to handle nonlinearities. Among them are the Extended Kalman Filter [4](pp190), truncated second order filter [2](pp336), and Gaussian second order filter [2](pp336). Due to its simplicity, the EKF is the most popular for handling nonlinearities.

All of these methods rely on the assumption that most of the area under the density function describing the state is near the mean, i.e., that the mean and mode of the distribution are near one another. This is because the derivation approximates the nonlinearities with low order Taylor series expansions for $\mathbf{f}(x,t)$

and $\mathbf{h}(x,t)$. This approximation severely limits the use of these methods in many of the multi-modal conditional distributions which would occur in the exact solution to many important filtering problems.

Density Approximations

A slightly more sophisticated approach, similar to the one proposed in this research, is one given by Sorenson [9]. His method approximates the state's density function with a series of orthogonal polynomials. The coefficients of these (Hermite) polynomials possess many of the properties of moments, and equations for their evolution can be obtained in a straightforward manner. However, this approximation requires that the actual density be nearly Gaussian. In addition, the resultant approximation may not truly be a pdf since many of the higher order terms of the series must be neglected for practical reasons.

A similar approach, not restricted to almost Gaussian processes, is presented by Alspach [10]. Here the approximating pdf is made up of a convex sum of individual Gaussian densities (multi-modal). Since the terms in such a sum are non-orthogonal, an assumption is made about the variance associated with each term. By making each Gaussian pdf very narrow (small variance) relative to the spacing between the means, interference between terms is negligible, allowing processing to proceed in parallel using N Kalman filters. While computationally attractive, this method is, unfortunately, not sufficiently general to model many situations with severe nonlinearities and/or large process noise. For example, if the model had a large amount of process noise, the variance of each Gaussian in the sum would eventually increase to the point where overlap of the Gaussians would occur, invalidating the non-overlapping assumption, thus not allowing separate processing.

A sum of Gaussians approach to the pdf approximation problem which is not restricted to Gaussians with small variances is described by Buxbaum [11]. Here, the approximation is again based on the results of N Kalman filters running in parallel. The state estimate is given by a weighted sum of estimates from the individual Kalman filters (KF) where the weights correspond to likelihood functions describing the filters. At each step of this method the number of KF's and likelihood functions increases by a factor of m, where m equals the number of terms in the Gaussian sum approximation. Therefore, although this method is more general than the work of [10], storage and computational load grow as m^k (where k is the step number). This forces the deletion of KF-likelihood pairs using various criteria. Numerical results indicate that even when many terms are deleted at each step, this method yields estimates with a smaller mean square errors than provided by a single EKF. However, none of the deletion criteria can guarantee that a term dropped at any one stage would not have led to terms which would influence the estimate significantly at a later stage. Numerical results have also shown this to be the case. This makes the degradation due to these deletions/approximations unpredictable.

Using a different approach, Bar-Shalom [8] arrives at a similar weighted Gaussian sum approximation technique using Bayes' rule and specializing to the context of multi-target tracking problems. In this case, the weighting is cleverly chosen to correspond to the probability of an event, which ensures the Gaussianess of the series terms. In the multi-target tracking problem where each target is assumed to have linear dynamics and measurements, multiple Kalman filters can be used to update the Gaussian terms, while the weighting probabilities must be determined in some a priori manner. This application is similar to that which motivates the work of this thesis. However, this scheme does not consider nonlinearities in either dynamical or measurement models and is therefore not sufficently general for anything but this specific application.

An approach to the implementation problem which considers aspects from both the engineering and mathematical communities is the work by Lo [12]. In this work, he considers processes which are restricted to the circle and hence have a probability density which is periodic with period 2π in the state variable. He then chooses the approximating density to be an exponential Fourier series. This choice of pdf can be made to approximate the true density arbitrarily closely by taking more terms in the Fourier series exponent. The key advantage to this form is that it can be designed such that the form is preserved over a discrete measurement update, i.e., a multiplication step. To guarantee this closure, the number of terms in the exponent of the exponential Fourier series must be chosen large enough to well approximate the measurement pdf in Bayes' rule[2] , i.e., $p(y/x)$. Then, each update step can be implemented by an addition to the coefficents of the Fourier series. Hence the pdf associated with a model consisting of linear dynamics and nonlinear measurements can be accurately modeled by repeatedly applying Bayes' rule to this exponential Fourier series pdf approximation.

Unfortunately this method, in addition to being restricted to processes on the circle, can only handle very limited dynamics. This is due to the fact that the exponential Fourier series form is not closed under the convolution operation, which is needed to predict the pdf forward in time. Therefore, although this scheme can handle a large class of filtering problems (e.g. phase estimation), it is not sufficiently general to handle a wide range of dynamical models. The approach we shall seek will not restrict our approximating pdf to a specific form or to particular dynamics and, in fact, could be applied to the above work to alleviate the problems caused by the predict step (convolution).

Summary

In summary we find that the engineering approach toward a nonlinear filtering solutions is dictated by the implementation aspects of the solution. In each case, the problem is broken up into a number of simpler problems to which solutions can be implemented. This scheme clearly has advantages when dealing with a problem whose form naturally breaks up into many sub-filters, such as multi-target tracking. But this natural breakup is difficult to conceive when we consider the general non-linear filtering problem. One overall approach, [12], makes the practical compromise between dynamics and measurements by considering models (or tranformations of models) which place all nonlinearities in the measurements. With this approach, an algorithm can be obtained which retains all the implementation aspects of the EKF. In the next section we shall outline what has been done regarding the generalized

[2] For discrete measurements and dynamics, the nonlinear filter equations describing the pdf are given by

$$p(x_k) \leftarrow (1/c)p(z_k/x_k) \int p(x_k/x_{k-1})p(x_{k-1})dx_{k-1}$$

which involves a multiplication for the measurements $(p(z_k/x_k))$ and a convolution for the dynamics $(p(x_k/x_{k-1}))$.

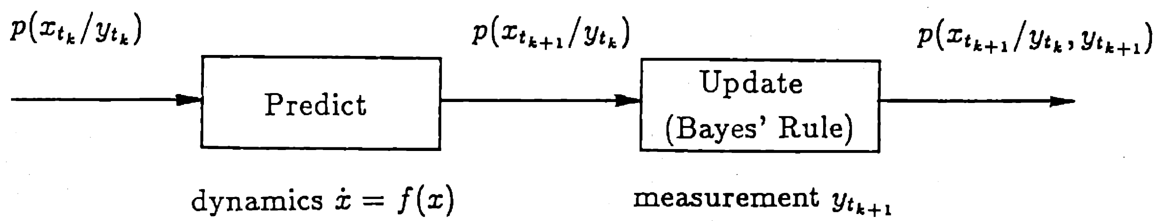


Figure 1.4.1: Separation between measurements(update) and dynamics(predict)

filtering problem.

1.4 Mathematical Background

Where the difficulty lies: Nonlinear dynamics

In the previous section we found that many of the engineering techniques restrict the pdf describing the state to the form of a weighted sum of Gaussian pdf's. This constraint limits their application to a small handful of physical models. In this section we will consider the work done on the unconstrained pdf associated with the general nonlinear filtering solution.

Before we proceed, we suggest that due to the fact that most practical applications involve discrete measurements, therefore removing the burden of including the measurment operation in the pdf evolution equations, the majority of the work in nonlinear filtering theory has been concentrated on solving or simplifying the nonlinear diffusion operator. This separation is displayed in Figure 1.4.1. The exception to this is the development of a simplified version of the diffusion operator with continuous measurements, given the originator's name – the Zakai equation [6]. Its recent popularity has caused a re-examination of the equations describing the evolution of pdf's. For this reason the following section will be devoted to describing the Zakai equation, after which the key results in this area will be described.

Zakai Equation

The diffusion operator describes the evolution of the unnormalized, conditional pdf. This pdf in turn gives a complete probabalistic description of the state of the process we would like to estimate, given all measurements up to some point. With this we may obtain the best estimate of the state of the system described by Equations 1.3.1. As it turns out, there is a natural separation in the diffusion operator between the dynamical model and the measurements. For the continuous dynamics/discrete measurement case, this separation is a practical one of applying measurements when they occur, i.e., at discrete time points via Bayes' rule. For the continuous dynamics/continuous measurement case, the measurement appears as an added term to the diffusion operator, $\mathcal{L}^*()$. It is in this added term that the Zakai and Kushner equation differ as shown below:

$$\frac{\partial p'}{\partial t} = \mathcal{L}^*(p') + \mathbf{h}^T(x,t)R^{-1}(t)p'z - \frac{1}{2}\mathbf{h}^T(x,t)R^{-1}(t)\mathbf{h}(x,t)p' \text{ (Zakai)} \quad (1.4.2)$$

$$\frac{\partial p}{\partial t} = \mathcal{L}^*(p) + (\mathbf{h}(x,t) - E\{\mathbf{h}(x,t)\})^T R^{-1}(t)(z - E\{\mathbf{h}(x,t)\}) \text{ (Kushner)}$$

$$\mathcal{L}^*(\cdot) = -\sum_{i=1}^{n} \frac{\partial(f_i(x,t)(\cdot))}{\partial x_i} + Q(t)\sum_{i,j=1}^{n} \frac{\partial^2(\cdot)}{\partial x_i \partial x_j} \quad (1.4.3)$$

$$E\{\mathbf{h}(x,t)\} = \int \mathbf{h}(x,t)p\,dx \quad (1.4.4)$$

The effect of this difference is in the way the resulting pdf of the two equations are interpreted. For Kushner's equation, p is the usual normalized density; however, for the Zakai equation, p is an unnormalized density, i.e., $\int p\,dx \neq 1$. Note also that the Zakai equation is in a simpler form due to a lack of the $E\{\mathbf{h}(x,t)\}$ term. This relatively reasonable tradeoff between normalization and simplicity is the cause for renewed interest in the general nonlinear filtering problem.

Existence Proofs

Mitter and Brockett [1], [13] present a mathematical framework using Lie algebras to study the nonlinear filtering problem. Other frameworks consist of dealing with the filtering problem in a quantum-mechanical and control theory context [1]. These frameworks have contributed greatly to the understanding of the nonlinear filtering problem. In particular, some of the results stemming from this framework are those concerning the existence and dimensionality of the exact solution [5]. These have clearly shown the difficulties associated with implementing nonlinear filters. However, the only practical implementations that have arisen from these frameworks are as yet few and specific (Kalman and Benes [3] filter).

Benes' Filter

The work by Benes' [3] is a good example of a result that has been obtained by the mathematical community. Benes' [3] has obtained an exact, yet implementable, solution to a class of models described as follows:

$$\mathbf{h}(x,t) = x \tag{1.4.5}$$

$$\frac{\partial \mathbf{f}(x,t)}{\partial x} + \mathbf{f}^2(x,t) = ax^2 + bx + c \quad \text{where } a \geq -1. \tag{1.4.6}$$

The results for this class are similar in form to the KF. They are therefore very desirable from an implementation point of view. As in the KF, there are only two parameters describing the pdf, which correspond roughly to mean and variance. However, the pdf itself consists of a time-independent 'shape-factor', which sets the pdf's initial shape, multiplied by a Gaussian described by the pseudo mean and variance parameters. This work clearly shows the merit of the tools used by researchers in the mathematical area of nonlinear filtering.

However, since our goal is toward obtaining nonlinear filter implementations, and since many researchers are already studying the general aspects of nonlinear filtering, it was deemed that our efforts would be better spent on deriving practical algorithms which approximate the general solution to the nonlinear filtering problem, rather than on obtaining exact solutions to a few specific problems.

Summary

Of all the approaches presented thus far (engineering and mathematical), the work on pdf approximation by Sorenson [9] and Lo [12] come closest to the goal of this research effort. Our goal, however, will be to determine an approach which encompasses a broader class (non-Gaussian, multi-modal, general dynamics) of nonlinear systems ocurring in practice, which can be derived easily, and be implemented practically (i.e., finite storage requirements). The following section describes our approach.

1.5 Overall Direction

Contributions of Engineering Past

In this section we describe the overall direction of the work in this thesis in relation to past efforts in both the engineering and mathematical fields of nonlinear filtering.

From the engineering point of view, we find that past contributions to nonlinear filtering have basically resulted in algorithms similar in form to the EKF. This is due primarily to its efficient implementation. The EKF, as well as second and higher order filters [4], exemplify this. The work by Bar-Shalom [8] and others [9-11], involving the combinined output of many KF's, shows the strong motivation toward preserving the characteristic efficiency of the KF/EKF. Finally, the work by Lo [12] results in a highly efficient filter implementation for the class of periodic processes. By considering nonlinearities only in the measurments, he is able to select a pdf approximation that is closed under discrete measurement updates, thus simplifying implementation. Therefore, the basic direction of the engineering community in the field of nonlinear filtering appears to be toward efficient implementation.

Contributions of Mathematical Past

From the mathematical point of view, we have seen that past contributions to nonlinear filtering have resulted in a better understanding of the true structure of the general filtering problem. This has led to the flexible representation of the state with probability density functions. Whereas in the engineering approach we consider only 2 or 3 parameters (mean,variance, and sometimes weighting), here we may consider an infinite number, namely all the moments associated with a pdf. Although equations describing the evolution of the pdf exist [2], [1], its implementation is currently impractical. Hence specific results in this area have been concerned with the existence of closed form solutions, and with different frameworks in which to study the evolution equations, in the hopes of simplifying the equations to a form that can be reasonably implemented. Therefore, the basic direction of the mathematical community appears to be toward finding an exact or logically simplified implementation of the pdf evolution equations.

Direction of this Thesis

In this thesis we shall attempt to seek a midpoint between the two camps to find an efficient, yet comprehensive, solution, by considering the contributions from both filtering areas. We consider our work as being categorized between the 'top-down' mathematical approach and the 'bottom-up' engineering approach. The fundamental motivation, however, will still be implementation. The key aspect we take from the bottom-up and top-down approaches will be the ideas of parameterization and probabilistic state descriptions, respectively. The quality that makes the EKF so efficient is the fact that it deals with only a fixed number of parameters, namely means and covariances. The quality that makes the general probabilistic formulation so appealing is the fact that it considers the complete probabilistic description of the state, namely the conditional pdf. Our approach shall be to derive a filter using the probabilistic formulations (F-P and Zakai equations), but with a finite number of parameters. That is, we will replace the true pdf in the evolution equations with a function characterized by a finite number of parameters and possessing the properties of a pdf. Although the parameters may be the mean and variance of the pdf, this is neither usual nor necessary. Specifically, in situations where our scheme shows the greatest promise, we shall consider more than 2 or 3 parameters. As would be expected, for the case of a 2 parameter pdf, our results are similar to those of the EKF. It is one of the advantages of this approach that an increase in the number of parameters leads to a graceful increase in the complexity of the equations while improving the general performance of our filter.

As a by-product of the unconstrained nature of the parameter definitions, we find that when applying discrete measurement updates via Bayes' rule, for a certain class of measurement functions, the update step becomes trivial, precisely as was seen in the work in [12]. By selecting the pdf parameterization, $\hat{p}(x;\underline{\alpha}(t))$, to be similar to the form of the update pdf, $p_{z/x}$, an update step can be effected by a simple modification of the parameters of the approximating pdf. This is just a generalization of work in [12].

We clarify the steps with the following example. If the update pdf had the form

$$p_{z/x} = exp(\sum_{i=0}^{m} \beta_i(z) x^i), \tag{1.5.1}$$

we could select the finite parameter pdf to be of the form

$$\hat{p}(x; \underline{\alpha}(t)) = exp(\sum_{i=0}^{n} \alpha_i x^i), \tag{1.5.2}$$

where $n \geq m$. This would make each update step take the form of:

$$\hat{p}(x; \underline{\alpha}(t)) \leftarrow \hat{p}(x; \underline{\alpha}(t)) \cdot p_{z/x} \tag{1.5.3}$$

$$\text{or } \alpha_i \leftarrow \alpha_i + \beta_i(z) \text{ for } i = 0...m \text{ and} \tag{1.5.4}$$

$$\alpha_i \leftarrow \alpha_i \text{ for } i = (m+1)...n. \tag{1.5.5}$$

Although this feature is not the primary direction of this work, it does further motivate our general parametric pdf approach to nonlinear filtering problems. In the following section we outline the steps in this thesis.

1.6 Outline of Thesis

In this section we shall outline the various steps in this thesis. To summarize, Chapter 2 develops the theory and derives the equations for our approximation technique ($F\dot{\underline{\alpha}} = g$), all in the context of arbitrary PDE's. Chapter 3 specializes the result to pdf's and addresses the question of what error function should we minimize to determine the approximation. The result is a pair of error criteria, both of which allow both F and g to be interpreted as moments. Referring to Table 1.1, Chapters 4,5,6,7 apply our scheme to various parameterization and model combinations beginning with the Gaussian parameterization with linear dynamics and increasing in complexity to a quartic parameterization with nonlinear dynamics. Chapter 8 determines some rough performance measures to aid the engineer in choosing a particular parameterization. Finally, Chapter 9 presents conclusions and some suggestions for further work.

Chapter 2: Parametric Filtering Methods

CHAPTER 1. INTRODUCTION

In this chapter we will present the mathematical concepts and derive the approximating relationships which are the foundation of the work in this thesis. The work described here is cast in a framework of partial differential equations; it does not depend on any specific filtering results and hence can be treated as theory to be later specialized to the filtering problem. Although the results of this chapter are of a general nature, the initial motivation was strictly the practical one of reducing the dimension of our filtering problem in some systematic way. We present the concepts in this general format for clarity only.

Section 2 describes the general problem of approximating solutions to differential equations, the basic framework, and fundamental definitions. Only simple arguments and conceptual proofs, sufficient to make the final result plausible, are given. Section 3 makes these derivations precise and formal. Section 2 is written in such a way that the mathematically uninterested reader may omit Section 3 without any loss in continuity. Section 4 assesses the resultant finite dimensional approximation with examples and discusses limitations. Section 5 considers some ancillary information produced as a by-product of the approximation process.

Chapter 3: Specialization to PDF's

In this chapter we specialize the results of Chapter 2 to the case where the partial differential equation describes the evolution of probability density functions on the state space of some stochastic process.

In Section 2 we will choose two error criteria for use with pdf's. These choices will actually involve manipulating the approximation step to simplify calculations. Section 3 will describe the structure of the F matrix under these error criteria, and Section 4 will do the same for the $\underline{g}$ vector. Since the $\underline{g}$ term also depends on the partial differential equation describing the pdf (unlike the F matrix), this section will describe $\underline{g}$ for the two pdf differential equations, namely the Fokker-Planck and Zakai equations. Section 5 will consider the problems associated with implementing the $F\dot{\underline{\alpha}} = \underline{g}$ equation. Finally, Section 6 will summarize the results of the chapter.

Chapter 4: Linear Dynamics/Gaussian Parameterization

In this chapter we will apply the results of Chapters 2 and 3 to the linear filtering problem. The motivation is twofold. First, we need some way of verifying that the approximating schemes yield sensible results. A comparison with the well known results of linear filtering provides such a verification. Second, it allows us to work through a simple example of applying our approximation scheme to a specific problem. Section 2 will derive the equations for the two error criteria chosen in Chapter 3. Section 3 will comment on their agreement with existing linear filtering results.

Chapter 5: Nonlinear Dynamics/Gaussian Parameterization

In this chapter, we apply our projection scheme to a nonlinear filtering problem using the same Gaussian parameterization as in Chapter 4. Section 2 will derive the general equations for the scalar nonlinear filter. In Section 3, we apply these results, as well as existing nonlinear filtering results, to a particular nonlinear dynamical model for comparison. In Section 4, we will describe the portion of our approximation due to the Zakai continuous measurement term, again, for a particular measurement equation to ease comparisons. In Section 5, we will outline the corresponding results for the vector case. Finally, in Section 6, we shall consider a scalar and vector example which shows some of the advantages and disadvantages of our approximation scheme.

Chapter 6: Nonlinear Dynamics/Gaussian Sum Parameterization

In this chapter, we will consider a weighted sum of second order parameterizations (Gaussians). In Section 2 we will derive the general equations for the scalar nonlinear filter and show their relationship with previous work and the results of Chapter 5. Then, in Section 3, we shall apply the results to a number of examples to verify perfomance and make comparisons. The goal is to show that the additional terms resulting from considering pairs of pdf's result in a better description of the true underlying distribution than single Gaussian parameterizations. Finally, in Section 4, we will summarize the results of the chapter.

Chapter 7: Nonlinear Dynamics/Quartic Parameterization

In this chapter we will consider a parameterization similar to the one discussed in Chapter 4 under the logarithmic error criterion, but much more general. Specifically, we consider the quartic parameterization

$$\hat{p}(x;\underline{\alpha}(t)) = e^{\alpha_0+\alpha_1 x+\alpha_2 x^2+\alpha_3 x^3+\alpha_4 x^4} \tag{1.6.1}$$

where the parameters are given by

$$\underline{\alpha}(t) = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix}. \tag{1.6.2}$$

Little is known about the relationship between $\underline{\alpha}$ and the moments of this class of distribution. This is due primarily to the fact that closed form expressions for integrals involving this function do not exist. As we shall see, the lack of such expressions forces us to apply our scheme using only one of the two error criteria considered in Chapter 3.

The motivation for selecting such a parameterization lies in simplifications afforded by its purely exponential structure, unlike the approximation of Chapter 6, and by its multi-modality, unlike the approximation of Chapter 5. Due to the increase in complexity associated with ordering for vector versions of Equation 1.6.1, and for clarity, this chapter will concern itself only with scalar processes.

Section 2 will derive the equations for our filter using the quartic pdf (Equation 1.6.1), initially leaving the dynamics of the process model unspecified. Later,in Section 3, we will choose a specific dynamical model as we did in Chapter 5 for comparison purposes. Section 4 will discuss some numerical issues associated with the resulting equations. Section 5 will consider a few examples using our specific dynamical model and discuss the results. Section 6 will apply the results of Section 2 to a particular example which includes nonlinear measurements, and will compare the results with the EKF. Finally, in Section 7 we summarize the results of this chapter.

Chapter 8: Factors in Choosing a Parameterization

In this chapter we shall address the performance issues associated with our filtering scheme. We shall do this by determining the rate at which the error between the true pdf and approximating pdf increases from a common starting point. Using this result, we then discuss the general implications of singularity of the F matrix, zeros in the g vector, and parametric transformations on filter performance. It is not the intent of this chapter to present a thorough error analysis of our filtering scheme; this we feel is beyond the scope of the thesis. It is the intent, however, to present some simple results which should prove useful in choosing good parameterizations with respect to filter perfomance.

Section 2 will derive the error rate using some of the results from Chapter 2. Section 3 will use these results to make statements about the relationship between magnitude of the $\underline{g}$ vector and parameter transformations on the error rate (and hence performance). Section 4 and 5 will evaluate the error rate for the specific parameterizations considered in this thesis. Section 6 will compare these results. Section 7 will consider the effects of a measurement update step on the true error between the underlying pdf and our approximation. Finally, Section 8 will summarize the results of this chapter.

Chapter 9: Conclusions

This chapter will outline the contributions made by this thesis and make suggestions for further work.

1.7 Cast of Characters

- $\underline{\alpha}, \underline{\alpha}(t)$ vector of parameters
- $\alpha_i, \alpha_i(t)$ ith parameter
- $\dot{\underline{\alpha}}, \dot{\underline{\alpha}}(t)$ derivative of parameters with respect to t
- $\mathbf{f}, \mathbf{f}(x)$ state dynamics
- $\mathbf{h}, \mathbf{h}(x)$ measurment function
- dw driving noise
- dv observation noise

- $y(x,t)$ function we wish to approximate
- $\hat{y}(x,t)$ approximating function
- $\hat{y}(x;\underline{\alpha}(t))$ our parametric approximation
- $p(x,t)$ actual underlying probability density function
- $\hat{p}(x,t)$ approximating probability density function
- $\hat{p}(x;\underline{\alpha}(t))$ our parametric pdf approximation
- $q(x,t)$ square root of the actual underlying pdf
- $\hat{q}(x;\underline{\alpha}(t))$ square root of our parametric pdf approximation
- $\varsigma(x,t)$ logarithm of the actual underlying pdf
- $\hat{\varsigma}(x;\underline{\alpha}(t))$ logarithm of our parametric pdf approximation
- δp infinitesimal change in the true pdf $p(x,t)$
- δy infinitesimal change in the true function $y(x,t)$
- δx infinitesimal change in x
- δt infinitesimal change in t
- $\delta\underline{\alpha}$ infinitesimal change in the vector $\underline{\alpha}$
- $\underline{0}$ zero vector
- $\underline{x}$ state vector
- $\underline{m}$ vector of means of our approximating pdf
- $\hat{p}_{\alpha_i} \quad = \frac{\partial \hat{p}}{\partial \alpha_i}$
- $\hat{\varsigma}_{\alpha_i} \quad = \frac{\partial \hat{\varsigma}}{\partial \alpha_i}$
- $\hat{q}_{\alpha_i} \quad = \frac{\partial \hat{q}}{\partial \alpha_i}$

Chapter 2

Parametric Filtering Methods

2.1 Introduction

In this chapter we will present the mathematical concepts and derive the approximating relationships which are the foundation of the work in this thesis. The work described here is cast in a framework of partial differential equations; it does not depend on any specific filtering results and hence can be treated as theory to be later specialized to the filtering problem. Although the results of this chapter are of a general nature, the initial motivation was strictly the practical one of reducing the dimension of our filtering problem in some systematic way. We present the concepts in this general format for clarity only.

Section 2 describes the general problem of approximating solutions to differential equations, the basic framework, and fundamental definitions. Only simple arguments and conceptual proofs, sufficient to make the final result plausible, are given. Section 3 makes these derivations precise and formal. Section 2 is written in such a way that the mathematically uninterested reader may omit Section 3 without any loss in continuity. Section 4 assesses the resultant finite dimensional approximation with examples, and discusses any limitations. Section 5 considers some ancillary information produced as a byproduct of the approximation process.

2.2 Theory

Framework

The basic practical problem with computing the solution to a general partial differential equation

$$\frac{\partial y(x,t)}{\partial t} = f(y(x,t),x,t) \text{ with initial condition } y(x,0) \tag{2.2.1}$$

is the infinite dimensionality of $y(x,t)$. One popular approximation method to solve this equation uses finite element techniques, where $y(x,t)$ is represented on a grid of points in x and t. (The situation is made worse when x is a vector quantity. This section considers x as a scalar for simplicity.) One way to make this approach practical is to approximate the x,t grid with a finite number of points.

Because computers store a finite set of 'real' numbers (ignoring truncation), finite element techniques reduce Equation 2.2.1 to the propagation of values of $y(x,t)$ at a finite set of points. The work in this thesis is merely a generalization of this concept, seeking a way to propagate these values in a way that is, in some specific sense, optimal. An overall view of the various approximation possibilities is shown in Figure 2.2.1.

Another common approximation replaces $y(x,t)$ with a function or sum of functions having simple or closed form solutions which are 'close' to $y(x,t)$. Although many practical issues affect the selection of these functions and the definition of 'close', such an approach has the potential for greatly simplifying the implementation of partial differential equations of the form 2.2.1. This concept is also a specific case of our general approximation scheme.

Let us first define 'close'. To do this we must specify some metric between two functions. Let $\hat{y}(x,t)$ be a function approximating $y(x,t)$. Most metrics depend on the difference $y(x,t) - \hat{y}(x,t)$, and a natural choice of metrics involves an inner product on the function space in which $y(x,t)$ resides. Depending on what aspect of $y(x,t)$ is important to a particular application, any one of a family of inner products may be chosen. Thus, we shall leave the specific form of the inner product unspecified, and proceed to derive an approximating function $\hat{y}(x,t)$.

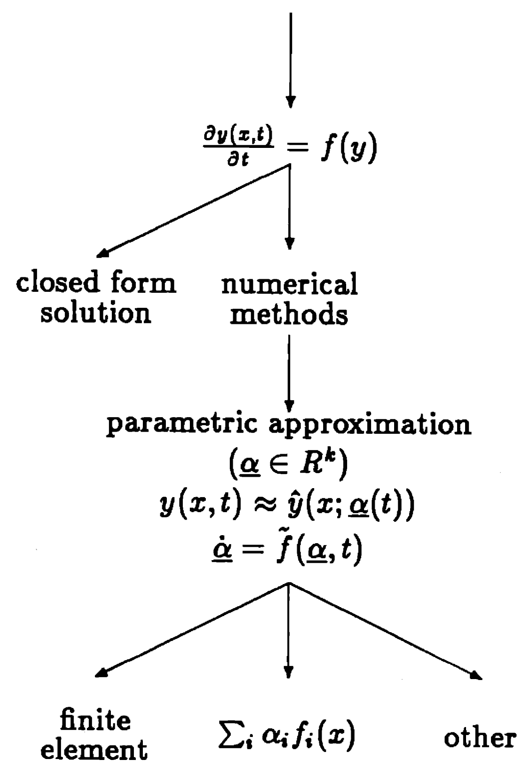


Figure 2.2.1: Family of Approximations

Since the 'best' choice of $\hat{y}(x,t)$ depends on the particular application, we elect to study the general class of parametric functions $\hat{y}(x;\underline{\alpha}(t))$, where $\underline{\alpha} \in R^k$. This function has the advantage of being finite dimensional in the sense that it is parameterized by a finite vector, whereas $y(x,t)$, in general, is not. Thus, using $\hat{y}(x;\underline{\alpha}(t))$ should lead to a finite dimensional approximation.

Since we want the dynamics of $\underline{\alpha}$ to account for the evolution over t of $y(x,t)$, we seek an inner product which involves only the x component of the solution, e.g.,

$$< f_1, f_2 >= \int f_1(x,t) f_2(x,t) dx \tag{2.2.2}$$

Now for any particular application, we need only to determine the dynamics for $\underline{\alpha}$ which minimizes the distance between $y(x,t)$ and $\hat{y}(x;\underline{\alpha}(t))$. We will assume this distance measure will be over x, i.e., distance will be a function of $\underline{\alpha}$ and t only. The goal here is to use the partial differential equation 2.2.1 as the source of an ordinary differential equation in terms of the parameters $\underline{\alpha}$, such as

$$\frac{d\underline{\alpha}}{dt} \quad = \quad G(\underline{\alpha}, f(\cdot), t) \tag{2.2.3}$$

where G is some operator relating f, from the partial differential equation 2.2.1, to $\dot{\underline{\alpha}}$. However, since we do not have access to the true solution $y(x,t)$ at every t, this new equation can only permit evaluation of $f(\cdot)$ at the point $\hat{y}(x;\underline{\alpha}(t))$. Thus, we require

$$\frac{d\underline{\alpha}}{dt} = G(\underline{\alpha}, f(\hat{y}(x;\underline{\alpha}(t))), t). \tag{2.2.4}$$

Since t and the $\underline{\alpha}$'s completely characterize the arguments to G, we can write

$$\frac{d\underline{\alpha}}{dt} \quad = \quad \tilde{f}(\underline{\alpha}, t) \tag{2.2.5}$$

where $\tilde{f}$ is a new function derived from G in Equation 2.2.3. The reduction of Equation 2.2.1 to something like Equation 2.2.5 produces an ODE which greatly simplifies issues associated with numerical solutions. In the following, we will derive a general procedure for such a reduction/approximation step.

Concept

Our approach to the approximation will make use of the projection techniques of linear algebra. We may think of $y(x,t)$ as a vector function of t indexed on x, and is thus infinite dimensional when x is continuous. The function $\hat{y}(x;\underline{\alpha}(t))$, on the other hand, is completely determined by the finite dimensional vector $\underline{\alpha}(t)$. Thus the scheme will be to project the infinite dimensional $y(x,t)$ onto the finite dimensional manifold defined by the mapping $\hat{y}(x;\underline{\alpha}(t))$, from R^k (in which $\underline{\alpha}$ resides) into the space of $y(\cdot,t)$. As $y(\cdot,t)$ evolves over time, so will the projection; we will use this to derive dynamics for $\underline{\alpha}$ which follow the projection. The resulting description of $\underline{\alpha}$, in terms of $\dot{\underline{\alpha}}$, will consist of a finite number of ordinary differential equations.

Since $\hat{y}(x;\underline{\alpha}(t))$ must be nonlinear, as a function of $\underline{\alpha}$, for all but the most trivial approximations, we must resort to the usual linearization approach which considers $\hat{y}(x;\underline{\alpha}(t))$ to be infinitesimally linear about some $\underline{\alpha}$, and then proceed with the projection steps. In what follows, we will describe the steps of this approximation.

Infinitesimal Linearity Condition

We start with some definitions. For clarity, we shall drop the dependence of $y(x,t)$ and $\hat{y}(x;\underline{\alpha}(t))$ on x and t from the notation. Hence from this point on

$$y(x,t) = y \text{ and } \hat{y}(x;\underline{\alpha}(t)) = \hat{y}(\underline{\alpha}). \tag{2.2.6}$$

Now define $\epsilon = y - \hat{y}(\underline{\alpha})$as the error between the approximate solution and true solution to Equation 2.2.1, and assume ϵ, y, and $\hat{y}(\underline{\alpha}) \in C^\infty$, and $\underline{\alpha} \in R^k$. Define $\|\epsilon\| =< \epsilon, \epsilon >$ as the distance between approximate and true solutions. Now we can cast our problem as finding the solution to

$$\underline{\alpha}_* = arg(\min_{\underline{\alpha}} \|\epsilon\|) \tag{2.2.7}$$

with the necessary (but not sufficient) condition for optimal $\underline{\alpha}$

$$\frac{\partial \|\epsilon\|}{\partial \underline{\alpha}} = \underline{0}. \tag{2.2.8}$$

For simplicity, assume initially that we are given the optimal $\underline{\alpha}$ (say $\underline{\alpha}(t)$) for some initial y (say $y(t)$). This starting point may have been given to us or computed using any of a number of techniques. Now consider the variation of $\hat{y}(\underline{\alpha})$ about

$\underline{\alpha} = \underline{\alpha}(t)$. For sufficiently small $\delta\underline{\alpha} = (\underline{\alpha}(t+\delta t) - \underline{\alpha}(t))$, we will assume that the manifold is linear about $\underline{\alpha}(t)$. Thus, we may approximate $\hat{y}(\underline{\alpha}(t+\delta t))$ as

$$\hat{y}(\underline{\alpha}(t+\delta t)) = \hat{y}(\underline{\alpha}(t)) + \tilde{y}(\delta\underline{\alpha}) \tag{2.2.9}$$

where $\tilde{y}()$ is a linear function of $\delta\underline{\alpha}$. Now Equation 2.2.8 becomes a sufficient, as well as necessary, condition for optimality so long as $\underline{\alpha}$ remains in the neighborhood of $\underline{\alpha}(t)$.

Before we may make use of Equation 2.2.8, however, we must rewrite it in terms of $\hat{y}(\underline{\alpha}(t+\delta t))$ and $\underline{\alpha}$. Expanding Equation 2.2.8 yields

$$\begin{aligned} \frac{\partial}{\partial\underline{\alpha}}\|\epsilon\| &= \frac{\partial}{\partial\underline{\alpha}} < y(t+\delta t) - \hat{y}(\underline{\alpha}(t+\delta t)), y(t+\delta t) - \hat{y}(\underline{\alpha}(t+\delta t)) > && (2.2.10) \\ &= \frac{\partial}{\partial\underline{\alpha}} < \hat{y}(\underline{\alpha}(t+\delta t)), \hat{y}(\underline{\alpha}(t+\delta t)) > -2\frac{\partial}{\partial\underline{\alpha}} < \hat{y}(\underline{\alpha}(t+\delta t)), y(t+\delta t) > \\ &= \underline{0}. && (2.2.11) \end{aligned}$$

which can be written as

$$\begin{pmatrix} \vdots \\ < \frac{\partial\hat{y}(\underline{\alpha}(t+\delta t))}{\partial\alpha_i}, (\hat{y}(\underline{\alpha}(t+\delta t)) - y(t+\delta t)) > \\ \vdots \end{pmatrix} = \underline{0} \quad (\in R^k). \tag{2.2.12}$$

Substituting Equation 2.2.9 into Equation 2.2.12 yields

$$\begin{pmatrix} \vdots \\ < \frac{\partial\tilde{y}(\delta\underline{\alpha})}{\partial\underline{\alpha}}, (\hat{y}(\underline{\alpha}(t)) + \tilde{y}(\delta\underline{\alpha}) - y(t+\delta t)) > \\ \vdots \end{pmatrix} = \underline{0} \tag{2.2.13}$$

whose solution is $\underline{\alpha}_*$. This result is valid only if $\delta\underline{\alpha}$ remains small, which implies that $y(t+\delta t)$ should be close to the initial $y(t)$, i.e.,

$$y(t+\delta t) = y(t) + \delta y \tag{2.2.14}$$

where δy is on the order of $\tilde{y}(\delta\underline{\alpha})$.

Evaluating Equation 2.2.13 in the context of this infinitesimal y approximation yields

$$\begin{pmatrix} \vdots \\ < \frac{\partial\tilde{y}(\delta\underline{\alpha})}{\partial\alpha_i}, (\tilde{y}(\delta\underline{\alpha}) - \delta y) > \\ \vdots \end{pmatrix} + \begin{pmatrix} \vdots \\ < \frac{\partial\tilde{y}(\delta\underline{\alpha})}{\partial\alpha_i}, (\hat{y}(\underline{\alpha}(t)) - y(t)) > \\ \vdots \end{pmatrix} = \underline{0}. \tag{2.2.15}$$

Here we assume $\tilde{y}(\delta\underline{\alpha})$ has the form

$$\tilde{y}(\delta\underline{\alpha}) = \delta\underline{\alpha}^T \frac{\partial \hat{y}(\underline{\alpha})}{\partial \underline{\alpha}}|_{\underline{\alpha}=\underline{\alpha}(t)} \tag{2.2.16}$$

which makes Equation 2.2.9 the two term Taylor series expansion for $\hat{y}(\underline{\alpha})$ about $\underline{\alpha}(t)$, i.e.,

$$\hat{y}(\underline{\alpha}(t+\delta t)) = \hat{y}(\underline{\alpha}(t)) + \delta\underline{\alpha}^T \frac{\partial \hat{y}(\underline{\alpha})}{\partial \underline{\alpha}}|_{\underline{\alpha}=\underline{\alpha}(t)}. \tag{2.2.17}$$

We shall assume that $\hat{y}(\underline{\alpha})$ is continuously differentiable in α. For $\hat{y}(\underline{\alpha})$'s which do not satisfy this assumption, our procedure may not yield consistent results. In fact, for y's which encourage a step change in $\underline{\alpha}$ (in $\hat{y}(\underline{\alpha})$), y will not be approximated well by this scheme due to the continuity constraint above.

Substituting 2.2.17 into Equation 2.2.15 yields

$$\begin{pmatrix} \vdots \\ < \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)}, \left(\left(\frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)} \right)^T \delta\underline{\alpha} - \delta y \right) > \\ \vdots \end{pmatrix} + \begin{pmatrix} \vdots \\ < \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)}, (\hat{y}(\underline{\alpha}(t)) - y(t)) > \\ \vdots \end{pmatrix} = \underline{0}. \tag{2.2.18}$$

Now since $\hat{y}(\underline{\alpha}(t)) - y(t)$ was assumed to have been already minimized against $\frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)}$ at the start, the second term is zero for all i. Hence, for a change δy in y, the corresponding optimal change in $\underline{\alpha}$ is the $\delta\underline{\alpha}$ given by

$$\delta\underline{\alpha}_* = solution\left(\begin{pmatrix} \vdots \\ < \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)}, \left(\frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}|_{\underline{\alpha}=\underline{\alpha}(t)} \right)^T \delta\underline{\alpha} - \delta y \right) > \\ \vdots \end{pmatrix} = \underline{0} \right) \tag{2.2.19}$$

and

$$\underline{\alpha}_* \leftarrow \underline{\alpha}(t) + \delta\underline{\alpha}_*. \tag{2.2.20}$$

We see that by setting $\underline{\alpha}(t+\delta t) = \underline{\alpha}_*$ and applying the above equations again, the next $\underline{\alpha}_*$ can be found. Thus, Equations 2.2.19, 2.2.20 describe a recursive approach to obtaining $\underline{\alpha}_*$ in steps of $\delta\underline{\alpha}$, in lieu of the direct solution of Equation 2.2.7. So long as we can obtain the infinitesimal progression of y and δy *exactly*, we may derive the same for $\underline{\alpha}_*$. Figure 2.2.2 describes this progression pictorially.

A possible algorithm implementing these steps is as follows:

1. Initialize y and find $\underline{\alpha}_*$.

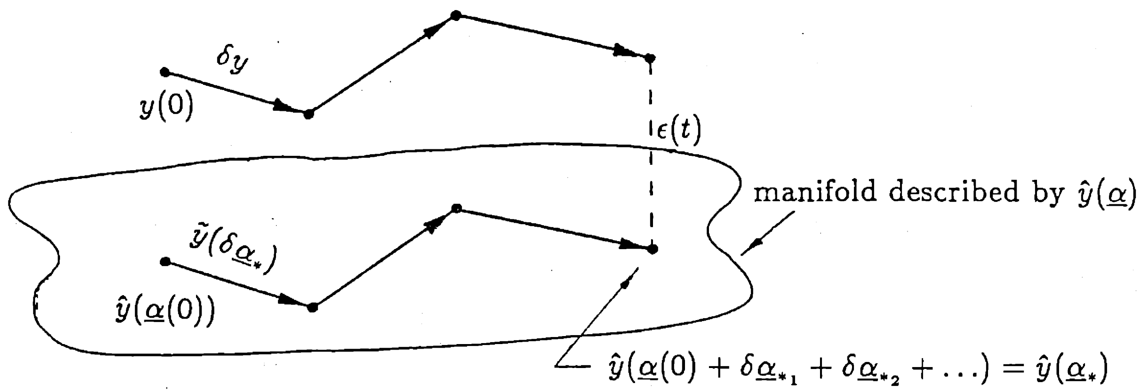


Figure 2.2.2: Progression of y and $\hat{y}(\underline{\alpha})$.

2. Set $y(t) \leftarrow y$, $\underline{\alpha}(t) \leftarrow \underline{\alpha}_*$
3. Get δy (we will discuss this next)
4. Solve for $\delta\underline{\alpha}_*$ using Equations 2.2.19, 2.2.20
5. Set $\underline{\alpha}_* \leftarrow \underline{\alpha}(t) + \delta\underline{\alpha}_*$
6. Goto 2.

Finally, we may include the differential equation which describes y,

$$\frac{\partial y}{\partial t} = f(y), \tag{2.2.21}$$

into our infinitesimal projection theorem, and let δt pass to the limit of 0. Combined with Equation 2.2.19, this will result in a differential equation for $\underline{\alpha}$.

In order to relate Equation 2.2.19 to Equation 2.2.21, we must find δy with respect to changes in t. We can do this by just dividing Equation 2.2.19 through by δt and replacing $\frac{\delta\underline{\alpha}}{\delta t}$ with $\dot{\underline{\alpha}}$ and $\frac{\delta y}{\delta t}$ with y_t, as $\delta t \to 0$. This yields our final equation

$$\begin{pmatrix} \ddots & \cdots & \\ \vdots & < \frac{\partial\hat{y}(\underline{\alpha})}{\partial\alpha_i}, \frac{\partial\hat{y}(\underline{\alpha})}{\partial\alpha_j} > & \\ \cdots & & \end{pmatrix} \dot{\underline{\alpha}} = \begin{pmatrix} \vdots \\ < \frac{\partial\hat{y}(\underline{\alpha})}{\partial\alpha_i}, f(y) > \\ \vdots \end{pmatrix}. \tag{2.2.22}$$

This completes the conversion from the infinite dimensional y equation to the finite approximation for $\underline{\alpha}$. Next, we will eliminate the dependence of $\underline{\alpha}$'s on y.

CHAPTER 2. PARAMETRIC FILTERING METHODS

y Elimination

While at this point it may appear that the approximation is finished, further inspection of Equation 2.2.22 reveals that it still depends on y. Since our goal was to completely avoid any derivations involving the exact solution y, we must determine how to eliminate the term $f(y)$, perhaps by introducing $\hat{y}(\underline{\alpha})$ terms. If, in Equation 2.2.21, $f(y)$ were only a function of x and t, i.e., $f(y,x,t) = f(x,t)$, then Equation 2.2.22 can be used as is. However, this case is not sufficiently general to handle the situations that will be described later. In order to apply Equation 2.2.22 to the more general cases, we will approximate $f(y)$ with $f(\hat{y}(\underline{\alpha}))$. We will justify this with the following argument.

Assume at $t = 0$ that not only is $\hat{y}(\underline{\alpha})$ the best approximation to y (as was assumed in Equation 2.2.22), but that also $y = \hat{y}(\underline{\alpha})$, so that initially Equation 2.2.22 is valid. Now due to the unconstrained/infinite dimensional nature of $f()$, $y \neq \hat{y}(\underline{\alpha})$ at some time later, and hence $f(y) \neq f(\hat{y}(\underline{\alpha}))$, so that at time δt we must invoke the approximation $y \approx \hat{y}(\underline{\alpha})$. However $\hat{y}(\underline{\alpha}(t+\delta t))$ is the best approximation to y at this point, so for lack of a better substitute, we shall accept whatever error this brings. The result of the next step will be again a further approximation of y, since it will be based on the previous approximation. Therefore, we see that the errors will tend to accumulate as we evolve from the initial point where $y = \hat{y}(\underline{\alpha})$. Note, however, that this is still the best we can do while keeping Equation 2.2.22 a function of $\underline{\alpha}$ only. Section 5 of this chapter explores these errors more fully. Figure 2.2.3 shows this pictorially (compare with Figure 2.2.2).

Rewriting Equation 2.2.22 with this assumption, we have

$$\underbrace{\begin{pmatrix} \ddots & \cdots & \\ \vdots & < \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}, \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_j} > & \\ \cdots & & \end{pmatrix}}_{F} \dot{\underline{\alpha}} = \underbrace{\begin{pmatrix} \vdots \\ < \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i}, f(\hat{y}(\underline{\alpha})) > \\ \vdots \end{pmatrix}}_{\underline{g}} . \tag{2.2.23}$$

To simplify notation, Equation 2.2.23 will be written henceforth as

$$F\dot{\underline{\alpha}} = \underline{g} \tag{2.2.24}$$

where F is a k by k matrix, $\dot{\underline{\alpha}}$, is a k vector of parameter differentials, and $\underline{g}$ is a k vector.

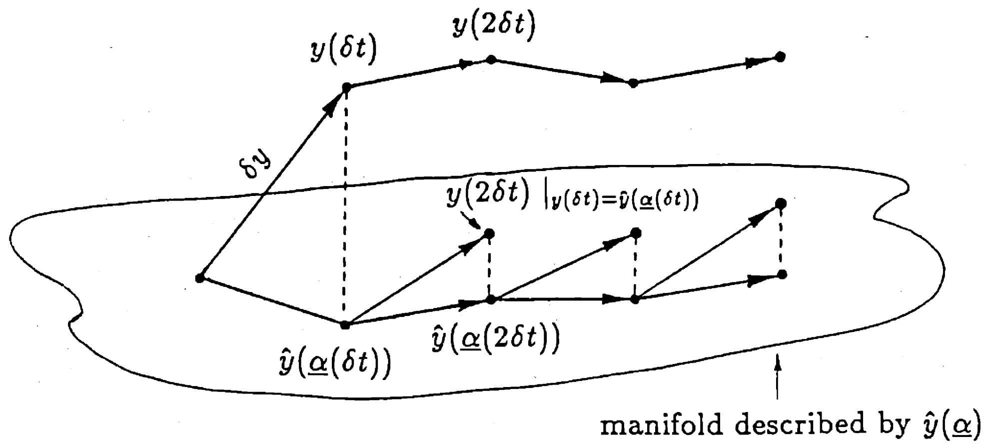


Figure 2.2.3: Progression of y and $\hat{y}(\underline{\alpha})$ with $f(y) = f(\hat{y}(\underline{\alpha}))$.

This completes our simplified derivation of the parametric approximation procedure, the final result being the $F\dot{\underline{\alpha}} = \underline{g}$ equation. The following section will mathematically formalize this derivation. For those wishing to go directly to examples and applications of the approximation procedure, this section may be skipped without any loss of continuity.

2.3 Derivation of the General Form

From the previous section we find that our scheme is basically a projection algorithm. Conceptually, our approach is to propagate the parametric approximation at some time t one infinitesimal time step forward using the equation $\frac{\partial y}{\partial t} = f(y)$. Since this does not guarentee that we remain on the parametric manifold, we must project the result back onto our manifold to retain closure in our approximation. In this section, we will consider this approximation technique from a mathematical point of view, drawing on the results of linear algebra. The derivation will show that our approximation technique can,in fact, be applied to a large class of PDE problems.

The first step is to choose a parametric function $\hat{y}(\underline{\alpha})$. The manifold described by this function is depicted in Figure 2.3.4. In order to simplify the exposition, assume initially that the true function, y, is on the manifold, i.e., that $y(\cdot, t) = \hat{y}(\cdot; \underline{\alpha}(t))$

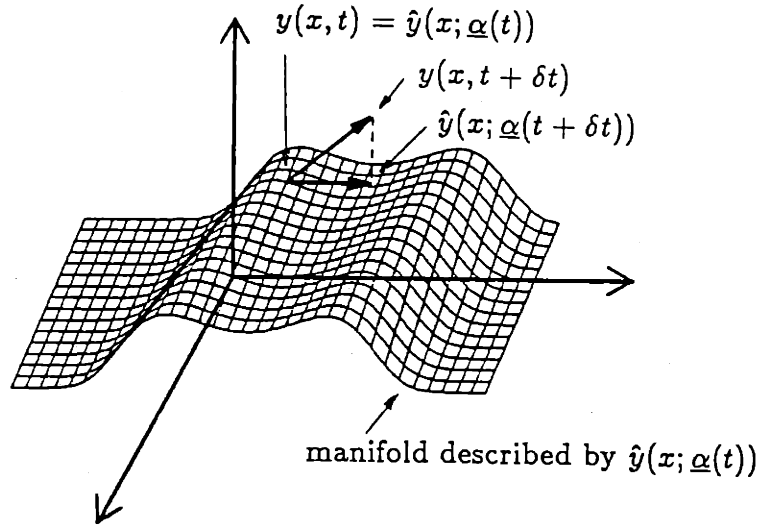


Figure 2.3.4: Manifold of Parametric Functions

for some $\underline{\alpha}(t)$. Next use the original PDE

$$\frac{\partial y}{\partial t} = f(y,x,t) \tag{2.3.1}$$

to propagate $\hat{y}(\underline{\alpha}) = y$ an infinitesimal time step δt forward, i.e.,

$$y(x,t+\delta t) = y + \delta t f(y,x,t).$$

Due to the unconstrained nature of $f(\cdot)$ this will, in general, take $y(x,t+\delta t)$ off the manifold and hence out of the realm of our parametric functions (as shown in Figure 2.3.4). In order to retain the form of our parametric function, we must approximate $y(x,t+\delta t)$ with a point on the manifold. To find the $\underline{\alpha}(t+\delta t)$ which will minimize the distance between $\hat{y}(x;\underline{\alpha}(t+\delta t))$ and $y(x,t+\delta t)$, we shall determine the projection of $y(x,t+\delta t)$ on the manifold. Specifically, to determine the $\hat{y}(x;\underline{\alpha}(t+\delta t))$ which yields the best approximation to $y(x,t+\delta t)$, we will project $(y(x,t+\delta t)-y)$ onto $(\hat{y}(x;\underline{\alpha}(t+\delta t)) - \hat{y}(\underline{\alpha}))$. This can be done as follows.

Define

$$\delta y \equiv y(x,t+\delta t) - y(x,t).$$

Now assume $\hat{y}(\underline{\alpha})$ describes a continuously differentiable manifold with respect to

$\underline{\alpha}$[1]. Then for infinitesimal

$$\delta\underline{\alpha} \equiv (\underline{\alpha}(t+\delta t) - \underline{\alpha}(t)),$$

we can define G by

$$(\hat{y}(x;\underline{\alpha}(t+\delta t)) - \hat{y}(\underline{\alpha})) \equiv G\delta\underline{\alpha} \tag{2.3.2}$$

$$\text{where } G : R^k \to C^\infty. \tag{2.3.3}$$

Hence the projection step becomes one of determining $\delta\underline{\alpha}$ so that $G\delta\underline{\alpha}$ is the best approximation to δy. Using results from linear system theory, we solve the approximation problem with

$$G^*G\delta\underline{\alpha} = G^*\delta y \tag{2.3.4}$$

where G^* is the adjoint of G. Replacing δy with its infinitesimal equivalent from Equation 2.3.1, we find

$$G^*G\delta\underline{\alpha} = G^*\delta t f(y,x,t) \tag{2.3.5}$$

$$\text{with } y = \hat{y}(\underline{\alpha}). \tag{2.3.6}$$

Hence, as desired, this equation gives us the means for determining the optimal change in $\underline{\alpha}(t)$ given the true change in y.

If we were to continue the process by propagating $y(x,t+\delta t)$, and project the resulting $y(x,t+2\delta t)$ to determine $\hat{y}(x;\underline{\alpha}(t+2\delta t))$, we find that we require the true value of $y(x,t+\delta t)$, which makes this algorithm useless for anything but the first step. Therefore, in order to retain closure, we will replace $y(x,t+\delta t)$ with $\hat{y}(x;\underline{\alpha}(t+\delta t))$ for each successive propagation step, thus making Equation 2.3.6

$$G^*G\delta\underline{\alpha}(t) = G^*\delta t f(\hat{y}(\underline{\alpha}),x,t). \tag{2.3.7}$$

To change the infinitesimal form of these equations to a differential form, we divide Equation 2.3.7 through by δt, and let $\delta t \to 0$ (so that $\frac{\delta\underline{\alpha}}{\delta t} \to \frac{d\underline{\alpha}}{dt}$), yielding

$$G^*G\dot{\underline{\alpha}}(t) = G^*f(\hat{y}(\underline{\alpha}),x,t) \tag{2.3.8}$$

$$\text{or } F\dot{\underline{\alpha}}(t) = \underline{g} \tag{2.3.9}$$

[1]This assumption is not very restrictive from a practical point of view since we are free to choose $\hat{y}(\underline{\alpha})$.

where $F = (G^*G) \in R^{k\times k}$ and $\underline{g} = (G^* f(\hat{y}(\underline{\alpha}), x, t)) \in R^k$. Hence, here we have a ordinary differential equation (ODE) which, combined with a specified parameterization $\hat{y}(\underline{\alpha})$, approximates the PDE of Equation 2.3.1. In what follows, we determine G and G^*.

To determine G we need only look at the definition of $G\delta\underline{\alpha}$. By definition:

$$G\dot{\underline{\alpha}} = \lim_{\delta t \to 0} \frac{\hat{y}(x; \underline{\alpha}(t+\delta t)) - \hat{y}(\underline{\alpha})}{\delta t} \tag{2.3.10}$$

$$= \frac{\partial \hat{y}(\underline{\alpha})}{\partial t} = \sum_{i=0}^{k} \frac{\partial \hat{y}(\underline{\alpha})}{\partial \alpha_i} \frac{d\alpha_i}{dt}. \tag{2.3.11}$$

Hence

$$G(\cdot) = \left(\tfrac{\partial \hat{y}}{\partial \alpha_0}(\cdot), \quad \cdots \quad \tfrac{\partial \hat{y}}{\partial \alpha_i}(\cdot), \quad \cdots \quad \tfrac{\partial \hat{y}}{\partial \alpha_k}(\cdot) \right) \tag{2.3.12}$$

and for the adjoint, $G^*(\cdot)$, we find

$$G^*(\cdot) = \begin{pmatrix} < \frac{\partial \hat{y}}{\partial \alpha_0}, (\cdot) >_x \\ \vdots \\ < \frac{\partial \hat{y}}{\partial \alpha_i}, (\cdot) >_x \\ \vdots \\ < \frac{\partial \hat{y}}{\partial \alpha_k}, (\cdot) >_x \end{pmatrix} \tag{2.3.13}$$

which satisfies the adjoint definition. Although the space of y is assumed to have an inner product associated with it, we have left the inner product unspecified for generality. Thus, the components of F and $\underline{g}$ can be written as

$$F_{ij} =< \frac{\partial \hat{y}}{\partial \alpha_i}, \frac{\partial \hat{y}}{\partial \alpha_j} > \text{ and } g_i =< \frac{\partial \hat{y}}{\partial \alpha_i}, f(\hat{y}, x, t) > . \tag{2.3.14}$$

In summary, the approximation is

$$\frac{\partial y}{\partial t} = f(y, x, t) \Rightarrow \begin{cases} y \approx \hat{y}(\underline{\alpha}) \\ F\dot{\underline{\alpha}} = \underline{g} \\ F_{ij} =< \frac{\partial \hat{y}}{\partial \alpha_i}, \frac{\partial \hat{y}}{\partial \alpha_j} > \\ g_i =< \frac{\partial \hat{y}}{\partial \alpha_i}, f(\hat{y}, x, t) > \end{cases}$$

with the inner products yet unspecified (this will be treated for the special case of pdf's in Chapter 3). Note that we have obtained this result with only one assumption: the continuous differentiability of $\hat{y}(\underline{\alpha})$ with respect to $\underline{\alpha}(t)$. Therefore, the results obtained in this section are applicable to a wide class of PDE's, not just the Fokker-Planck equation considered in this thesis. In the following section, we shall consider the heat equation as an example to illustrate our results.

2.4 Example

In this section we will apply the approximations derived in Sections 2 and 3 to an example to answer any remaining questions. We will use the one dimensional heat flow equation as our example. This equation is, in fact, a simplified version of the general diffusion equation we will explore later. However, this simplification will allow us to more easily examine the approximation procedure.

The one dimensional heat equation for the temperature along a rod is given by

$$\frac{\partial u(x,t)}{\partial t} = \frac{k}{c\rho}\frac{\partial^2 u(x,t)}{\partial x^2} \tag{2.4.1}$$

where $u(x,t)$ is the temperature distribution at point x along the rod at time t, c is the heat capacity, ρ is the material density, and k is the thermal conductivity. For simplicity, we will group all the constants into one constant, say r, and drop x and t from $u(x,t)$ when convenient. We now have

$$\frac{\partial u}{\partial t} = r\frac{\partial^2 u}{\partial x^2}. \tag{2.4.2}$$

The first step in reducing this equation is to choose a parameterization. This is where some a priori knowledge of the form of u is helpful. The approximating parameterization should be sufficiently complex to include the important qualitative characteristics of u, yet sufficiently simple to manipulate in determining F and g. In this example, we are fortunate to be able to compute a finite difference implementation of Equation 2.4.2 easily. The difference equation is

$$u(x,t+\delta t) = r\frac{\delta t}{(\delta x)^2}(u(x+\delta x) + u(x-\delta x)) + (1 - 2r\frac{\delta t}{(\delta x)^2})u(x,t). \tag{2.4.3}$$

Assume the initial condition $u(x,0) = 100x(2-x)$, which yields the boundary conditions $u(0,t) = u(2,t) = 0$, and choose $r\frac{\delta t}{(\delta x)^2} = 0.4$, ($\delta x = 0.1$ cm and $\delta t =$

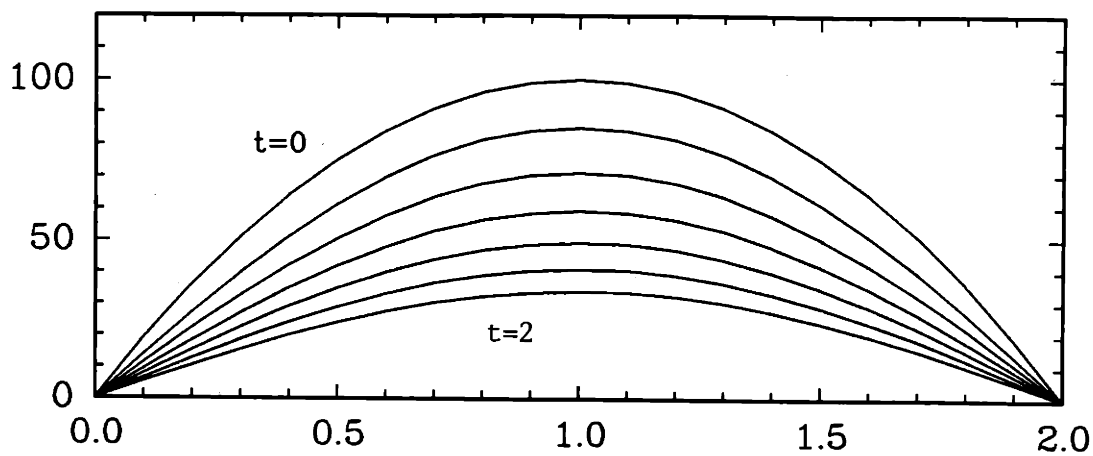


Figure 2.4.5: Heat distribution along a metal rod

0.016 sec). With these conditions, we get the results shown in Figure 2.4.5, which plots the temperature distribution in the rod for $t = 0$ to 2 seconds. Note that, although the true $u(x,t)$ is not exactly of the form $x(2-x)$ for $t > 0$, $x(2-x)$ does form a very good approximation. So we will choose as our approximating function the second order polynomial

$$\hat{u}(x,t) = \alpha_1(t)x + \alpha_2(t)x^2. \tag{2.4.4}$$

Step two in the approximation process is to choose an inner product. We will use the Euclidean inner product on C^2, modified to cover a finite range, i.e.,

$$< f_1, f_2 >= \int_a^b f_1 f_2 dx. \tag{2.4.5}$$

where f_1 and f_2 are functions of x. Substituting Equation 2.4.4 and Equation 2.4.5 in Equation 2.2.23 (restated here in terms of Equation 2.4.2)

$$\begin{pmatrix} \ddots & \cdots & \\ \vdots & < \frac{\partial \hat{u}}{\partial \alpha_i}, \frac{\partial \hat{u}}{\partial \alpha_j} > & \\ \cdots & & \end{pmatrix} \underline{\dot{\alpha}} = \begin{pmatrix} \vdots \\ < \frac{\partial \hat{u}}{\partial \alpha_i}, r\frac{\partial^2 \hat{u}}{\partial x^2} > \\ \vdots \end{pmatrix} \tag{2.4.6}$$

results in the following

$$\begin{pmatrix} \frac{a^3-b^3}{3} & \frac{a^4-b^4}{4} \\ \frac{a^4-b^4}{4} & \frac{a^5-b^5}{5} \end{pmatrix} \underline{\dot{\alpha}} = 2r\alpha_2(t) \begin{pmatrix} \frac{a^2-b^2}{2} \\ \frac{a^3-b^3}{3} \end{pmatrix}. \tag{2.4.7}$$

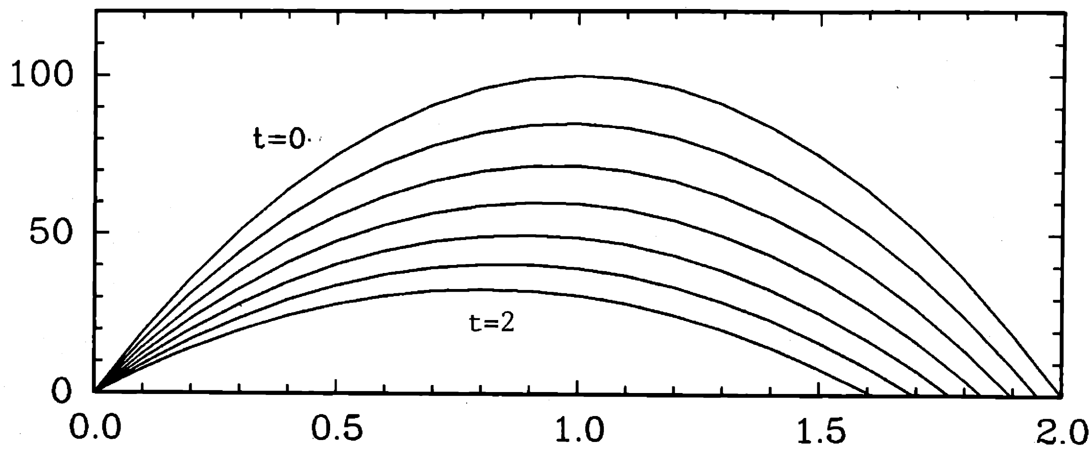


Figure 2.4.6: Heat distribution using parametric approximation

where a and b are the limits on the inner product. Since the rod only extends from 0 to 2 cm, we will choose as the limits of our inner product $a = 0$ and $b = 2$. Solving for $\dot{\underline{\alpha}}$ we find:

$$\dot{\underline{\alpha}} = \begin{pmatrix} \dot{\alpha}_1(t) \\ \dot{\alpha}_2(t) \end{pmatrix} = 2r\alpha_2(t) \begin{pmatrix} 2 \\ -5/6 \end{pmatrix}. \tag{2.4.8}$$

The final step in the approximation involves making good the assumption that the exact and approximate diffusion equations start off with the same initial condition. For our approximation, and $u(x,0) = 100x(x-2)$ as the initial condition, $\underline{\alpha}(0)$ is given by

$$\alpha_1(0) = 200, \quad \alpha_2(0) = -100. \tag{2.4.9}$$

Using this same initial condition for the exact equation (Equation 2.4.2), we are ready for a comparison. Figure 2.4.6 shows the heat distribution resulting from our approximation along the rod for $t = 0$ to 2 seconds. Note that at one end ($x = 0$), our approximation agrees with the exact result of Figure 2.4.5, while at the other end ($x = 2$), there is a considerable disagreement. This is due to the lack of a means to enforce the boundary conditions, which apply over all time, with this parameterization. At $x = 0$, this is not a problem due to the nature of our $\hat{u}$. This suggests that we choose another parameterization which could include these

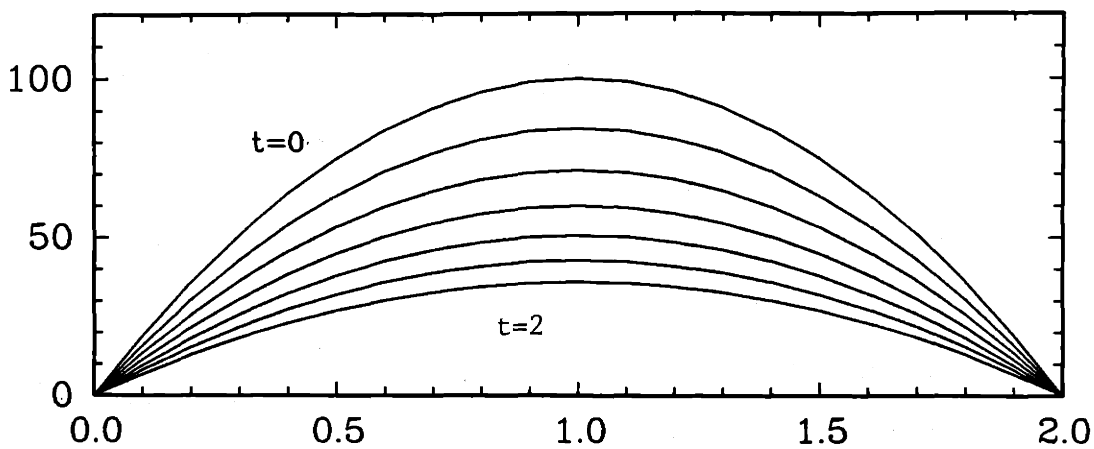


Figure 2.4.7: Approximated heat distribution

boundary constraints implicitly.[2] For the present problem, choosing

$$\hat{u}(x,t) = 100x(2-x)\alpha(t) \tag{2.4.10}$$

enforces the proper boundary constraints. Applying Equation 2.2.23 with the same choice for inner product as before, yields

$$\dot{\alpha} = -2.5r\alpha \tag{2.4.11}$$

which can be solved explicitly for α, yielding

$$\alpha(t) = e^{-2.5rt} \text{ for } 0 \leq t \leq 2 \text{ seconds} \tag{2.4.12}$$

or, for $r = 0.25$ as before,

$$\hat{u}(x,t) = 100x(2-x)e^{-(5/8)t}. \tag{2.4.13}$$

This is plotted in Figure 2.4.7. Note that the boundary conditions are met, and, at least initially, the approximate heat distributions are close to the exact ones (Figure 2.4.5). As $t > 0.5$, however, the approximation diverges from the true solution. A plot of the r.m.s. error versus time is shown in Figure 2.4.8 for both approximations. Note the vast improvement overall for the second approximation.

[2]With respect to our final application to probability density functions(pdf), this is no great task, since the boundary conditions for pdf's are that $\lim_{x\to\pm\infty} pdf(x,t) = 0$.

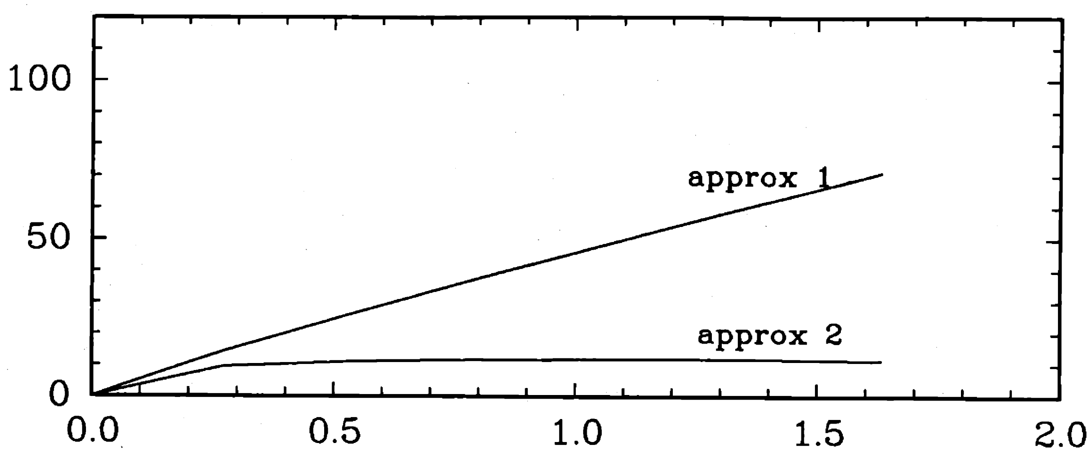


Figure 2.4.8: r.m.s. error versus time for both approximations

2.5 Ancillary Information

In this section we will make use of some of the ancillary information provided by the approximation scheme presented in Section 2. This information will be used later (Chapter 8) to aid us in determining the accuracy of our approximation.

We start by defining the error at time t as

$$\epsilon(x,t) = y(x,t) - \hat{y}(x;\underline{\alpha}(t)). \tag{2.5.1}$$

We assume that at time $t = 0$, $\hat{y}$ and y are equal (refer to Figure 2.5.9), so that the initial error is zero. Now at some small time step δ forward, the error is given by

$$\epsilon(x,\delta) = (f(\hat{y}(x;\underline{\alpha}(0))) - \hat{f}(\hat{y}(x;\underline{\alpha}(0))))\delta \tag{2.5.2}$$

where

$$f(y) = \frac{\partial y}{\partial t} \text{ and } \hat{f}(\hat{y}) = \frac{\partial \hat{y}}{\partial \underline{\alpha}}^T \dot{\underline{\alpha}}. \tag{2.5.3}$$

Forming the difference, $\epsilon(x,\delta) - \epsilon(x,0)$, dividing through by δ, and letting $\delta \to 0$, we find the derivative or error rate at $t = 0$ to be given by

$$\frac{\partial \epsilon(x,t)}{\partial t}\Big|_{t=0} = f(\hat{y}) - \hat{f}(\hat{y}). \tag{2.5.4}$$

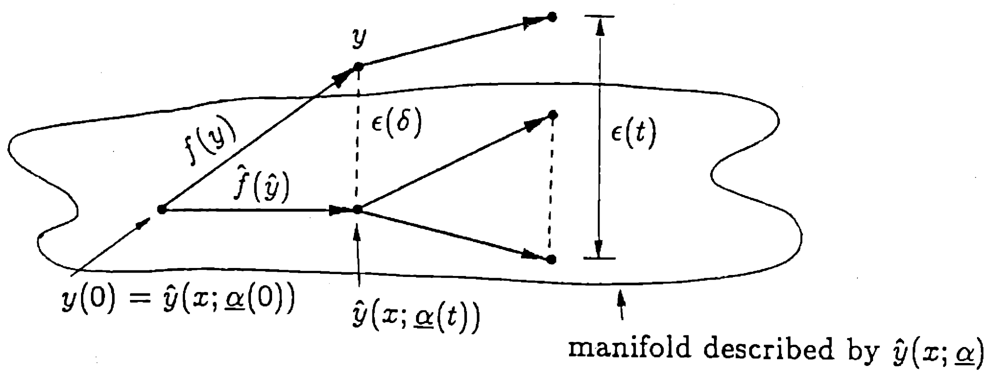


Figure 2.5.9: View of an approximation step

In words, Equation 2.5.4 describes the rate at which the error in our approximation increases at any point $\hat{y}$ on the manifold of parametric functions. Put another way, since our approximation scheme guarantees that we remain on the manifold, Equation 2.5.4 is the rate of error for our approximation. Hence, although this quantity does not describe the true error between y and the approximation, $\hat{y}$, at any time t, it does tell us the rate at which our error increases at any time t. For comparison, we shall actually study the norm of the error rate as given by

$$\|f(\hat{y}(x; \underline{\alpha}(t))) - \hat{f}(\hat{y}(x; \underline{\alpha}(t)))\| \tag{2.5.5}$$

In Chapter 8 we will not only evaluate Equation 2.5.5 for specific parameterizations, but will also use it to determine the performance of various parameterizations.

2.6 Conclusion

In summary, we have developed in this chapter an approach to approximating solutions to partial differential equations of the form

$$\frac{\partial y(x,t)}{\partial t} = f(y(x,t)) \tag{2.6.1}$$

with parametric functions of the form

$$\hat{y}(x;\underline{\alpha}(t)) \tag{2.6.2}$$

where the parameter vector $\underline{\alpha}(t)$ completely specifies $\hat{y}$. The basic result is a differential equation for the parameters

$$\dot{\underline{\alpha}} = F^{-1}\underline{g} \tag{2.6.3}$$

where F is a matrix function which depends on the choice of parameterization of $\hat{y}(\underline{\alpha})$, and $\underline{g}$ is a vector function of both the parameterization and $f(\cdot)$ above. In Section 5, we used the basic premise of our approximation (projection) to obtain a quantity relating the rate at which the error accumulates in our approximation to the approximation itself, $\hat{y}(\underline{\alpha})$. This quantity will be evaluated and studied further in Chapter 8. In the next chapter, we shall specialize the results of this chapter to probability density functions (pdf's) and take advantage of the interpretations that come with pdf's. In later chapters, we shall apply the results of Chapters 2 and 3 to specific examples with the purpose of determining the pros and cons of our scheme.

Chapter 3

Specialization to Pdfs

3.1 PDF's

In the previous chapter, we presented an approach to approximating solutions to general partial differential equations (PDE's) using parametric methods. In this chapter, we specialize the results of Chapter 2 to the case where the PDE describes the evolution of probability density functions (pdf's) on the state space of some stochastic process. These pdf's have the following properties:

$$p(x,t) > 0 \quad \forall \quad x,t \qquad \text{property 1}$$

$$\int_{-\infty}^{+\infty} p(x,t)dx = \begin{cases} 1 & \text{normalized pdf's} \\ c(t) & \text{unnormalized pdf's} \end{cases} \qquad \text{property 2}$$

$$\lim_{x \to \pm\infty} p(x,t) \to 0 \qquad \text{property 3}$$

The stochastic process of concern is described by the following model:

$$dx = \mathbf{f}(x)dt + dw \tag{3.1.1}$$

where x = state of process

$\mathbf{f}(\cdot)$ = noise-free dynamics

dw = zero mean noise with intensity Q

(Note: We leave the case where Q is dependent on x as an extension for the reader.) Using the special properties of pdf's, we will choose the error criteria and inner products which will minimize the effort involved in obtaining approximate solutions (i.e. computing $F\dot{\underline{\alpha}} = \underline{g}$). These special properties, along with a priori knowledge of the general characteristics of the underlying pdf's, will also be used to motivate parameterizations $\hat{y}(x;\underline{\alpha}(t))$. Before we proceed, we will modify the notation of Chapter 2 to emphasize the fact that we are dealing with pdf's. These definitions are summarized below:

$$p(x,t) = y(x,t) \quad \text{pdf given by Fokker-Planck or Zakai equation} \tag{3.1.2}$$

$$\hat{p}(x;\underline{\alpha}(t)) = \hat{y}(x;\underline{\alpha}(t)) \quad \text{approximating pdf to be determined} \tag{3.1.3}$$

$$\underline{\alpha}(t) \in R^k \quad \text{vector of parameters which describes } \hat{p}(x;\underline{\alpha}(t)) \tag{3.1.4}$$

In Section 2 we will motivate and choose two error criteria for the pdf approximation problem. These choices will actually involve reformulating the approximation step to simplify calculations. Section 3 will describe the structure of the F matrix under these two new formulations and Section 4 will do the same for the $\underline{g}$ vector. Since the $\underline{g}$ term also depends on the PDE describing the pdf (unlike the F matrix), this section will describe $\underline{g}$ for two important pdf equations, namely the Fokker-Planck and Zakai equations. Section 5 will consider the numerical problems associated with implementing the $F\dot{\underline{\alpha}} = \underline{g}$ equation, as well as possible simplifications. Finally, Section 6 summarizes the results of this chapter.

3.2 Error Criteria

Sections 2.2 and 2.3 derived $F\dot{\underline{\alpha}} = \underline{g}$ for a rather general class of PDE's, with an obvious, yet specific, error criterion, and with an unspecified inner product on the set $\mathcal{Y}$. This section will consider modifications of the error criterion used in Sections 2.2 and 2.3 while assuming the standard Euclidean inner product. Our requirements for choosing a particular modification are that the resulting error criterion (a) captures a useful measure of distance between two pdf's, and (b) leads to easily computable F and $\underline{g}$ terms. The approach will be to address the computational issue in order to find an error criterion which simplifies the calculations of F and $\underline{g}$, and then to determine whether this error criterion meets (a). We have adopted this approach to take advantage of the 'soft' constraint of (a), which will give us much leeway with respect to the 'hard' (b) constraint.

Square Root Formulation

Expressions involving the inner product, i.e., the F and $\underline{g}$ terms, involve derivatives of the pdf of the form

$$\frac{\partial \hat{p}(x;\underline{\alpha}(t))}{\partial \alpha_i}. \tag{3.2.1}$$

A typical F term has the form

$$F_{ij} =< \frac{\partial \hat{p}(x;\underline{\alpha}(t))}{\partial \alpha_i}, \frac{\partial \hat{p}(x;\underline{\alpha}(t))}{\partial \alpha_j} > \tag{3.2.2}$$

while a typical $\underline{g}$ term has the form

$$g_i =< \frac{\partial \hat{p}(x;\underline{\alpha}(t))}{\partial \alpha_i}, f(\hat{p}(x;\underline{\alpha}(t))) > . \tag{3.2.3}$$

Without loss of generality, we can let

$$\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}(x;\underline{\alpha}(t))} \tag{3.2.4}$$

for some $\hat{\varsigma}(x;\underline{\alpha}(t))$ and substitute it into the differential (Equation 3.2.1), which yields

$$\frac{\partial \hat{p}(x;\underline{\alpha}(t))}{\partial \alpha_i} = \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}\hat{p}(x;\underline{\alpha}(t)). \tag{3.2.5}$$

This yields a F matrix term of the form

$$F_{ij} =< \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}\hat{p}(x;\underline{\alpha}(t)), \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_j}\hat{p}(x;\underline{\alpha}(t)) > . \tag{3.2.6}$$

Using the standard inner product

$$< f_1, f_2 >= \int_{-\infty}^{+\infty} f_1 f_2 dx \tag{3.2.7}$$

where f_1 and f_2 are arbitrary functions of x, F becomes

$$F_{ij} = \int \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}\frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_j}\hat{p}(x;\underline{\alpha}(t))^2 dx. \tag{3.2.8}$$

As it stands, this form is, in general, difficult to evaluate. However, if $\hat{p}(x;\underline{\alpha}(t))^2$ could be interpreted as a pdf, F_{ij} can be interpreted as an expected value of some quantity over the approximating pdf $\hat{p}(x;\underline{\alpha}(t))^2$. We can induce this change by modifying the error criterion to one which minimizes the error between the square roots of the densities. Equivalently, we may reformulate the original problem in terms of $\sqrt{p(x,t)}$ instead of $p(x,t)$. Thus the error criterion becomes

$$\text{error criterion} = \|\sqrt{p(x,t)} - \sqrt{\hat{p}(x;\underline{\alpha}(t))}\| \tag{3.2.9}$$

$$\text{with inner product } < f_1, f_2 >= \int f_1 f_2 dx \tag{3.2.10}$$

This is a 'sensible' error criterion since $\sqrt{p(x,t)}$ retains the same relative amplitude characteristics of $p(x,t)$. Note also that pdf property 1, $p(x,t) > 0$, guarantees a well behaved square root. This new error criterion will not affect any of the derivations thus far since they have been carried out with all functions unspecified. However, to be consistent with the derivations, we must re-write the governing partial differential equation in terms of $\sqrt{\hat{p}(x;\underline{\alpha}(t))}$ instead of $\hat{p}(x;\underline{\alpha}(t))$.

For example, in the heat equation of Chapter 2, replacing u with, say, v^2 yields

$$\frac{\partial v^2}{\partial t} = r\frac{\partial^2 v^2}{\partial x^2} \tag{3.2.11}$$

$$\text{or } \frac{\partial v}{\partial t} = r(\frac{\partial^2 v}{\partial x^2} + \frac{1}{v}(\frac{\partial v}{\partial x})^2). \tag{3.2.12}$$

Hence it would be Equation 3.2.12 to which our approximation would be applied.

In order to simplify further calculations, define

$$q(x,t) = \sqrt{p(x,t)} \tag{3.2.13}$$

$$\hat{q}(x;\underline{\alpha}(t)) = \sqrt{\hat{p}(x;\underline{\alpha}(t))} = e^{\hat{s}(x;\underline{\alpha}(t))} \tag{3.2.14}$$

Note that we have implicitly redefined $\hat{p}(x;\underline{\alpha}(t))$ as $e^{2\hat{s}(x;\underline{\alpha}(t))}$ to simplify dealings with $\hat{q}(x;\underline{\alpha}(t))$. Note that, however, when computing inner products as expected values, these expectations must be taken over $\hat{p}(x;\underline{\alpha}(t)) = e^{2\hat{s}(x;\underline{\alpha}(t))}$. Hence care must be exercised in keeping track of which function is being manipulated, i.e. $\hat{p}(x;\underline{\alpha}(t))$ or $\sqrt{\hat{p}(x;\underline{\alpha}(t))}$.

Using Equations 3.2.14 to re-evaluate our $F\underline{\dot{\alpha}} = \underline{g}$ equation to minimize $\|\hat{q}(x;\underline{\alpha}(t)) - q(x,t)\|$, we find that the F matrix terms become

$$F_{ij} = E\{\frac{\partial\hat{\varsigma}(x;\underline{\alpha}(t))}{\partial\alpha_i}\frac{\partial\hat{\varsigma}(x;\underline{\alpha}(t))}{\partial\alpha_j}\} \tag{3.2.15}$$

with the expectation being over $\hat{p}(x;\underline{\alpha}(t))$. As we will see later, the same advantages afforded to the F matrix will be afforded to the $\underline{g}$ vector. We will not detail them here, but defer them to Section 3.4.

Summarizing the results of this subsection, we have chosen the Euclidean inner product and reformulated the original approximation procedure to manipulate $\hat{q}(x;\underline{\alpha}(t))$, the square root of the probability densities. This led to simplified inner products which can be interpreted as expectations. In the next subsection we will determine an alternative reformulation (but with the same Euclidean inner product) which also simplifies the computation of the $F\underline{\dot{\alpha}} = \underline{g}$ equation.

Logarithmic Formulation

Since our main goal is simple approximating solutions to PDE's, and is not constrained to specific error criteria, we have the flexibility to modify the original problem statement. We took advantage of this fact in the previous section by reformulating the approximation to deal with the square root of pdf's. This concept of minimizing the distance between functions of pdf's can be taken a step further, as follows.

If we assume $\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$, as above, and further assume that $p(x,t) = e^{\varsigma(x,t)}$, then we may minimize the distance between the logarithm of the two pdf's:

$$\|\hat{\varsigma}(x;\underline{\alpha}(t)) - \varsigma(x,t)\|. \tag{3.2.16}$$

This approach leads to inner products in the expressions for F and $\underline{g}$ which are only functions of the derivatives of $\hat{\varsigma}(x;\underline{\alpha}(t))$, i.e.,

$$F_{ij} = < \frac{\partial\hat{\varsigma}(x;\underline{\alpha}(t))}{\partial\alpha_i}, \frac{\partial\hat{\varsigma}(x;\underline{\alpha}(t))}{\partial\alpha_j} > \tag{3.2.17}$$

$$g_i = < \frac{\partial\hat{\varsigma}(x;\underline{\alpha}(t))}{\partial\alpha_i}, \tilde{f}(\hat{\varsigma}(x;\underline{\alpha}(t))) > . \tag{3.2.18}$$

The function $\tilde{f}()$ is the new RHS of Equation 2.2.1, determined after applying the substitution $p = e^{\varsigma}$, and is given by

$$\frac{\partial \varsigma(x,t)}{\partial t} = \tilde{f}(\varsigma(x,t)). \tag{3.2.19}$$

Since Equations 3.2.17 and 3.2.18 do not explicitly involve pdf's, as was true previously (Equations 3.2.6, 3.2.3), we are free to choose the inner product which makes the most sense from the error criteria viewpoint.

A possible family of inner products might be

$$< f_1, f_2 >= \int_{-\infty}^{+\infty} f_1 f_2 w(x) dx. \tag{3.2.20}$$

where $w(x)$ is some weighting function chosen to emphasize or de-emphasize various parts of the space in which x lives, or in our case, various parts of the state space of the underlying stochastic process.

A necessary constraint on $w(x)$, in order to keep the inner products of the logarithms of pdf's bounded, is that $w(x) \rightarrow 0$ as $x \rightarrow \pm\infty$. This is due to pdf Property 3, which implies that $\lim_{x \rightarrow \pm\infty} \varsigma(x,t) \rightarrow -\infty$. This, in turn, implies that the terms in F and $\underline{g}$ corresponding to f_1, f_2 are unbounded below. Hence, in order for F and g to remain bounded, $w(x)$ must approach zero at a faster rate than f_1 and f_2 approach $-\infty$. Assuming that most of the mass of the true and approximating pdfs does not lie far from zero, the overall effect of this constraint on $w(x)$ is to de-emphasize the tails of our pdf's and hence the distance or error at the tails between the approximate and true pdf. This effect may be acceptable in many situations. For situations where the mean lies far from zero, $w(x)$ could easily be modified to account for the shift.

A function satisfying the properties above is a negative exponential. Although the specific form of the function depends on the parameterization we choose, for a large group of functions, replacing $w(x)$ with a Gaussian keeps the inner products bounded.

The motivation for choosing such a $w(x)$ is the same as it was for the square root formulation, namely to permit the interpretation of the inner products as expectations. Here the expectations clearly depend on the first two moments of the Gaussian pdf used for $w(x)$.

error criterion	inner product
$\|\sqrt{p(x,t)} - \sqrt{\hat{p}(x;\underline{\alpha}(t))}\|$	$< f_1, f_2 >= \int_{-\infty}^{\infty} f_1 f_2 dx$
$\|log(p(x,t))w^{\frac{1}{2}}(x) - log(\hat{p}(x;\underline{\alpha}(t)))w^{\frac{1}{2}}(x)\|$ $w(x) \sim N(x;0,b)$	SAME

Figure 3.2.1: Summary of error criteria

For example, if we are not concerned with minimizing the errors at the tails of the pdf, choosing $w(x)$ to be Gaussian with zero mean does not seem unreasonable. The variance of the Gaussian can be used to emphasize a specific range of x. Specifically, we can write

$$< f_1, f_2 >= \int_{-\infty}^{+\infty} f_1 f_2 N(x;0,b)dx = \underbrace{E}_{N(x;0,b)} \{f_1 f_2\} \tag{3.2.21}$$

where $b \in R^1$ is the variance of the weighting function $w(x)$.

For consistency with the square root formulation, we shall reformulate the problem to reflect the effects of $w(x)$ in the error criterion instead of in the inner product, and thus will retain the same inner product as in the square root formulation. Hence, we will choose as our second error criterion the Euclidean distance between the logarithms of the true and parametric pdf's multiplied by $w^{\frac{1}{2}}(x)$.

We summarize the results of this section in the Table 3.2.1.

Expected Values

From this point on, we shall assume that expected values are (a) over the square of the parameterization, $\hat{q}(x;\underline{\alpha}(t))^2 = \hat{p}(x;\underline{\alpha}(t)) = e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$, when referring to the square root formulation, and (b) over $w(x) = N(x;0,b)$ when referring to the logarithmic formulation. Especially in the former, there is the possibility for confusion, in particular, taking expectations over $e^{\hat{\varsigma}(x;\underline{\alpha}(t))} = \hat{q}(x;\underline{\alpha}(t))$ instead of over $e^{2\hat{\varsigma}(x;\underline{\alpha}(t))} = \hat{q}(x;\underline{\alpha}(t))^2$, since they only differ by a factor of two in the exponent.

In the following sections we will apply these inner products and reformulations to the nonlinear filtering problem.

error criterion	$\hat{p}(x;\underline{\alpha}(t))$	F_{ij}	$E\{\ \}$ over
square root	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\}$	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$
log	$e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\}$	$N(0,b)$

Figure 3.3.2: F matrix summary.

3.3 F matrix

In this section we will describe the structure of the F matrix in more detail. We will start with describing the matrix under the square root error criterion, and then present it under the logarithmic error criterion.

Expanding the F matrix using $\hat{q}(x;\underline{\alpha}(t)) = \sqrt{\hat{p}(x;\underline{\alpha}(t))} = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$, and using the results of the previous section, we find

$$F = \begin{pmatrix} \ddots & & \cdots \\ \vdots & < \frac{\partial \hat{q}(x;\underline{\alpha}(t))}{\partial \alpha_i}, \frac{\partial \hat{q}(x;\underline{\alpha}(t))}{\partial \alpha_j} > & \\ \cdots & & \end{pmatrix} \tag{3.3.1}$$

$$= E\begin{pmatrix} \hat{\varsigma}_{\alpha_1}\hat{\varsigma}_{\alpha_1} & \cdots & \hat{\varsigma}_{\alpha_1}\hat{\varsigma}_{\alpha_k} \\ \vdots & \ddots & \vdots \\ \hat{\varsigma}_{\alpha_k}\hat{\varsigma}_{\alpha_1} & \cdots & \hat{\varsigma}_{\alpha_k}\hat{\varsigma}_{\alpha_k} \end{pmatrix} \tag{3.3.2}$$

where $\hat{\varsigma}_{\alpha_i} = \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}$ and the expected values, $E\{\ \}$, are over the pdf $\hat{p}(x;\underline{\alpha}(t))$. As can be seen, the F matrix depends only on $e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$. This means that for a given parameterization $\hat{q}(x;\underline{\alpha}(t))$ (or equivalently $\hat{\varsigma}(x;\underline{\alpha}(t))$), the F matrix (Equation 3.3.2) remains unchanged regardless of our PDE model (Equation 3.1.1).

As we will see in the next section, the effects of the PDE model are reflected in the $\underline{g}$ term. Before we do so, we note that for the logarithmic error criterion, the only changes in Equation 3.3.2 are that the expectations are now over the weighting pdf $w(x)$, which we have chosen to be Gaussian with zero mean and covariance b. Also note that, for the logarithmic error criterion, the $\hat{\varsigma}(x;\underline{\alpha}(t))$ differs from that for the square root error criterion by a factor of 2, since for the log criterion, we have assumed $e^{\hat{\varsigma}(x;\underline{\alpha}(t))} = \hat{p}(x;\underline{\alpha}(t))$ (not $\hat{q}(x;\underline{\alpha}(t))$) for computational simplicity.

Table 3.3.2 summarizes the results of this section.

3.4 $\underline{g}$ vector

In this section we will describe the structure of the $\underline{g}$ vector in more detail. We will again start by describing the vector under the square root error criterion, and then present it under the logarithmic error criterion.

Expanding the $\underline{g}$ vector using $\hat{q}(x;\underline{\alpha}(t)) = \sqrt{\hat{p}(x;\underline{\alpha}(t))} = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$, and using the results of Section 2, we find

$$\underline{g} = \begin{pmatrix} \vdots \\ < \frac{\partial \hat{q}(x;\underline{\alpha}(t))}{\partial \alpha_i}, \tilde{f}(\hat{q}(x;\underline{\alpha}(t))) > \\ \vdots \end{pmatrix} \tag{3.4.1}$$

$$= E\begin{pmatrix} \hat{\varsigma}_{\alpha_1}\tilde{f}(e^{\hat{\varsigma}})e^{-\hat{\varsigma}} \\ \vdots \\ \hat{\varsigma}_{\alpha_k}\tilde{f}(e^{\hat{\varsigma}})e^{-\hat{\varsigma}} \end{pmatrix} \tag{3.4.2}$$

where $\hat{\varsigma}_{\alpha_i} = \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}$, $\tilde{f}()$ is defined by $\frac{\partial q(x,t)}{\partial t} = \tilde{f}(q(x,t))$, and the expected values, $E\{\ \}$, are over $e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$. The seemingly complicated form of $\underline{g}$ is used to allow it to be interpreted as an expectation. $\tilde{f}()$ is a new function which results when we re-formulate the original PDE for $p(x,t)$ in terms of $q(x,t)$. We shall see later in this section a simplification in this term.

For the logarithmic error criterion, the expectation is over $w(x) \sim N(0,b)$ and $\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$, not $e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$. In addition, since ς takes the place of q, $\tilde{f}()$ is now given by $\tilde{f}(\varsigma(x,t)) = \frac{\partial \varsigma(x,t)}{\partial t}$. Taking these facts into account, $\underline{g}$ becomes

$$\underline{g} = \begin{pmatrix} \vdots \\ < \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i}, \tilde{f}(\hat{\varsigma}(x;\underline{\alpha}(t))) > \\ \vdots \end{pmatrix} \tag{3.4.3}$$

$$= E\begin{pmatrix} \hat{\varsigma}_{\alpha_1}\tilde{f}(\hat{\varsigma}) \\ \vdots \\ \hat{\varsigma}_{\alpha_k}\tilde{f}(\hat{\varsigma}) \end{pmatrix}. \tag{3.4.4}$$

In all the forms of $\underline{g}$, it is the dependence on $\tilde{f}()$ that makes the computation of $\underline{g}$ the most difficult part of the approximation process. In the following subsection, we will proceed to describe the $\underline{g}$ vector for the pdf diffusion equations which describe $f()$.

Diffusion Operators

Referring to the section in Chapter 1 describing the pdf diffusion operators, we see that there were two main equations which describe the evolution of pdf's for filtering problems, namely the Fokker-Planck and Zakai equations. The two are identical except for a measurement term in the latter. The Fokker-Planck equation considers only the dynamical portion of the stochastic model, leaving the measurement to be applied via some other means, such as Bayes' rule. The Zakai equation, on the other hand, includes the measurement directly as part of the equation. In any case, the most complicated and, in fact, the common portion of these equations is encompassed in the Fokker-Planck equation.

Hence, throughout this work, we will concentrate on the dynamical portion of the pdf differential equation, although when convenient, we shall include the measurement terms in the derivations. We restate the Zakai equation here for continuity and to establish notation:

$$\frac{\partial p(x,t)}{\partial t} = \underbrace{\underbrace{-\sum_i \frac{\partial \mathbf{f}_i(x)p(x,t)}{\partial x_i}}_{\text{process dynamics}} + \underbrace{(1/2)\sum_{i,j} Q_{ij}\frac{\partial^2 p(x,t)}{\partial x_i \partial x_j}}_{\text{process noise}}}_{\text{Fokker-Planck portion}} + \underbrace{h^T(z - \frac{1}{2}h)p(x,t)}_{\text{measurement term}} \,. \tag{3.4.5}$$

where

$$x, \mathbf{f}(x) \in R^n \quad Q \in R^{n\times n} \quad z \in R^m \quad h \in R^{n\times m} \tag{3.4.6}$$

Some additional notation which will be used throughout this section is defined here:

$$\mathbf{f}_{i_{x_i}} = \frac{\partial \mathbf{f}_i(x)}{\partial x_i} \tag{3.4.7}$$

$$\hat{\varsigma}_{\alpha_i} = \frac{\partial \hat{\varsigma}(x;\underline{\alpha}(t))}{\partial \alpha_i} \tag{3.4.8}$$

$$\varsigma_t = \frac{\partial \varsigma(x,t)}{\partial t} \tag{3.4.9}$$

$$\varsigma_{x_i} = \frac{\partial \varsigma(x,t)}{\partial x_i} \tag{3.4.10}$$

$$\varsigma_{x_i x_j} = \frac{\partial^2 \varsigma(x,t)}{\partial x_i \partial x_j} \tag{3.4.11}$$

We will start by studying the $\underline{g}$ term under the square root error criterion. The first step is to rewrite the Zakai equation in terms of $q(x,t)$ by substituting $p(x,t) = q(x,t)^2$. This results in

$$\begin{aligned}\frac{\partial q}{\partial t} &= -\sum_i \left(\frac{1}{2}\frac{\partial \mathbf{f}_i}{\partial x_i} q + \frac{\partial q}{\partial x_i}\mathbf{f}_i\right) + \frac{1}{2}\sum_{i,j} Q_{ij}\left((1/q)\frac{\partial q}{\partial x_i}\frac{\partial q}{\partial x_j} + \frac{\partial^2 q}{\partial x_i \partial x_j}\right) \\ &\quad + \frac{1}{2}h^T\left(z - \frac{1}{2}h\right)q \end{aligned} \tag{3.4.12}$$

where we have used the shorthand notation $q = q(x,t)$. Replacing q with e^{ς} (again using shorthand notation for $\varsigma(x,t)$) as before, Equation 3.4.12 becomes

$$\frac{\partial e^{\varsigma}}{\partial t} = \tilde{f}(e^{\varsigma}) \tag{3.4.13}$$

$$= \left(-\sum_i\left(\frac{1}{2}\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i\right) + \frac{1}{2}\sum_{i,j} Q_{ij}\left(2\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}\right) + \frac{1}{2}h^T\left(z - \frac{1}{2}h\right)\right)e^{\varsigma} \tag{3.4.14}$$

$$= \varsigma_t e^{\varsigma}. \tag{3.4.15}$$

Replacing ς with $\hat{\varsigma}$ and using this ς_t in Equation 3.4.2 for $\underline{g}$ we find

$$\underline{g} = E\begin{pmatrix} \hat{\varsigma}_{\alpha_1}\hat{\varsigma}_t \\ \vdots \\ \hat{\varsigma}_{\alpha_k}\hat{\varsigma}_t \end{pmatrix} \tag{3.4.16}$$

where the expected value is over $\hat{p}(x;\underline{\alpha}(t)) = e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$.

For the logarithmic case we again find the same form for $\underline{g}$; however $\tilde{f}()$ is now obtained by substituting $p(x,t) = e^{\varsigma(x,t)}$ into Equation 3.4.5 and reformulating in terms of $\varsigma(x,t)$, i.e.,

$$\frac{\partial \varsigma}{\partial t} = \tilde{f}(\varsigma) \tag{3.4.17}$$

$$= -\sum_i\left(\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i\right) + \frac{1}{2}\sum_{i,j} Q_{ij}\left(\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}\right) + h^T\left(z - \frac{1}{2}h\right). \tag{3.4.18}$$

The expectations in Equation 3.4.16 are now over $w(x) \sim N(0,b)$, and our final approximation for $p(x,t)$ is $\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$.

Note that in both cases, the important equation is the one for ς_t and that, apart from small differences in ς_t, the only difference in $\underline{g}$ between the different error criteria is in the underlying density used for the expectations. So, except for the expectations, $\underline{g}$ is expressed in terms of the exponent of our parameterization (since ς_t from Equation 3.4.15 is only a function of ς, not e^{ς}). Hence the $\underline{g}$ term (as well as the F matrix) will simply be a combination of the moments of the exponent of our parameterization. Table 3.4.3 summarizes these results. In the following section we will consider some of the problems associated with the actual computation of F and $\underline{g}$.

3.5 Application Issues

Implementation Problems

In this section we will consider the issues in implementing $F\dot{\underline{\alpha}} = \underline{g}$. We will consider issues of dimensionality, i.e., the size of the F matrix, integration in the F and $\underline{g}$ elements, coordinate transformations, and ordering of elements in data structures. These issues can be used to further direct the choice of a parameterization.

The size of the F matrix is $k \times k$, where k is the number of parameters in our approximation. Note that this number increases quickly when we consider vector processes. For example, if $\underline{x} \in R^n$, then for a second order parameterization, i.e., $e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$ for $\hat{\varsigma}(x;\underline{\alpha}(t)) = \alpha_0 + \underline{\alpha}_1^T\underline{x} + \underline{x}^T\alpha_2\underline{x}$, where α_2 is a $n \times n$ matrix of parameters, the number of parameters is given by

$$k = \underbrace{1}_{\alpha_0} + \underbrace{n}_{\underline{\alpha}_1} + \underbrace{\frac{n(n+1)}{2}}_{\alpha_2}. \tag{3.5.1}$$

We see that as n increases, k goes up as $O(n^2)$ due to the third term, and hence the size of the F matrix goes up as $O(n^4)$. However, we shall see that for certain situations, e.g., linear systems, the computational burden of our approximation scheme reduces to $O(n^2)$, equivalent to that of the EKF. In general, for an Mth order parameterization, we find k to be

$$k = 1 + \sum_{m=1}^{M} \frac{(n+m-1)!}{m!(n-1)!} \tag{3.5.2}$$

error criterion	$\hat{p}(x;\underline{\alpha}(t))$	g_i	expectation over
square root	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\}$	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$
log	$e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\}$	$N(0,b)$

error criterion	Fokker-Planck $\hat{\varsigma}_t$
square root	$-\sum_i(\frac{1}{2}\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(2\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j})$
log	$-\sum_i(\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j})$

error criterion	Zakai $\hat{\varsigma}_t$
square root	$-\sum_i(\frac{1}{2}\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(2\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}) + \frac{1}{2}h^T(z - \frac{1}{2}h)$
log	$-\sum_i(\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}) + h^T(z - \frac{1}{2}h)$

Figure 3.4.3: $\underline{g}$ vector summary.

resulting in an F matrix of $O(n^{2M})$. This all suggests that we should be very frugal in choosing a particular parameterization.

The next issue concerning the F matrix is in the computation of its elements. These issues will also apply to the elements of the $\underline{g}$ vector, although these will in general be more complicated than the F elements due to the generality of the dynamical model, $f()$, on which $\underline{g}$ is based. Each element of either F or $\underline{g}$ involves an integration over all x. If we had to perform this integration for each element, there would be little or no advantage to our filter implementation over implementing the pdf diffusion operator directly.

In the previous derivations of the F matrix and $\underline{g}$ vector we wrote the elements in terms of the moments of either the approximating pdf itself or some weighting function interpreted as a pdf. In the case of the square root error criterion, we must be able to write these moments in terms of the parameters, i.e., $m = m(\underline{\alpha})$, to maintain closure. The following equations summarize the relations needed to obtain this closure property:

$$F(\underline{m})\dot{\underline{\alpha}} = g(\underline{m}) \tag{3.5.3}$$

$$\underline{m} = \underline{m}(\underline{\alpha}) \tag{3.5.4}$$

$$\rightarrow \quad F(\underline{\alpha})\dot{\underline{\alpha}} = g(\underline{\alpha}) \tag{3.5.5}$$

When this simplification is not possible, i.e., a closed form expression for Equation 3.5.4 does not exist (as it may be with pdf's of higher than second order, where little or no work exists concerning moments for general classes of pdf's), another minimization criterion may have to be chosen to keep F computable. This is the case for the logarithmic error criterion discussed previously. Here, regardless of the complexity of $\hat{p}(x;\underline{\alpha}(t))$, the expectations involved in F and $\underline{g}$ are still over a Gaussian pdf (or whatever we have chosen for $w(x)$), and are thus computable. Under the square root error criterion, the expectations are over $\hat{p}(x;\underline{\alpha}(t))$ and therefore may not be computable without resorting to numerical integration techniques[1]

The final issue we will consider in this section will be ordering; namely, for a vec-

[1] The advantage of this error criterion is that the resulting inner product is well understood and does not require the selection of the width parameter b.

tor process, how do we order the parameters of the n dimensional pdf in data structures? For example, if we use the multi-dimensional Gaussian pdf for $\hat{p}(x;\underline{\alpha}(t))$, do we equate the parameters $\alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4 ....$ with $1, m_1, m_2, m_3, m_4 ... P_{11} ..$ or with $1, m_1, P_{11}, P_{12}, P_{22} .....$? Theoretically the performance of our approximation is not dependent on this ordering. Each m_i or P_{ij} will evolve the same way regardless of how we associate parameter indices with the various parameters. However, when we attempt to simplify F and g, or seek any intuition about a particular feature of our approximation (e.g. covariance or mean), the proper ordering will help make computations more efficient. Take, for example, the case where two of the processes are uncoupled, or nearly so, so that we would expect any cross-correlation terms in the F matrix to be zero. An ordering like $1, m_1, m_2, P_{11}, P_{12}, P_{22}$ would bring this structure to light. For this particular example, ideally, we would like to think of the F matrix as a tensor $F(i,j,k,l..)$ where $F(i,1,1,1..)$ describes all single cross terms m_i, $F(i,j,1,1....)$ all second cross terms P_{ij}, $F(i,j,k,1,1..)$ all third order terms T_{ijk}, etc. But since such a system is difficult to implement, we will adhere to matrix representations and use an ordering corresponding to $1, m_1, m_2, ..., P_{11}, P_{12}, .., T_{111}, T_{112}, T_{122}, ...$ for $\alpha_0, \alpha_1, \alpha_2 ... \alpha_{11}, \alpha_{12} .. \alpha_{111}, \alpha_{112}, \alpha_{122}$, which will group together the 1st, 2nd, 3rd, etc order terms. We do this in the hope of getting structures similar to linear filtering results, not only for their simplicity, but also to aid in comparison. These ideas also extend to other parameterizations: parameters affecting the same order should be grouped together.

3.6 Summary

In this chapter we have specialized the results of Chapter 2 to probability density functions. This resulted in F and $\underline{g}$ terms that could be computed in terms of moments of either the approximating distribution or some other distribution based on the error criterion used for the approximation. The two corresponding criteria are the square root and the logarithmic error criterion. We found that the F matrix is completely determined by the parameterization, $\hat{p}(x;\underline{\alpha}(t))$, and is independent of the model components, i.e., $\mathbf{f}, Q, h$. However, $\underline{g}$ depends on all of the model components.

error criterion	$\hat{p}(x;\underline{\alpha}(t))$	F_{ij}	g_i	expectation over
square root	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\}$	$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}$
log	$e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\}$	$E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\}$	$N(0,b)$

error criterion	$\hat{\varsigma}_t$
square root	$-\sum_i(\frac{1}{2}\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(2\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}) + \frac{1}{2}h^T(z - \frac{1}{2}h)$
log	$-\sum_i(\mathbf{f}_{i_{x_i}} + \varsigma_{x_i}\mathbf{f}_i) + \frac{1}{2}\sum_{i,j} Q_{ij}(\varsigma_{x_i}\varsigma_{x_j} + \varsigma_{x_i x_j}) + h^T(z - \frac{1}{2}h)$

Figure 3.6.4: F and $\underline{g}$ summary.

A summary is found in Table 3.6.4. Finally, application issues concerning size, integrations, closure, and ordering are considered. These issues direct us to:

- Keep the number of parameters in our approximation to a minimum
- Always try to interpret integrations as expectations of known pdf's
- Order the parameters to reflect groupings according to order

In the following chapter we will apply these results to the case where $\hat{p}(x;\underline{\alpha}(t))$ is second order or Gaussian. This will serve to validate our approximation scheme as well as to clarify its application. Later chapters will extend applications of our scheme to multi-modal $\hat{p}(x;\underline{\alpha}(t))$'s, including Gaussian sums and quartic exponentials.

Chapter 4

Applications: Linear Dynamics/Gaussian Parameterization

4.1 Introduction

In this chapter we will apply the results of Chapters 2 and 3 to the linear filtering problem. The motivation is twofold. First we need some way of verifying that the approximating schemes derived yield sensible results. A comparison with the well known results of linear filtering, namely the Kalman Filter (KF), would give us such a verification. Second, it allows us to go through a simple example of applying our approximation scheme to a specific problem. Section 2 will derive the equations for both square root and logarithmic formulations. Section 3 will comment on their agreement with existing linear filtering results.

4.2 Equations

In this section we shall go through the steps for deriving our filter for a linear state model. To further clarify the approach we will be taking, we shall label the key steps as we go along. Summarizing, the approach will be

1. Write down the state model

2. Select a error criterion/formulation

3. Choose the parameterization

4. Evaluate the F matrix

5. Evaluate the $\underline{g}$ vector

6. Solve for $\dot{\underline{\alpha}}(t)$

In this chapter there will actually be one additional step. To simplify comparisons, the final results will be put in terms of the mean and variance of the conditional pdf. The actual parameterization, however, will be chosen with regard to computational simplicity under a given formulation. Hence the conversion to mean and variance will be made after solving the $F\dot{\underline{\alpha}} = \underline{g}$ equation for $\dot{\underline{\alpha}}$. In addition, in order to make good our assumption that the pdf's could be expressed as e^{ς}, we will absorb any normalizing constants into the exponent ς. We will derive the $\dot{\underline{\alpha}}$ equations for both square root and logarithmic formulations.

Write down the state model

We will start with the following scalar linear model

$$dx = \mathrm{F}x dt + dv \tag{4.2.1}$$

$$z = hx + dw \tag{4.2.2}$$

for simplicity and clarity, where dv and dw are Brownian processes with intensity Q and 1 respectively.

Later we shall present the vector case.

Select a formulation

We begin with the square root formulation so that $\hat{\varsigma}_t$ (from Figure 3.3.2 but scalar, linear) is given by

$$\hat{\varsigma}_t = -(\frac{1}{2}\mathrm{F} + \hat{\varsigma}_x \mathrm{F}x) + \frac{1}{2}Q(2\hat{\varsigma}_x^2 + \hat{\varsigma}_{xx}) + \frac{1}{2}h(zx - \frac{1}{2}hx^2). \tag{4.2.3}$$

Choose a parameterization

Now with our approximating pdf being Gaussian, say with mean m and variance P, we have

$$e^{2\hat{\varsigma}(x;\underline{\alpha}(t))} \sim N(m,P) \rightarrow \hat{\varsigma}(x;\underline{\alpha}(t)) = -\frac{1}{4}\frac{(x-m)^2}{P} + \frac{c}{2} \tag{4.2.4}$$

where $c = -\frac{1}{2}(ln(P) + ln(2\pi))$. If we were to choose as our parameters some variation of

$$\underline{\dot{\alpha}} = \begin{pmatrix} c \\ m \\ P \end{pmatrix}, \tag{4.2.5}$$

we see that computing $\frac{\partial \hat{\varsigma}}{\partial P}$ can be further simplified by using $P^{-1} = V$ as the last parameter. An advantage to this parameterization is that under the square root formulation, all terms that are composed of centered moments, $(x-m)^n$, evaluate to zero for odd n. These characteristics make this parameterization a good choice. Summarizing,

$$\hat{p}(x;\underline{\alpha}(t)) \sim N(m, V^{-1}), \quad \underline{\dot{\alpha}} = \begin{pmatrix} c \\ m \\ V \end{pmatrix}. \tag{4.2.6}$$

Evaluate F

Computing F requires derivatives of the pdf exponent with respect to each of the parameters. The $\underline{g}$ term in addition requires the first and second derivatives of the exponent with respect to x. These are computed below.

$$\hat{\varsigma} = -\frac{1}{4}V(x-m)^2 + \frac{c}{2} \tag{4.2.7}$$

$$\hat{\varsigma}_c = \frac{1}{2} \tag{4.2.8}$$

$$\hat{\varsigma}_m = \frac{1}{2}V(x-m) \tag{4.2.9}$$

$$\hat{\varsigma}_V = -\frac{1}{4}(x-m)^2 \tag{4.2.10}$$

$$\hat{\varsigma}_x = -\frac{1}{2}V(x-m) \tag{4.2.11}$$

$$\hat{\varsigma}_{xx} = -\frac{V}{2} \tag{4.2.12}$$

Computing the various expectations of products of the above terms for the F matrix yields

$$F = E_{N(m,V^{-1})}\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\} = \begin{pmatrix} \frac{1}{4} & 0 & -\frac{1}{8}V^{-1} \\ 0 & \frac{1}{4}V & 0 \\ -\frac{1}{8}V^{-1} & 0 & \frac{3}{16}V^{-2} \end{pmatrix}. \qquad (4.2.13)$$

Evaluate $\underline{g}$

The $\underline{g}$ term, after some effort, is

$$g = E_{N(m,V^{-1})}\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\} = \begin{pmatrix} 0 \\ \frac{1}{4}V\mathrm{F}m \\ -\frac{1}{4}V^{-1}\mathrm{F} - \frac{1}{8}Q \end{pmatrix} + \begin{pmatrix} \frac{1}{4}h(zm - \frac{1}{2}h(P+m^2)) \\ \frac{1}{4}h(z-hm) \\ -\frac{1}{8}hP(zm - \frac{1}{2}h(3P+m^2)) \end{pmatrix}. \qquad (4.2.14)$$

Solve for $\underline{\dot{\alpha}}(t)$

Inverting F

$$F^{-1} = \begin{pmatrix} 6 & 0 & 4V \\ 0 & 4V^{-1} & 0 \\ 4V & 0 & 8V^2 \end{pmatrix} \qquad (4.2.15)$$

and solving for $\underline{\dot{\alpha}}$ gives

$$\underline{\dot{\alpha}} = \begin{pmatrix} \dot{c} \\ \dot{m} \\ \dot{V} \end{pmatrix} = \begin{pmatrix} -\mathrm{F} - \frac{1}{2}QV \\ \mathrm{F}m \\ -2V\mathrm{F} - QV^2 \end{pmatrix} + \begin{pmatrix} hm(z - \frac{1}{2}hm) \\ hP(z-hm) \\ h^2 \end{pmatrix} \qquad (4.2.16)$$

which once put in terms of c, m, P (using $P^{-1} = V$) yield the familiar Kalman filter equations.

$$\begin{aligned} \dot{c} &= -\frac{1}{2P}(2P\mathrm{F} + Q) + hm(z - \frac{1}{2}hm) \\ \dot{m} &= \mathrm{F}m + hP(z-hm) \\ \dot{P} &= 2P\mathrm{F} + Q - P^2h^2 \end{aligned} \qquad (4.2.17)$$

(Note that the 'R' term is missing, as $R = 1$ in Equation 4.2.2.)

Before we start any comparisons, we will go through the same steps for to find the solution to $F\underline{\dot{\alpha}} = \underline{g}$ for the logarithmic formulation.

Select a formulation

For this formulation

$$\hat{\varsigma}_t = -(\mathrm{F} + \hat{\varsigma}_x \mathrm{F} x) + (1/2)Q(\hat{\varsigma}_x^2 + \hat{\varsigma}_{xx}) + h(zx - \frac{1}{2}hx^2). \tag{4.2.18}$$

Choose a parameterization

In the previous parameterization the exponent consisted of a $(x - m)$ term. Under the square root formulation these terms had the advantage of being centered moments. However, under the logarithmic formulation they only complicate matters since the expectations will be over a zero mean pdf $(N(0, b))$. Therefore we will choose the following parameterization for its simplicity

$$\hat{p} = e^{2\hat{\varsigma}}, \quad \hat{\varsigma} = \alpha_0 + \alpha_1 x + \alpha_2 x^2. \tag{4.2.19}$$

There should be no change due to this choice of parameterization, since we can cast this form into a Gaussian pdf using a simple transformation on the parameters i.e.

$$\alpha_0 = c - \frac{m^2}{2P}, \quad \alpha_1 = \frac{m}{P}, \quad \alpha_2 = -\frac{1}{2P}. \tag{4.2.20}$$

These transformations will be applied after the equation for $\underline{\dot{\alpha}}$ has been obtained.

Again we need the derivatives of the exponent with respect to the parameters and x.

$$\hat{\varsigma} = \alpha_0 + \alpha_1 x + \alpha_2 x^2 \tag{4.2.21}$$

$$\hat{\varsigma}_{\alpha_0} = 1 \tag{4.2.22}$$

$$\hat{\varsigma}_{\alpha_1} = x \tag{4.2.23}$$

$$\hat{\varsigma}_{\alpha_2} = x^2 \tag{4.2.24}$$

$$\hat{\varsigma}_x = \alpha_1 + 2\alpha_2 x \tag{4.2.25}$$

$$\hat{\varsigma}_{xx} = 2\alpha_2 \tag{4.2.26}$$

Evaluate F

Computing the various expectations of products of the above terms for F yields

$$F = E_{N(0,b)}\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\} = \begin{pmatrix} 1 & 0 & b \\ 0 & b & 0 \\ b & 0 & 3b^2 \end{pmatrix} \tag{4.2.27}$$

with inverse:

$$F^{-1} = \begin{pmatrix} \frac{3}{2} & 0 & -\frac{1}{2b} \\ 0 & \frac{1}{b} & 0 \\ -\frac{1}{2b} & 0 & \frac{1}{2b^2} \end{pmatrix}. \tag{4.2.28}$$

Evaluate $\underline{g}$

The $\underline{g}$ term after some effort becomes

$$g = E_{N(0,b)}\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\} = \begin{pmatrix} -\mathrm{F} - 2\alpha_2 b\mathrm{F} + 2Q\alpha_2^2 b + \frac{Q}{2}(\alpha_1^2 + \alpha_2) \\ -\alpha_1 b\mathrm{F} + 2Q\alpha_1\alpha_2 b \\ -b\mathrm{F} - 6\alpha_2\mathrm{F}b^2 + 6Q\alpha_2^2 b^2 + \frac{Q}{2}(\alpha_1^2 + \alpha_2)b \end{pmatrix} + \begin{pmatrix} -\frac{1}{2}h^2 b \\ hbz \\ -\frac{1}{2}h^2 3b^2 \end{pmatrix}. \tag{4.2.29}$$

Note that g_0 is no longer equal to zero. This characteristic will be discussed in the next chapter.

Solving for $\underline{\dot{\alpha}}(t)$

Now solving for $\underline{\dot{\alpha}}$ gives

$$\underline{\dot{\alpha}} = \begin{pmatrix} -\mathrm{F} + \frac{Q}{2}(\alpha_1^2 + \alpha_2) \\ -\alpha_1\mathrm{F} + 2Q\alpha_1\alpha_2 \\ -2\alpha_2\mathrm{F} + 2Q\alpha_2^2 \end{pmatrix} + \begin{pmatrix} 0 \\ hz \\ -\frac{1}{2}h^2 \end{pmatrix} \tag{4.2.30}$$

which once put in terms of c, m, P again (using Equations 4.2.20), yields the familiar Kalman filter equations.

$$\begin{aligned} \dot{c} &= -\frac{1}{2P}(2P\mathrm{F} + Q) + hm(z - \frac{1}{2}hm) \\ \dot{m} &= \mathrm{F}m + hP(z - hm) \\ \dot{P} &= 2P\mathrm{F} + Q - P^2h^2 \end{aligned} \tag{4.2.31}$$

Note that since we have considered the normalizing constant as a separate parameter, $\dot{c} \neq 0$ to account for its dependence on P.

When we work out the vector versions of the above equations, we find a tremendous amount of bookkeeping is necessary. This is due to the reordering of the covariance matrix into a vector form as part of $\underline{\alpha}$. Very careful attention also has to be paid to handling the symmetric property of the covariance matrix. After all these precautions are taken, we solve for $\underline{\alpha}$ to get

$$\begin{aligned} \dot{\underline{m}} &= \mathbf{F}\underline{m} + PH^T(\underline{z} - H\underline{m}) \qquad (4.2.32) \\ \dot{P} &= \mathbf{F}P + P\mathbf{F}^T + Q - PH^THP \qquad (4.2.33) \end{aligned}$$

(where $\underline{m} \in R^n$ and $\mathbf{F}, P, Q \in R^{n\times n}$) for both square root and logarithmic formulations as expected.

4.3 Comparisons

Its clear from Equations 4.2.17 and 4.2.31 that our projection scheme yields results consistent with traditional linear filtering techniques. Since the same procedure will be used for higher order parameterizations and dynamics, this is a good indication that the results we obtain for higher order densities will also be consistent. In fact, we will show in the following chapter that our approach will correctly determine the higher order terms when we consider Gaussian parameterizations and nonlinear dynamics. Before we do this there are a few things to be considered in the derivations. First note that by pursuing centered moments the F matrix becomes 50 percent sparse. Second, the structure is band symmetric. These two characteristics simplify solutions to the $F\dot{\underline{\alpha}} = g$ equation. Note that the logarithmic formulation retains this characteristic even though the exponent is in a general form. This will become important when we study higher order densities of Chapter 7.

Chapter 5

Application: Nonlinear Dynamics/ Gaussian Parameterization

5.1 Introduction

In the previous chapter we applied our approximation scheme to the class of linear filtering problems and found our results to be consistent with the well known results of linear filtering. In this chapter we will apply our projection scheme to a nonlinear filtering problem using the same second order parameterizations as in Chapter 4.

In Section 2 we will derive the general equations for the scalar nonlinear filter. In Section 3, we apply these results, as well as existing nonlinear filtering results, to a particular nonlinear dynamical model for comparison. Section 4 will describe the portion of our approximation due to the Zakai continuous measurement term, again, for a particular measurement equation to ease comparisons. In Section 5, we outline the corresponding results for the vector case. Finally, Section 6 considers a scalar and vector example which shows some of the advantages and disadvantages of our approximation scheme.

5.2 Equations

The first step in solving $F\underline{\dot{\alpha}} = \underline{g}$ is to find F. However, since we will be using the same parameterizations as in Chapter 4, we may use the same F's found there.

However, the $\underline{g}$ term will have to be recalculated since it depends on the dynamics $\mathbf{f}(x)$. Initially, we will leave $\mathbf{f}(x)$ unspecified; later, in order to simplify comparisons, we will specialize to a particular $\mathbf{f}(x)$ which should be sufficiently general to excite most of the modes of the filter. The example we shall use will be:

$$\mathbf{f}(x) = f_0 + f_1 x + f_2 x^2 + f_3 x^3 + f_4 x^4. \tag{5.2.1}$$

We will first apply the approximation scheme using the square root formulation. Using the equations in Figure 3.4.3 for $\underline{g}$ we find:

$$\underline{g} = \begin{pmatrix} \frac{1}{2} V E\{(x-m)\mathbf{f}(x)\} - \frac{1}{2} E\{\mathbf{f}_x(x)\} \\ \frac{1}{4} V^2 E\{(x-m)^2 \mathbf{f}(x)\} - \frac{1}{4} V E\{(x-m)\mathbf{f}_x(x)\} \\ -\frac{1}{8} V E\{(x-m)^3 \mathbf{f}(x)\} + \frac{1}{8} E\{(x-m)^2 \mathbf{f}_x(x)\} - \frac{1}{8} Q \end{pmatrix}, \tag{5.2.2}$$

where the expectations are over $N(m, V^{-1})$ and $\mathbf{f}_x = \frac{\partial \mathbf{f}}{\partial x}$. In Chapter 4, we found that g_0, the first element of $\underline{g}$, was equal to zero. This turns out also to be true here. We can see how this happens by re-examining the definition of g_0 under the square root formulation:

$$g_0 = < \varsigma_{\alpha_0}, \varsigma_t > = \int \varsigma_{\alpha_0} \varsigma_t q^2 dx = \int \varsigma_{\alpha_0} q \varsigma_t q dx = \int \varsigma_{\alpha_0} q q_t dx. \tag{5.2.3}$$

As before $\varsigma_{\alpha_0} = 1/2 = \text{constant}$, so that

$$g_0 = (\text{some constant}) \int q q_t dx = \frac{1}{2} \frac{\partial}{\partial t} \int q^2 dx = \frac{1}{2} \frac{\partial}{\partial t} \int p dx = \frac{1}{2} \frac{\partial 1}{\partial t} = 0. \tag{5.2.4}$$

The key points here are that the integral of the pdf, p, is unity and that under the square root formulation the pdf of interest appears alone in the integral. In general, we will be dealing with un-normalized pdf's and other weighting functions, so $g_0 \neq 0$. However, since we are dealing here with Gaussians and the square root formulation, $g_0 = 0$ is guaranteed.

Proceeding to solve for $\underline{\dot{\alpha}}$ with $g_0 = 0$ and using F^{-1} from Chapter 4 (Equation 4.2.15) for the square root formulation, we have

$$\underline{\dot{\alpha}} = \begin{pmatrix} \dot{c} \\ \dot{m} \\ \dot{V} \end{pmatrix} = \begin{pmatrix} E\{-\frac{1}{2} V^2 (x-m)^3 \mathbf{f}(x)\} + E\{\frac{1}{2} V (x-m)^2 \mathbf{f}_x(x)\} - \frac{1}{2} V Q \\ V E\{(x-m)^2 \mathbf{f}(x)\} - E\{(x-m)\mathbf{f}_x(x)\} \\ E\{-V^3 (x-m)^3 \mathbf{f}(x)\} + E\{V^2 (x-m)^2 \mathbf{f}_x(x)\} - V^2 Q \end{pmatrix}. \tag{5.2.5}$$

When we translate this into means and variances we get

$$\begin{aligned}
\dot{c} &= -\frac{1}{2P}E\{(P^{-1}(x-m)\mathbf{f}(x) - \mathbf{f}_x(x))(x-m)^2\} - \frac{1}{2P}Q \\
\dot{m} &= E\{(P^{-1}(x-m)\mathbf{f}(x) - \mathbf{f}_x(x))(x-m)\} \\
\dot{P} &= E\{(P^{-1}(x-m)\mathbf{f}(x) - \mathbf{f}_x(x))(x-m)^2\} + Q. \qquad (5.2.6)
\end{aligned}$$

Before we go on to analyze this result, we will also derive the mean and variance equations resulting from our approximation scheme under the logarithmic formulation. Going through similar steps with $\mathbf{f}(x)$, as yet undefined, we find

$$\underline{g} = \begin{pmatrix} -E\{\mathbf{f}_x(x) + (2\alpha_2 x + \alpha_1)\mathbf{f}(x)\} + 2Q\alpha_2^2 b + \frac{Q}{2}(\alpha_1^2 + \alpha_2) \\ -E\{x\mathbf{f}_x(x) + (2\alpha_2 x^2 + \alpha_1 x)\mathbf{f}(x)\} + 2Q\alpha_1\alpha_2 b \\ -E\{x^2\mathbf{f}_x(x) + (2\alpha_2 x^3 + \alpha_1 x^2)\mathbf{f}(x)\} + 6Q\alpha_2^2 b^2 + \frac{Q}{2}(\alpha_1^2 + \alpha_2)b \end{pmatrix} \qquad (5.2.7)$$

where expectations are over $N(0,b)$, with b left unspecified. Applying F^{-1} for the logarithmic formulation (Equation 4.2.28) to $\underline{g}$ and expressing this in terms of c, m, P we find

$$\begin{aligned}
\dot{c} &= -\frac{1}{2P}\frac{P^2}{b}E\{((x-m)V\mathbf{f}(x) - \mathbf{f}_x(x))(\frac{x^2}{b} - 3)\} - \frac{1}{2P}Q \\
\dot{m} &= \frac{P}{b}E\{((x-m)V\mathbf{f}(x) - \mathbf{f}_x(x))(x + m(\frac{x^2}{b} - 1))\} \\
\dot{P} &= \frac{P^2}{b}E\{((x-m)V\mathbf{f}(x) - \mathbf{f}_x(x))(\frac{x^2}{b} - 1)\} + Q. \qquad (5.2.8)
\end{aligned}$$

Given Equations 5.2.6 and 5.2.8 for mean and variance under the two formulations, we are ready to compare their performance. However, in their present form it is difficult to gain any insight into the operation of these filters. To gain further insight into the implications of these results, we will now pick the $\mathbf{f}(x)$ of Equation 5.2.1 and make comparisons between the resulting filter, the EKF [4] and the second order [2] filters. Using the $\mathbf{f}(x)$ of Equation 5.2.1 in Equations 5.2.6 and 5.2.8, and evaluating the expectations yields the following set of equations: (for comparison, the EKF and second order filter results for this same $\mathbf{f}(x)$ are also presented)

- Square root formulation

$$\dot{m} = f_0 + f_1 m + f_2(P + m^2) + f_3(3Pm + m^3)$$

$$+f_4(3P^2 + 6Pm^2 + m^4) \tag{5.2.9}$$

$$= E\{\mathbf{f}(x)\} \tag{5.2.10}$$

$$\dot{P} = f_1 2P + f_2 4Pm + f_3(6P^2 + 6Pm^2) + f_4(36P^2m + 8Pm^3)$$

$$+Q \tag{5.2.11}$$

- Logarithmic formulation

$$\dot{m} = f_0 + f_1 m + f_2(3b - 2P - 2m^2) + f_3(9bm - 6Pm)$$

$$+f_4(-12Pb + 15b^2 - 12bm^2) \tag{5.2.12}$$

$$\dot{P} = f_1 2P + f_2(-2Pm) + f_3(12Pb - 6P^2) + f_4(-12Pmb)$$

$$+Q \tag{5.2.13}$$

- EKF

$$\dot{m} = f_0 + f_1 m + f_2 m^2 + f_3 m^3 + f_4 m^4 \tag{5.2.14}$$

$$= \mathbf{f}(m) \tag{5.2.15}$$

$$\dot{P} = f_1 2P + f_2 4Pm + f_3 6Pm^2 + f_4 8Pm^3 + Q \tag{5.2.16}$$

- Second order filter

$$\dot{m} = f_0 + f_1 m + f_2(P + m^2) + f_3(3Pm + m^3)$$

$$+f_4(6Pm^2 + m^4) \tag{5.2.17}$$

$$= \mathbf{f}(m) + E\{(x - m)\frac{\partial \mathbf{f}(x)}{\partial x}|_{x=m}\} \tag{5.2.18}$$

$$\dot{P} = f_1 2P + f_2 4Pm + f_3 6Pm^2 + f_4 8Pm^3 + Q \tag{5.2.19}$$

5.3 Comparison

Note that our scheme has many more terms than either the EKF or second order filter. Also, note that the second order filter has more terms in common with our square root filter than the EKF. As it turns out, this is the general trend. The second order filter is obtained by taking one more term in the Taylor series expansion of $\mathbf{f}(x)$ than the EKF. As we take more terms, we will match more of the terms of our approximation. As seen in Equations 5.2.10, 5.2.15, and 5.2.18 , our filter corresponds to taking the expectation of $\mathbf{f}(x)$ exactly, with no approximation. These observations suggest that our scheme does in fact yield the best approximation for a given parameterization.

Before we proceed to some examples, we shall complete the description of the solution by describing the measurement terms and the vector versions of our approximation.

5.4 Measurement Terms

For completeness, in this section we will describe the effects on the additional continuous measurement term of the scalar Zakai equation, which we restate here as:

$$\mathbf{h}(x)(z - \frac{1}{2}\mathbf{h}(x)). \tag{5.4.1}$$

The only modification to our filter will be the addition of the effects of this term on $\underline{g}$, and we will therefore show only this term in our derivations. As was the case for the previous results with $\mathbf{f}(x)$ unspecified, leaving $\mathbf{h}(x)$ unspecified leads to little insight into filter operation. Therefore, we will choose a sample $\mathbf{h}(x)$ similar to $\mathbf{f}(x)$ to ease comparisons:

$$\mathbf{h}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 \tag{5.4.2}$$

For completeness, we show, below, the results for square root and logarithmic formulations with $\mathbf{h}(x)$ unspecified:

error criterion	addition to $\dot{m}$	addition to $\dot{P}$
$\sqrt{\ }$	$E\{(x-m)\mathbf{h}(x)(z-\frac{1}{2}\mathbf{h}(x))\}$	$-E\{(P-(x-m)^2)\mathbf{h}(x)(z-\frac{1}{2}\mathbf{h}(x))\}$
log	$\frac{P}{b}E\{(\frac{mx^2}{b}+x-m)\mathbf{h}(x)(z-\frac{1}{2}\mathbf{h}(x))\}$	$\frac{P^2}{b}E\{(\frac{x^2}{b}-1)\mathbf{h}(x)(z-\frac{1}{2}\mathbf{h}(x))\}$

In these results the expected values are over $N(m, V^{-1})$ and $N(0, b)$ for square root and logarithmic formulations, respectively. Using Equation 5.4.2 in the above expressions yields many additional terms for our scheme as well as for the EKF and second order filters; hence we relegate their complete display to this chapter's appendix and show only square root and EKF mean equations here.

- Square root formulation

$$\begin{aligned}\dot{m} \;=\;& ((12\beta_4 m + 3\beta_3)P^2 + (4\beta_4 m^3 + 3\beta_3 m^2 + 2\beta_2 m + \beta_1)P)z \\ & -(420\beta_4^2 m + 105\beta_3\beta_4)P^4 \\ & -(420\beta_4^2 m^3 + 315\beta_3\beta_4 m^2 + (90\beta_2\beta_4 + 45\beta_3^2)m + 15\beta_1\beta_4 + 15\beta_2\beta_3)P^3 \\ & -(84\beta_4^2 m^5 + 105\beta_3\beta_4 m^4 + (60\beta_2\beta_4 + 30\beta_3^2)m^3 + (30\beta_1\beta_4 + 30\beta_2\beta_3)m^2 \\ & \quad +(12\beta_0\beta_4 + 12\beta_1\beta_3 + 6\beta_2^2)m + 3\beta_0\beta_3 + 3\beta_1\beta_2)P^2 \\ & -(4\beta_4^2 m^7 + 7\beta_3\beta_4 m^6 + (6\beta_2\beta_4 + 3\beta_3^2)m^5 + (5\beta_1\beta_4 + 5\beta_2\beta_3)m^4 \\ & \quad +(4\beta_0\beta_4 + 4\beta_1\beta_3 + 2\beta_2^2)m^3 + (3\beta_0\beta_3 + 3\beta_1\beta_2)m^2 \\ & \quad +(2\beta_0\beta_2 + \beta_1^2)m + \beta_0\beta_1)P \end{aligned} \tag{5.4.3}$$

- EKF

$$\begin{aligned}\dot{m} \;=\;& (4\beta_4 m^3 + 3\beta_3 m^2 + 2\beta_2 m + \beta_1)Pz \\ & -(4\beta_4^2 m^7 + 7\beta_3\beta_4 m^6 + (6\beta_2\beta_4 + 3\beta_3^2)m^5 + (5\beta_1\beta_4 + 5\beta_2\beta_3)m^4 \\ & \quad +(4\beta_0\beta_4 + 4\beta_1\beta_3 + 2\beta_2^2)m^3 + (3\beta_0\beta_3 + 3\beta_1\beta_2)m^2 + (2\beta_0\beta_2 + \beta_1^2)m \\ & \quad +\beta_0\beta_1)P \end{aligned} \tag{5.4.4}$$

Summarizing, we find, as in the case of $\mathbf{f}(x)$, that the differences between the EKF, second order, logarithmic, and square root filters for $\mathbf{h}(x)$ are reflected in additional terms. Scrutinizing these equations carefully reveals that, with the exception of the logarithmic filter, each more complicated filter contains the terms of the lesser filters. This, again, indicates that our filtering scheme does lead us in the right direction since EKF and second order filters appear as a subset of our filter. Although the logarithmic filter contains many of the same terms as the square root filter, some of the terms are lost. The reason for this is the choice of inner product which causes P to be replaced by b, and m to be replaced by 0 in several terms, thus modifying and cancelling many of the terms. What we gain from the lost terms is a reduction in the complexity of our filter to a complexity close to that of the EKF. What is lost in the simplification is intuition. The terms in the expressions for $\dot{m}$ and $\dot{P}$ under the square root formulation can be easily identified with the EKF results, while the same expressions under the logarithmic formulation have few similarities.

For reasons of clarity and the practical advantages of discrete measurements over continuous, we will not carry the description of these continuous measurement terms into the vector case or into the higher order filters disscused in later chapters. They were used here mainly to further support the reasonableness of our filter.

5.5 Vector Case

For the vector case, we must again pay great attention to the ordering of the parameters (mean, covariance). Other than this, the calculations themselves are time consuming and unenlightening, so only the results are presented here.

For the square root formulation we find

$$\dot{\underline{m}} = E\{(\underline{x}-\underline{m})(\underline{x}-\underline{m})^T V\mathbf{f}(\underline{x}) - \underline{\mathbf{1}}^T\acute{\mathbf{f}}(\underline{x})\} \tag{5.5.1}$$

$$\dot{P} = E\{(\underline{x}-\underline{m})(\underline{x}-\underline{m})^T(\underline{x}-\underline{m})^T V\mathbf{f}(\underline{x}) - \underline{\mathbf{1}}^T\acute{\mathbf{f}}(\underline{x})\} + Q \tag{5.5.2}$$

$$\text{where} \quad \underline{\mathbf{1}} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \quad \text{and } \acute{\mathbf{f}}(\underline{x}) = \begin{pmatrix} \frac{\partial \mathbf{f}_1}{\partial x_1} \\ \frac{\partial \mathbf{f}_2}{\partial x_2} \\ \vdots \\ \frac{\partial \mathbf{f}_n}{\partial x_n} \end{pmatrix}. \tag{5.5.3}$$

Similarly, for the logarithmic formulation:

$$\dot{m} = E\{(\underline{x} + (b^{-1}\underline{x}\underline{x}^T - I)\underline{m})(\underline{x}-\underline{m})^T V\mathbf{f}(\underline{x}) - \underline{\mathbf{1}}^T\acute{\mathbf{f}}(\underline{x})\}b^{-1}P \tag{5.5.4}$$

$$\dot{P} = PE\{(b^{-1}\underline{x}\underline{x}^T - I)(\underline{x}-\underline{m})^T V\mathbf{f}(\underline{x}) - \underline{\mathbf{1}}^T\acute{\mathbf{f}}(\underline{x})\}b^{-1}P + Q. \tag{5.5.5}$$

A comparison of these equations with their scalar counterparts make them appear consistent with our expectations. The only anomaly is the $\underline{\mathbf{1}}^T\acute{\mathbf{f}}$ terms of Equation 5.5.1, 5.5.2, 5.5.4, and 5.5.5. It appears as though the role of $\underline{1}^T \acute{\mathbf{f}}$ is to correct for the symmetrical property of the covariance, P, which is lost when we cast P into the vector form $\underline{\alpha}$.

This effect can be seen more clearly when we look at a particular example. Here, we outline the derivation for the linear case under the square root formulation for $\underline{\mathbf{f}}(\underline{x}) = \mathbf{F}\underline{x}$:

$$\dot{\underline{m}} = \mathbf{F}\underline{m} \tag{5.5.6}$$

$$\dot{P} = Ptrace(\mathbf{F}) + \mathbf{F}P + P\mathbf{F}^T \underbrace{-Ptrace(\mathbf{F})}_{\text{due to } \mathbf{1}^T \acute{\mathbf{f}}} + Q. \tag{5.5.7}$$

Note that the two $trace(\mathbf{F})$ terms cancel out, thus confirming our suspicions.

Now we are ready to look at some examples. The first will involve a scalar process which will use the results of the previous two sections; the second, a vector process, uses the results of this section.

5.6 Examples

Scalar Example

For this example we will plot the mean and variance of our approximating conditional pdf for the following model:

$$dx = -x^3 dt + dv, \tag{5.6.1}$$

where dv is a Brownian process with intensity Q. Then, we will study the same dynamics with continuous measurements, z, obtained from the following model:

$$dz = \beta x^3 dt + dw, \tag{5.6.2}$$

where dw is a Brownian process with intensity 1.

The results are plotted in Figure 5.6.1 as $m + \sqrt{P}$ and $m - \sqrt{P}$ to display the evoluton of both mean and variance versus time. for the case of no measurements with $m(0) = 0, P(0) = 0, Q = 3, b = 1$ (logarithmic formulation).

filter	mean	variance
$\sqrt{\ }$	$\dot{m} = -(3Pm + m^3)$	$\dot{P} = -(6P^2 + 6Pm^2) + Q$
log	$\dot{m} = -(9bm - 6Pm)$	$\dot{P} = -(12Pb - 6P^2) + Q$
2nd order	$\dot{m} = -(3Pm + m^3)$	$\dot{P} = -6Pm^2 + Q$
EKF	$\dot{m} = -m^3$	$\dot{P} = -6Pm^2 + Q$

Table 5.1: $\mathbf{f}(x) = -x^3$ approximating mean and variance equations

$\dot{m}+$	$\dot{P}+$
$3\beta(P + m^2)Pz - 3\beta^2 Pm(15P^2 + 10Pm^2 + m^4)$	$6\beta P^2 mz - 15P^2\beta^2(3P^2 + 6Pm^2 + m^4)$
$3b\beta Pz - 45b^2\beta^2 Pm$	$-45b^2P^2\beta^2$
$3\beta m^2 Pz - 3\beta^2 Pm^3(3P + m^2)$	$6\beta mP^2 z - 3\beta^2 P^2 m^2(6P + 5m^2)$
$3\beta m^2 Pz - 3\beta^2 Pm^5$	$-9\beta^2 m^4 P^2$

Table 5.2: Modifications for continuous measurements

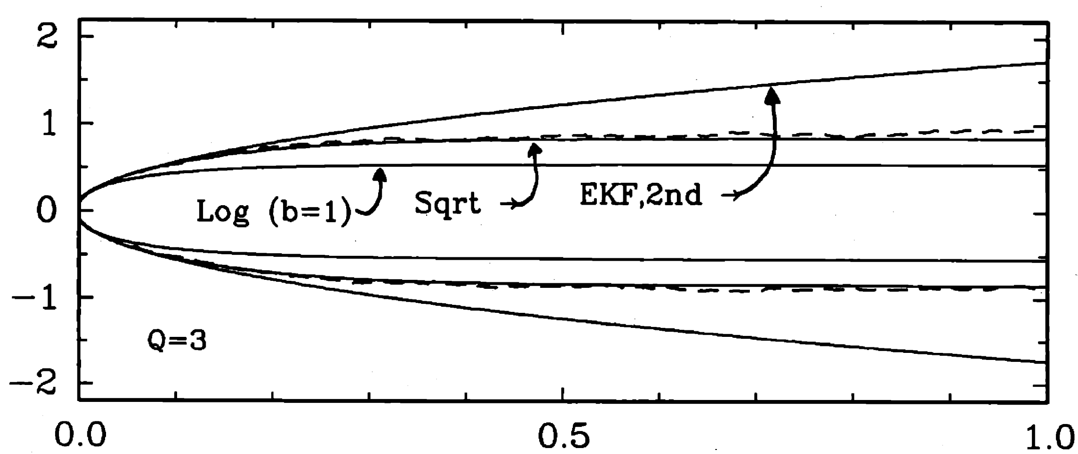


Figure 5.6.1: Plot of $m \pm \sqrt{P}$ for $dx = -x^3dt + dv$ for various filters

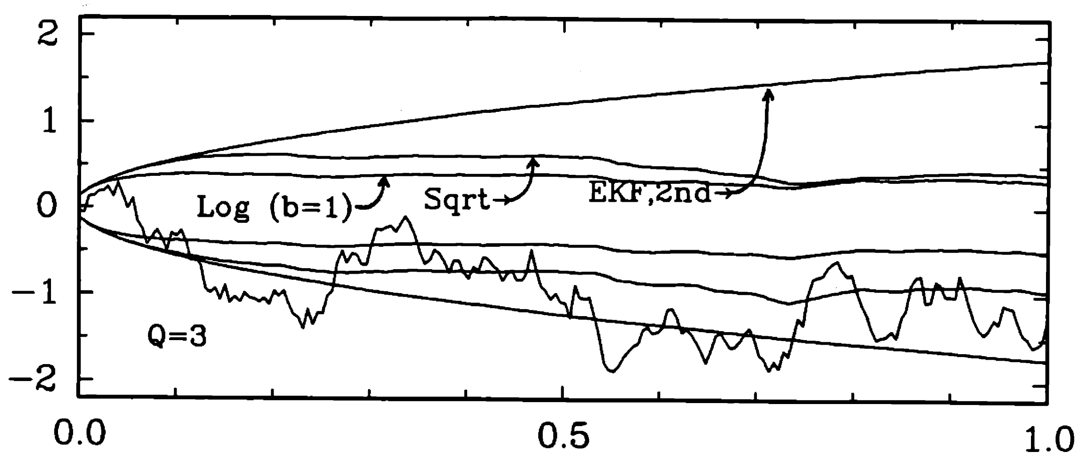


Figure 5.6.2: Plot of $m \pm \sqrt{P}$ for $dx = -x^3dt + dv$, $dz = x^3dt + dw$

From the superlinear structure of Equation 5.6.1 we know, a priori, that the true process, x, should remain finite in variance over time. This is verified by the results of a Monte Carlo simulation which appear as the dashed lines in Figure 5.6.1. With this in mind, and refering to Figure 5.6.1, note that the EKF and second order filters lead to unbounded P while both our schemes remain bounded. In fact, the square root case results coincide exactly with the simulation results. The reason for this stability as well as the accuracy is the P^2 term in the $\dot{P}$ equations in Table 5.1 for the square root and logarithmic formulations. Similarly, for the case with measurements (Table 5.1+5.2), shown in Figure 5.6.2 in $m \pm \sqrt{P}$ format along with the true value of x for this run, we see that stability is maintained for our filters while lost in the EKF and second order filters, again due to the lack of the second order terms in the variance equation. Note also how the square root filter and, to a lesser extent, the logarithmic filter track the true path with greater accuracy. This can be seen as due to the lack of P^2 terms multiplying the measurement, z, in the $\dot{m}$ equations in Table 5.2.

Although this simple example does not guarantee that the extra terms will always yield desirable results for all parameterizations, it does once again suggest that our scheme does lead the parameters in the right direction. In the next subsection, we will present a vector example to further support the merits of our approximation

$$\underline{\dot{x}} = \begin{pmatrix} x_2 x_3 \\ -x_1 x_3 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ v \end{pmatrix} \quad v \sim N(0, q_n) \tag{5.6.3}$$

$$\underline{y} = H\underline{x} + \underline{w} \quad \underline{w} \sim N(0, R) \tag{5.6.4}$$

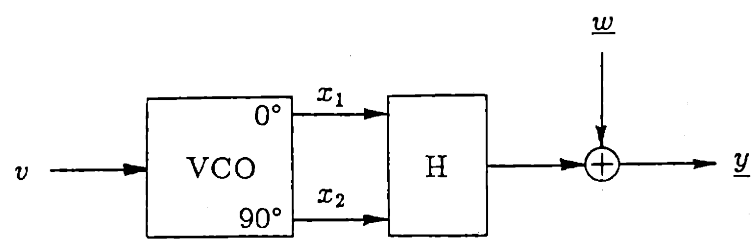


Figure 5.6.3: VCO/PLL model

scheme.

Phase Locked Loop(PLL) Example

In the following example, to get some further insight into our filter operation, we will study a vector example, which carries some physical interpretation (Figure 5.6.3).

Looking at the dynamical portion of this model alone, Equation 5.6.3, we find it represents a Voltage Controlled Oscillator(VCO) with quadrature outputs. The states are given by $x_1 = cos(x_3 t)$, $x_2 = sin(x_3 t)$, and x_3 = frequency in radians/sec, when v represents phase noise. When we include observations of x_1 or x_2, Equations 5.6.3 and 5.6.4 become a model for a Phase Locked Loop(PLL). This is precisely the systems considered by Gustafson [14] in his work in reducing the highly nonlinear classical PLL into one with linear measurements and 2nd order dynamical nonlinearities. The desirable features of this model, as an example, are the nonlinear, yet simple nature of $\mathbf{f}(\underline{x})$, and the solid physical interpretation of the states. These features will allow us to calculate and interpret the filters relatively easily.

We will apply the square root formulation to this problem for its intuitive advantages. Using Equation 5.5.2 for $\underline{m}$ and P under the square root formulation, and being careful with the ordering of the parameters, we get:

$$\dot{\underline{m}} = \underbrace{\underbrace{\begin{pmatrix} m_2 m_3 \\ -m_1 m_3 \\ 0 \end{pmatrix}}_{\text{EKF}} + \begin{pmatrix} P_{23} \\ -P_{13} \\ 0 \end{pmatrix}}_{\text{2nd order},\sqrt{\ }} \tag{5.6.5}$$

$$\dot{P} = \underbrace{\mathbf{F}P + P\mathbf{F}^T + Q}_{\text{EKF,2nd order},\sqrt{\ }} , \tag{5.6.6}$$

$$\text{where } \mathbf{F} = \begin{pmatrix} 0 & m_3 & m_2 \\ -m_3 & 0 & -m_1 \\ 0 & 0 & 0 \end{pmatrix} \text{ and } Q = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & q_n \end{pmatrix} . \tag{5.6.7}$$

We note, in this case that the second order filter is equivalent to our result, while the EKF lacks the P terms in its mean equation. This is due to the fact that $\mathbf{f}(x)$ is second order, and terms higher than second order in the Taylor series expansion evaluate to zero.

Before we simulate these equations, we shall try to generate some insight into the true, underlying pdf for this model. Since we know the model constrains x_1 and x_2 to be sinusoids in quadrature, we expect that the actual states should occupy points around a circle. This implies that all of the mass of the underlying pdf will also lie on the circle. A possible contour plot of the pdf in x_1, x_2 space is shown in Figure 5.6.4. If we add the third dimension of frequency, x_3, to the picture we can think of the pdf as moving along a helix in a Brownian fashion. So, for the case of no measurements (VCO), with the initial phase and frequency known, we would expect that as time went by the variance of our estimate would increase. We know, however, that the VCO can only produce sinusoids, so our uncertainty would cause the pdf to spread around the circle as shown in Figure 5.6.4. A plot of the mean of the pdf in x_1, x_2 space versus time is shown in Figure 5.6.5. With these pictures in mind, the results of the our parametric filter can be properly interpreted.

Clearly, we need more than just the mean and variance statistics in x_1, x_2 space to get a reasonable estimate of the sinusoid. However, since our filter produces

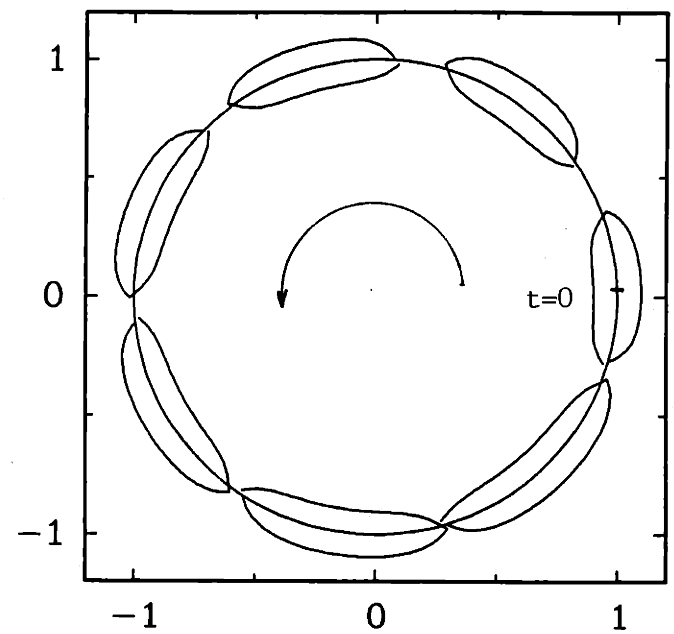


Figure 5.6.4: Contour plot of the underlying pdf for the VCO in x_1, x_2 space

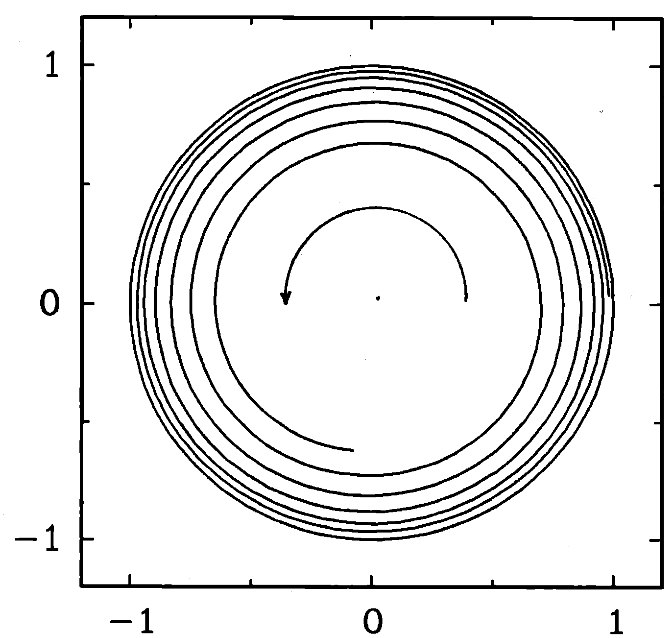


Figure 5.6.5: Underlying pdf mean for various times in x_1, x_2 space

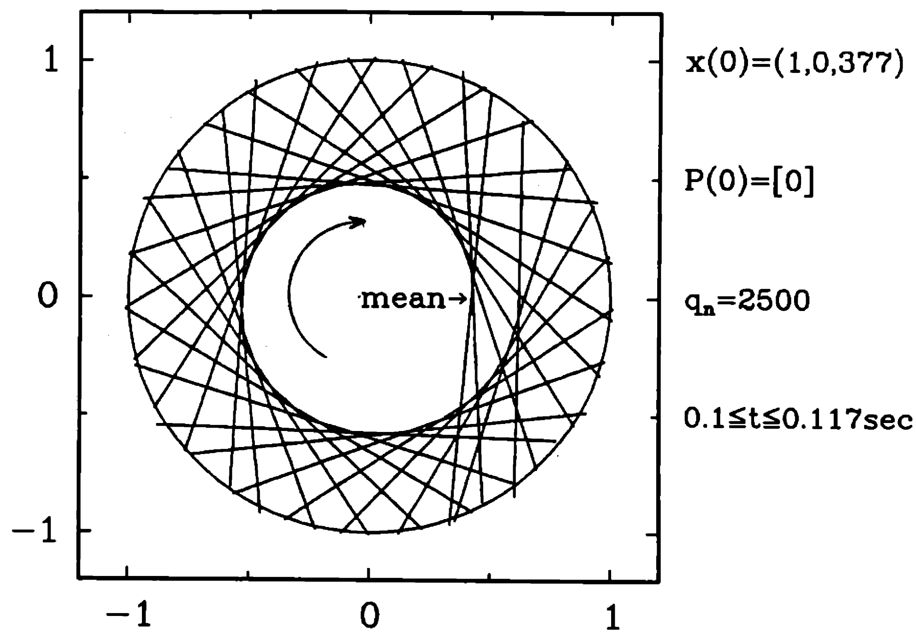


Figure 5.6.6: Contour plots of our approximating pdf at various times

only these, we shall attempt to gather some insight from these statistics alone. Plotting our results for mean and variance, as contour plots at various instants of time, we have Figure 5.6.6. Note the similarities of the first two statistics of this figure with the assumed underlying pdf of Figures 5.6.4 and 5.6.5, especially the mean. Studying the EKF results (Figure 5.6.7) for the same model, we find a very 'poor' match. The match is poor in the sense that the first and second moments are not estimated well. However, it is better than our filter in the sense that the sinusoidal structure of x_1, x_2 is retained, as shown in Figure 5.6.8. Clearly, the EKF solution is more 'sensible'. This amplitude modifying behavior of our filter is also characteristic of the results obtained in [14].

So, here we have an example that clearly approaches the 'correct' solution under one criterion (statistical), but does poorly in estimating the 'sensible' solution (sinusoidal). This behavior is due to the fact that second order statistics are not sufficient to completely describe the process. A parameterization which had as one of its parameters, the maximum point in the true pdf, for example, would most

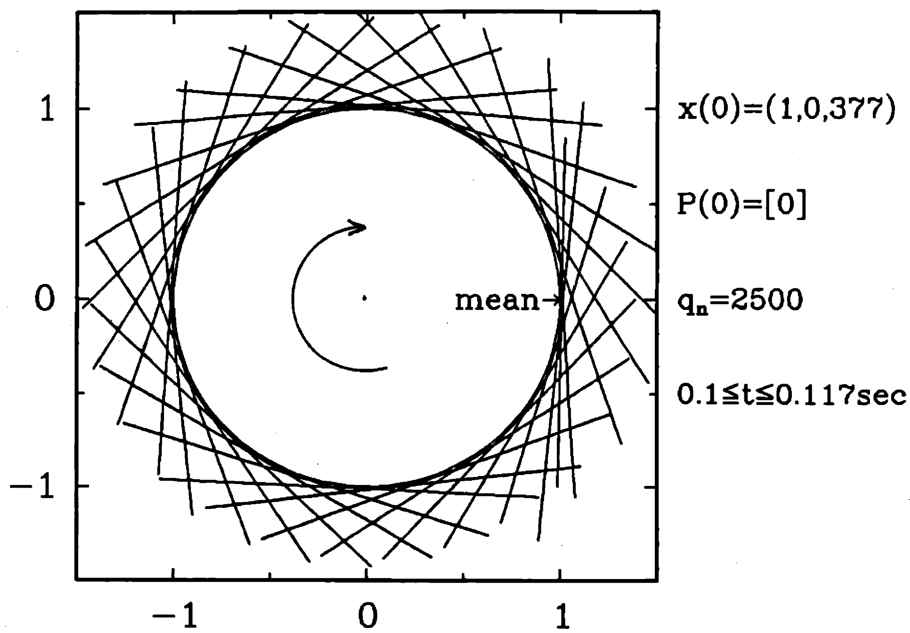


Figure 5.6.7: EKF contour plots

likely do better than any of these. This example demonstrates that the choice of parameterization is critical.

5.7 Summary

In this chapter we have applied our approximation scheme to the case where the dynamics, $\mathbf{f}(x)$, and measurements, $\mathbf{h}(x)$, are nonlinear, but the approximating pdf is second order. For comparison, we chose specific forms for $\mathbf{f}(x)$ and $\mathbf{h}(x)$ which lead to filtering equations similar to existing filters, namely the EKF and second order filter. However, our scheme resulted in additional terms in the equations which more accurately describe what the filter is suppose to do. Specifically, our filter determined the exact values for $E\{\dot{x}\}$ while the EKF and second order filters only approximate it. Continuous measurement terms were considered in Section 5.4 where we found results similar to that in Section 5.3 in that our filters included terms which described higher order effects in the measurement term, $\mathbf{h}(x)$. This

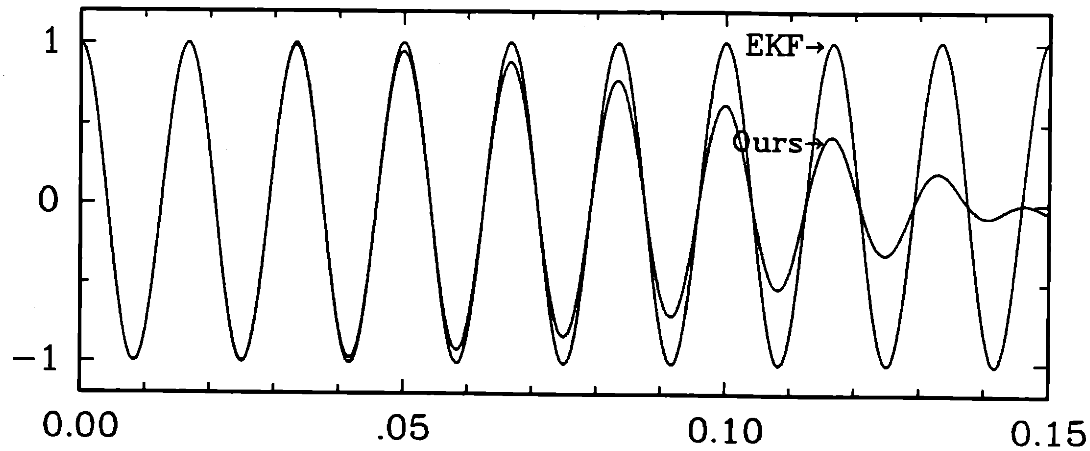


Figure 5.6.8: x_1 versus time for our filter and EKF

provided additional encouragement for our results thus far. In the next section, we displayed the vectorized versions of our filter and noted the similarities to their scalar counterparts. Finally, we examined a scalar example which showed that our square root and logarithmic filters outperform EKF and second order filters due to the extra terms considered in the variance equations. The same outcome was achieved over the EKF when we applied our square root filter to the vector VCO model. However, here we found that the statistically superior result was not necessarily the desired one, thus pointing to the importance of choosing a good parameterization. In the next chapter, we will demonstrate the generality of our scheme by applying it to a parameterization, which although popular cannot be expressed as an exponential. The approximating pdf there will consist of a weighted sum of Gaussian pdf's with the weights and individual means and variances comprising the parameters, $\underline{\alpha}$.

5.8 Appendix

Additional Zakai Measurement Terms

- Square root formulation

$$\begin{aligned}\dot{m} &= ((12\beta_4 m + 3\beta_3)P^2 + (4\beta_4 m^3 + 3\beta_3 m^2 + 2\beta_2 m + \beta_1)P)z \\ &\quad -(420\beta_4^2 m + 105\beta_3\beta_4)P^4\end{aligned}$$

$$
\begin{aligned}
& -(420\beta_4^2 m^3 + 315\beta_3\beta_4 m^2 + (90\beta_2\beta_4 + 45\beta_3^2)m + 15\beta_1\beta_4 + 15\beta_2\beta_3)P^3 \\
& -(84\beta_4^2 m^5 + 105\beta_3\beta_4 m^4 + (60\beta_2\beta_4 + 30\beta_3^2)m^3 + (30\beta_1\beta_4 + 30\beta_2\beta_3)m^2 \\
& \quad +(12\beta_0\beta_4 + 12\beta_1\beta_3 + 6\beta_2^2)m + 3\beta_0\beta_3 + 3\beta_1\beta_2)P^2 \\
& -(4\beta_4^2 m^7 + 7\beta_3\beta_4 m^6 + (6\beta_2\beta_4 + 3\beta_3^2)m^5 + (5\beta_1\beta_4 + 5\beta_2\beta_3)m^4 \\
& \quad +(4\beta_0\beta_4 + 4\beta_1\beta_3 + 2\beta_2^2)m^3 + (3\beta_0\beta_3 + 3\beta_1\beta_2)m^2 \\
& \quad +(2\beta_0\beta_2 + \beta_1^2)m + \beta_0\beta_1)P
\end{aligned} \tag{5.8.1}
$$

$$
\begin{aligned}
\dot{P} \;=\; & ((12\beta_4 m^2 + 6\beta_3 m + 2\beta_2)P^2 + 12\beta_4 P^3)z \\
& -420\beta_4^2 P^5 \\
& -(1260\beta_4^2 m^2 + 630\beta_3\beta_4 m + 90\beta_2\beta_4 + 45\beta_3^2)P^4 \\
& -(420\beta_4^2 m^4 + 420\beta_3\beta_4 m^3 + (180\beta_2\beta_4 + 90\beta_3^2)m^2 + (60\beta_1\beta_4 + 60\beta_2\beta_3)m \\
& \quad +12\beta_0\beta_4 + 12\beta_1\beta_3 + 6\beta_2^2)P^3 \\
& -(28\beta_4^2 m^6 + 42\beta_3\beta_4 m^5 + (30\beta_2\beta_4 + 15\beta_3^2)m^4 + (20\beta_1\beta_4 + 20\beta_2\beta_3)m^3 \\
& \quad +(12\beta_0\beta_4 + 12\beta_1\beta_3 + 6\beta_2^2)m^2 + (6\beta_0\beta_3 + 6\beta_1\beta_2)m + 2\beta_0\beta_2 \\
& \quad +\beta_1^2)P^2
\end{aligned} \tag{5.8.2}
$$

- Logarithmic formulation

$$
\begin{aligned}
\dot{m} \;=\; & ((12b\beta_4 + 2\beta_2)m + 3b\beta_3 + \beta_1)Pz \\
& -(420b^3\beta_4^2 + (90b^2\beta_2 + 12b\beta_0)\beta_4 + 45b^2\beta_3^2 + 12b\beta_1\beta_3 \\
& \quad +6b\beta_2^2 + 2\beta_0\beta_2 + \beta_1^2)Pm \\
& -(105b^3\beta_3 + 15b^2\beta_1)P\beta_4 + (15b^2\beta_2 + 3b\beta_0)P\beta_3 + 3bP\beta_1\beta_2 \\
& \quad +P\beta_0\beta_1
\end{aligned} \tag{5.8.3}
$$

$$
\begin{aligned}
\dot{P} \;=\; & (12b\beta_4 + 2\beta_2)P^2 z \\
& -420b^3P^2\beta_4^2 - (90b^2\beta_2 + 12b\beta_0)P^2\beta_4 - 45b^2P^2\beta_3^2 - 12bP^2\beta_1\beta_3 \\
& -6bP^2\beta_2^2 - 2P^2\beta_0\beta_2 - b\beta_1^2
\end{aligned} \tag{5.8.4}
$$

- EKF

$$
\dot{m} \;=\; (4\beta_4 m^3 + 3\beta_3 m^2 + 2\beta_2 m + \beta_1)Pz
$$

$$-(4\beta_4^2 m^7 + 7\beta_3\beta_4 m^6 + (6\beta_2\beta_4 + 3\beta_3^2)m^5 + (5\beta_1\beta_4 + 5\beta_2\beta_3)m^4 + (4\beta_0\beta_4 + 4\beta_1\beta_3 + 2\beta_2^2)m^3 + (3\beta_0\beta_3 + 3\beta_1\beta_2)m^2 + (2\beta_0\beta_2 + \beta_1^2)m + \beta_0\beta_1)P \quad (5.8.5)$$

$$\dot{P} = -(16\beta_4^2 m^6 + 24\beta_3\beta_4 m^5 + (16\beta_2\beta_4 + 9\beta_3^2)m^4 + (8\beta_1\beta_4 + 12\beta_2\beta_3)m^3 + (6\beta_1\beta_3 + 4\beta_2^2)m^2 + 4\beta_1\beta_2 m + \beta_1^2)P^2 \quad (5.8.6)$$

- Second order filter

$$\dot{m} = (4\beta_4 m^3 + 3\beta_3 m^2 + 2\beta_2 m + \beta_1)Pz - (24\beta_4^2 m^5 + 30\beta_3\beta_4 m^4 + (16\beta_2\beta_4 + 9\beta_3^2)m^3 + (6\beta_1\beta_4 + 9\beta_2\beta_3)m^2 + (3\beta_1\beta_3 + 2\beta_2^2)m + \beta_1\beta_2)P^2 - (4\beta_4^2 m^7 + 7\beta_3\beta_4 m^6 + (6\beta_2\beta_4 + 3\beta_3^2)m^5 + (5\beta_1\beta_4 + 5\beta_2\beta_3)m^4 + (4\beta_0\beta_4 + 4\beta_1\beta_3 + 2\beta_2^2)m^3 + (3\beta_0\beta_3 + 3\beta_1\beta_2)m^2 + (2\beta_0\beta_2 + \beta_1^2)m + \beta_0\beta_1)P \quad (5.8.7)$$

$$\dot{P} = (12\beta_4 m^2 + 6\beta_3 m + 2\beta_2)P^2 z - (72\beta_4^2 m^4 + 72\beta_3\beta_4 m^3 + (24\beta_2\beta_4 + 18\beta_3^2)m^2 + 12\beta_2\beta_3 m + 2\beta_2^2)P^3 - (28\beta_4^2 m^6 + 42\beta_3\beta_4 m^5 + (30\beta_2\beta_4 + 15\beta_3^2)m^4 + (20\beta_1\beta_4 + 20\beta_2\beta_3)m^3 + (12\beta_0\beta_4 + 12\beta_1\beta_3 + 6\beta_2^2)m^2 + (6\beta_0\beta_3 + 6\beta_1\beta_2)m + 2\beta_0\beta_2 + \beta_1^2)P^2 \quad (5.8.8)$$

Chapter 6

Application: Nonlinear Dynamics/Gaussian Sum Parameterization

6.1 Introduction

In this chapter, we shall consider the next step in choosing a parameterization after Chapter 5. Specifically, we will consider a weighted sum of second order parameterizations (Gaussians). In Section 2 we will derive the general equations for the scalar nonlinear filter and show their relationship with previous work and the results of Chapter 5. Then, in Section 3 we shall apply the results to a number of examples to verify perfomance and make comparisons. The goal is to show that terms in addition to those developed previously (in Chapter 5) resulting from considering pairs of pdf's result in a description of the underlying distribution more acurate than the ones obtained in previous chapters and by previous works (in particular, [10]). Finally, Section 4 summarizes the results of the chapter.

6.2 Equations

In this section we will apply our approximation scheme to the problem of determining the optimum values for the parameters of a pdf approximation composed of a weighted sum of Gaussian pdf's. The parameters are the means and covariances of the pdf's and the coefficients weighting them. The complete parameterization is given below.

$$\hat{p}(\underline{x};\underline{\alpha}(t)) = \sum_i c_i N(\underline{x};\underline{m}_i,P_i), \quad (6.2.1)$$

$$\text{where } N(\underline{x};\underline{m}_i,P_i) = e^{-\frac{1}{2}((\underline{x}-\underline{m})^T V_i(\underline{x}-\underline{m})+N ln(2\pi)-ln(det(V)))}, \quad (6.2.2)$$

$$\underline{\alpha}_i(t) = \begin{pmatrix} c_i \\ \underline{m}_i \\ V_i \end{pmatrix} \quad (6.2.3)$$

$$\text{and } P_i = V_i^{-1} \quad (6.2.4)$$

(Note that we have chosen not to use a separate normalization constant as was done in Chapters 4 and 5. This is done strictly in the interest of reducing the number of parameters and hence the dimension of the $F\dot{\underline{\alpha}} = g$ equation.)

The motivation for such a parameterization is two-fold. First is its ability to model pdf's which are multi-modal in nature. Second is the fact that integrals involving Gaussian pdf's are easy to evaluate in closed form. Note that a drawback of this parameterization is that we cannot place it in the form $\hat{p}(x;\underline{\alpha}(t)) = e^{\zeta}$ without destroying the Gaussian structures. Therefore, in designing a filter, we must revert back to the fundamental definitions of F and g (as in Sections 2.2 and 2.3) instead of the ones obtained in Chapter 3. We shall find, however, that for the inner product chosen in this chapter, the resultant F and g terms are still in the form of expectations over the approximating pdf.

Due to the increase in the number of parameters, evaluation of the $F\dot{\underline{\alpha}} = g$ equations must be done even more carefully than in previous chapters. For Equation 6.2.1, the ordering of the parameters into $\underline{\alpha}$ involves reformatting the mean vector and covariance matrix not for one, but for many, pdf's. In addition, we must now include weighting coefficients, c_i. For clarity and simplicity we will outline the derivation for scalar state models only.

Step one is to evaluate the F matrix. From its definition, we know that

$$F_{ij} = < \hat{p}_{\alpha_i}, \hat{p}_{\alpha_j} >,$$

where the α_i correspond to c_i, m_i, or V_i. We summarize the various derivatives needed for the computation of the F matrix and g vector below

$$\begin{aligned}
\hat{p} &= \sum_i c_i N(x; m_i, P_i) \\
\hat{p}_x &= -\sum_i c_i V_i (x - m_i) N(x; m_i, P_i) \\
\hat{p}_{xx} &= \sum_i c_i V_i (V_i (x - m_i)^2 - 1) N(x; m_i, P_i) \\
\hat{p}_{c_i} &= N(x; m_i, P_i) \\
\hat{p}_{m_i} &= c_i V_i (x - m_i) N(x; m_i, P_i) \\
\hat{p}_{V_i} &= \frac{1}{2} c_i (P_i - (x - m_i)^2) N(x; m_i, P_i).
\end{aligned} \tag{6.2.5}$$

It is clear from the various derivatives in Equation 6.2.5 that F_{ij} will, in general, involve a term polynomial in the α's multiplied by the product of a pair of (different) Gaussian densities.

For Gaussian pdf's it turns out that the Gaussian product term can be easily evaluated and is displayed in Equation 6.2.6.

$$N(x; m_i, P_i) N(x; m_j, P_j) = K_{ij} N(x; m_{ij}, P_{ij}), \tag{6.2.6}$$

where we have defined:

$$\begin{aligned}
K_{ij} &= N(m_i - m_j; 0, P_i + P_j) \\
m_{ij} &= \frac{P_i m_j + P_j m_i}{P_i + P_j} \\
P_{ij} &= \frac{P_i P_j}{P_i + P_j}.
\end{aligned}$$

Hence we shall not attempt to reformulate the approximation steps to simplify the calculations as was done in Chapter 3 and later applied in Chapters 4 and 5. Instead, we will use the Euclidean inner product and the relationships in Equation 6.2.6 to evaluate F and g.

Studying Equation 6.2.6 we find that the product of two Gaussian pdf's is a Gaussian pdf with a mean equal to the weighted average of the starting pdf means and a covariance which is a parallel combination of the starting pdf covariances.

Hence, the resulting pdf has the 'expected' center of mass and is narrower. In addition, the resulting pdf is multiplied by a constant which has an amplitude proportional to the spacing between the two starting pdf means. This constant turns out to be simply a zero mean Gaussian evaluated at $x = m_i - m_j$ with covariance $P_i + P_j$ (or, equivalently, a $m_i - m_j$ mean Gaussian evaluated at $x = 0$). The dependance of this constant on the mean spacing and variance will become important later in this exposition. Using Equation 6.2.6 we may write

$$F_{ij} = \int b_i b_j N(x; m_i, P_i) N(x; m_j, P_j) dx, \tag{6.2.7}$$

where the b_i's are the remaining terms associated with the $\hat{p}_{\alpha_i}$'s after factoring out the Gaussian pdf's, as

$$\begin{aligned} F_{ij} &= K_{ij} \int b_i b_j N(x; m_{ij}, P_{ij}) dx \\ &= K_{ij} \underbrace{E}_{N(x;m_{ij},P_{ij})} \{b_i b_j\}. \end{aligned} \tag{6.2.8}$$

A similar effect on the g vector will also occur.

Armed with Equation 6.2.6 we now proceed to determine F and g explicitly. First we must address the ordering issue.

In light of the fact that a common constant (K_{ij}) can be factored out of each Gaussian product, we will group the parameters according to pdf number. Hence we let

$$\underline{\alpha}(t) = \begin{pmatrix} \underline{a}_1 \\ \vdots \\ \underline{a}_i \\ \vdots \\ \underline{a}_n \end{pmatrix}, \tag{6.2.9}$$

$$\text{where } \underline{a}_i = \begin{pmatrix} c_i \\ m_i \\ V_i \end{pmatrix}. \tag{6.2.10}$$

So that we may also write

$$\hat{p}_{\underline{a}_i} = \begin{pmatrix} \hat{p}_{c_i} \\ \hat{p}_{m_i} \\ \hat{p}_{V_i} \end{pmatrix}. \tag{6.2.11}$$

This will in turn alow us to write the F matrix as follows: structure

$$F = \begin{pmatrix} F_{11} & \cdots & F_{1n} \\ \vdots & \ddots & \\ F_{n1} & \cdots & F_{nn} \end{pmatrix}, \tag{6.2.12}$$

$$\text{where } F_{ij} = < \hat{p}_{\underline{a}_i}, \hat{p}_{\underline{a}_j} >$$

$$= \begin{pmatrix} < \hat{p}_{c_i}, \hat{p}_{c_j} > & < \hat{p}_{c_i}, \hat{p}_{m_j} > & < \hat{p}_{c_i}, \hat{p}_{V_j} > \\ < \hat{p}_{m_i}, \hat{p}_{c_j} > & < \hat{p}_{m_i}, \hat{p}_{m_j} > & < \hat{p}_{m_i}, \hat{p}_{V_j} > \\ < \hat{p}_{V_i}, \hat{p}_{c_j} > & < \hat{p}_{V_i}, \hat{p}_{m_j} > & < \hat{p}_{V_i}, \hat{p}_{V_j} > \end{pmatrix}. \tag{6.2.13}$$

Similarly, we may break up the g vector as follows. Since the Fokker-Planck (F-P) operator, $\mathcal{L}^*(\cdot)$, is linear[1], we may write $\mathcal{L}^*(\hat{p})$ as

$$\mathcal{L}(\hat{p}) = \sum_{j=1}^{n} c_j \mathcal{L}(N(x; m_j, P_j)). \tag{6.2.14}$$

This in turn allows us to write

$$\underline{g} = < \hat{p}_{\underline{\alpha}}, \mathcal{L}^*(\hat{p}) > = \begin{pmatrix} < \hat{p}_{\alpha_1}, \mathcal{L}^*(\hat{p}) > \\ < \hat{p}_{\alpha_2}, \mathcal{L}^*(\hat{p}) > \\ \vdots \end{pmatrix} \tag{6.2.15}$$

$$= \sum_j c_j \begin{pmatrix} \underline{\gamma}_{1j} \\ \vdots \\ \underline{\gamma}_{ij} \\ \vdots \\ \underline{\gamma}_{nj} \end{pmatrix}, \tag{6.2.16}$$

$$\text{where } \underline{\gamma}_{ij} = \begin{pmatrix} < \hat{p}_{c_i}, \mathcal{L}(N(x; m_j, P_j)) > \\ < \hat{p}_{m_i}, \mathcal{L}(N(x; m_j, P_j)) > \\ < \hat{p}_{V_i}, \mathcal{L}(N(x; m_j, P_j)) > \end{pmatrix}. \tag{6.2.17}$$

Using this notation the $F\dot{\underline{\alpha}} = g$ equation becomes

$$\sum_j F_{ij} \dot{\underline{a}}_j = \sum_j \underline{\gamma}_{ij} c_j \quad \text{for } i = 1, n. \tag{6.2.18}$$

[1]Under the previous two formulations, the original F-P operator had to be modified making it nonlinear. Here, since we need no modification, it remains linear.

Now inserting the various derivatives of Equations 6.2.5 we find

$$
\begin{aligned}
F_{ij} &= K_{ij}E\{\begin{pmatrix} 1 & c_jV_jy_j & \frac{c_j}{2}(P_j - y_j^2) \\ c_iV_iy_i & c_ic_jV_iV_jy_iy_j & \frac{c_ic_jV_i}{2}(P_jy_i - y_iy_j^2) \\ \frac{c_i}{2}(P_i - y_i^2) & \frac{c_ic_jV_i}{2}(P_iy_j - y_jy_i^2) & \frac{c_ic_j}{4}(P_iP_j - P_jy_i^2 - P_iy_j^2 + y_i^2y_j^2) \end{pmatrix}\}, \\
\text{where } y_i &= (x - m_i) \qquad (6.2.19)
\end{aligned}
$$

where the expectation is taken over $N(x; m_{ij}, P_{ij})$. Fortunately, the result of this expectation is simple enough to display here:

$$
F_{ij} = K_{ij}\begin{pmatrix} 1 & \frac{c_j\Delta_{ij}}{R_{ij}} & \frac{c_jP_j^2(R_{ij}-\Delta_{ij}^2)}{2R_{ij}^2} \\ \frac{c_i\Delta_{ij}}{R_{ij}} & \frac{c_ic_j(R_{ij}-\Delta_{ij}^2)}{R_{ij}^2} & -\frac{c_ic_j\Delta_{ij}P_j^2(3R_{ij}-\Delta_{ij}^2)}{2R_{ij}^3} \\ \frac{c_iP_i^2(R_{ij}-\Delta_{ij}^2)}{2R_{ij}^2} & -\frac{c_ic_j\Delta_{ij}P_i^2(3R_{ij}-\Delta_{ij}^2)}{2R_{ij}^3} & \frac{c_ic_jP_i^2P_j^2(3R_{ij}^2+\Delta_{ij}^4-6R_{ij}\Delta_{ij}^2)}{4R_{ij}^4} \end{pmatrix},
$$

where (6.2.20)

$$\Delta_{ij} = m_i - m_j,$$

and

$$R_{ij} = P_i + P_j.$$

Similarly, for $\underline{\gamma}_{ij}$ we have

$$
\underline{\gamma}_{ij} = K_{ij}E\{\begin{pmatrix} 1 \\ c_iV_iy_i \\ \frac{c_i}{2}(P_i - y_i^2) \end{pmatrix}(\mathbf{f}(x)V_jy_j - \mathbf{f}_x(x) + \frac{Q}{2}V_j(V_jy_j^2 - 1))\},
$$

where (6.2.21)

$$y_i = (x - m_i)$$

with the expectation again over the combined density $N(x; m_{ij}, P_{ij})$. Here, however, the result of the expectation becomes too cumbersome to display. Therefore, we shall only display parts of it as needed.

Comment on the Results

Note that since K_{ij} peaks at $i = j$ and depends only on the absolute value of the difference between m_i and m_j, F_{ij} shall also peak at $i = j$. This has the effect of weighting the diagonal blocks of F most heavily and de-emphasizing off-diagonal terms. In particular, it will be the terms corresponding to each individual Gaussian alone which will be weighted most heavily. The same effect can be seen in the sum which forms the g vector.

To clarify the contibutions from the individual pdf's, we write the $F\dot{\underline{\alpha}} = g$ equation as

$$F_{ii}\dot{\underline{a}}_i + \sum_{j\neq i} F_{ij}\dot{\underline{a}}_j = \underline{\gamma}_{ii}c_i + \sum_{j\neq i}\underline{\gamma}_{ij}c_j, \quad \text{for } i = 1, n. \tag{6.2.22}$$

This implies that for a problem which produces small individual variances in comparison to the separation between means, our approximation scheme would treat all cross terms, $F_{ij}, \underline{\gamma}_{ij}$, as negligible, thus creating a block diagonal structure in F which would then allow us to separate the $F\dot{\underline{\alpha}} = g$ equation into n $F_{ii}\dot{\underline{\alpha}}_i = g_i$ equations, each corresponding to a single Gaussian filter operating independently. Using the above form, this corresponds to

$$F_{ii}\dot{\underline{a}}_i \;=\; \underline{\gamma}_{ii}c_i \tag{6.2.23}$$

Note that F_{ii} and $\underline{\gamma}_{ii}c_i$ will be equal to F and g of Chapter 5, respectively. In general, this corresponds to running n Chapter 5 type filters in parallel. For linear dynamics, it corresponds to running n Kalman filters in parallel. Encouragingly, this behavior is in line with what we would expect for n widely spread processes, since the amount of overlap between pdf's becomes insignifigant.

This scenario is exactly what Alspach [10] depends upon in order to make his filter work. However, since our goal is more general, we will be interested in just what these cross terms do in the g vector as well as in the F matrix. One would suspect that this cross term information should yield an improvement in performance in ambiguous situations where pdf's overlap greatly. As we see, this improvement is not without a cost. The computational cost of this improvement over n individual filters is a factor of n^2 (assuming an $O(M^3)$ algorithm for solving a $M \times M$ set of equations).

It should be obvious that if at any time $\underline{a}_i = \underline{a}_j$, for some $i \neq j$, the F matrix becomes singular. In fact, it can be shown that we need only the mean and variance of two densities in the sum to be equal to cause singularity in the F matrix and hence an ill-defined filter. This is the key drawback of the sum parameterization. In an actual application, such a situation must be circumvented either by special modifications or through an alternative parameterization. (In the following chapter, we will treat an alternative parameterization which will not have this problem.)

As a final note, the purpose of breaking up the F and g terms into separate blocks has not only been to help identify individual pdf contributions, but also to modularize the computations, thus simplifying the implementation.

6.3 Examples

In this section we shall go through a few examples to investigate the claims made in the previous section, particularly with respect to any performance advantages provided by the cross terms.

Scalar Examples

For clarity and ease of analysis we shall consider a 2-pdf sum as our parameterization.

We start by considering linear dynamics. Specifically, in Figure 6.3.1 we consider $\mathrm{f}(x) = x$ dynamics. We plot the results in $m \pm \sqrt{P}$ format to display the changes in the mean and variance. In studying the result we also found that the two weighting coefficients remained equal ($c_1 = c_2$) over the duration of the run, and hence we have not dedicated a plot to them. Due to the form of the dynamical model, we may expect to see an $x = x_0 e^t$ behavior in the mean, as in the single pdf case. Initially, at $t = 0$, we see exactly this behavior, however, for $t > 1$, we see quite a different behavior, namely, the means become constant. Note that this occurs at the same point the variances of the two pdf's become so large as to cause overlap. The explanation lies in the fact that the cross terms are no longer negligible since $K_{ij} = N(m_i - m_j; 0, P_i + P_j)$ is no longer small. This behavior can also be supported from the intuitive standpoint as follows.

At $t = 0$, the two pdf's have very little overlap and can thus be considered as two separate proccesses. Hence, each follow the dynamics equation independently. However, as time progresses, the two pdf's widen due to the nature of this same dynamics, causing an eventual large overlap, making the sum of the two pdf's appear as one very wide pdf. The dynamics of a single wide pdf centered about zero, as this

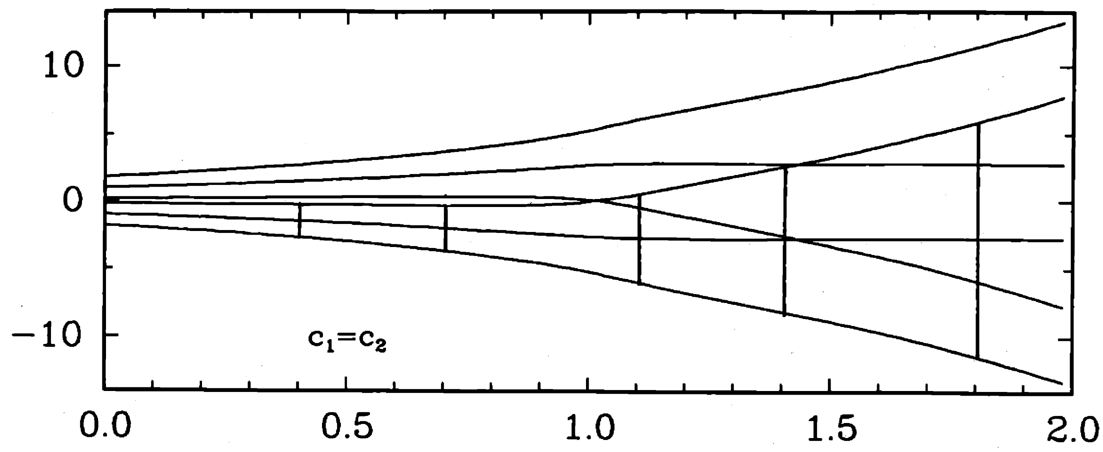


Figure 6.3.1: $\mathbf{f}(x) = x \quad p(x,0) = 0.5N(-1,0.667) + 0.5N(1,0.667) \quad T = 2$

one is, would be exactly to leave the mean unaffected while continuing to increase the variance. Hence, this example has shown us that the small variance scenario ($t < 1$) does indeed allow us to consider our sum approximation as two independent filters, thus verifying Equation 6.2.24, while as the variance increased (for $t > 1$), it has shown the merits of the cross terms in predicting correct behavior.

Next, we analyze the results for a nonlinear example which will show the merits of the overall scheme in dealing with complex dynamics. We select $\mathbf{f}(x) = x^2$ dynamics. Again, we have plotted means and variances in $m \pm \sqrt{P}$ format (Figure 6.3.3). However, in this case, c_1, c_2 do vary, so they appear in Figure 6.3.4. Figure 6.3.2 displays $\hat{p}(x; \underline{\alpha}(t))$ versus time for completeness.

From the form of the dynamics we would expect that the general movement of the pdf will be toward the right, since $\dot{x}$ is always positive. However, we should see a bunching phenomenon from the left of $x = 0$, since $\dot{x}$ also becomes zero at this point. On the right, the pdf should simply continue heading toward large x at an increasing rate, while spreading. We see precisely this behavior in the pdf plot of Figure 6.3.2, although the right pdf is not very prominent.

However, in the associated $m \pm \sqrt{P}$ plot of Figure 6.3.3, we see a very peculiar behavior. An explanation based on the deterministic nature of the dynamics ($Q = 0$) is as follows. The solution to $\dot{x} = x^2$ is of the form $x = \frac{1}{c-t}$. Assuming $c > 0$,

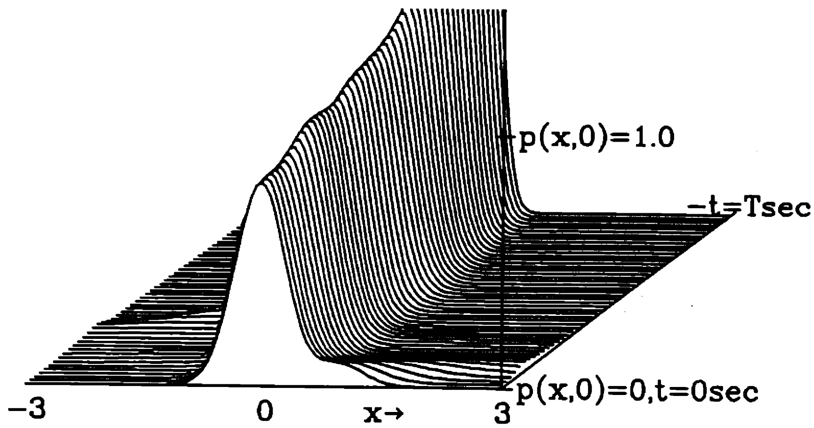


Figure 6.3.2: $\mathbf{f}(x) = x^2 \quad p(x,0) = 0.9N(0,0.1) + 0.1N(1,.1) \quad T = 5$

x is initially equal to $\frac{1}{c}$. As t approaches c, x heads towards infinity. For t slightly greater than c x is at $-\infty$ and asymtotically heads toward zero. From Figure 6.3.3 we see very much this behavior with $c = 1$ and the area around $t = 1$ approximated by a smooth transition. This is encouraging since it verifies that our filtering scheme is attempting to approximate even the unstable portion of this model.

Finally, we note that for the single Gaussian pdf case with $m(0) = 1$ (as the right pdf above), the mean and variance simply head off to $+\infty$, while for $m(0) = 0$, the same path results as for the left pdf above are obtained.

Finally we display an example (Figure 6.3.5) which shows the stability of our approximation and allows for further verification. In Chapter 5, we saw the $\mathbf{f}(x) = -x^3$ example and will see much of it later in Chapter 7, hence, it will be useful to describe it here. From this form of dynamics, we would expect that the true underlying density would bunch up around $x = 0$ and eventually become impulsive at 0. In reference to our example, the two pdf's should quickly move in until their means equal ± 1, at which point the rate toward zero should decrease, while the variance decreases. Encouragingly, from Figure 6.3.5 we see precisely this behavior.

With this motivation, we run through the same example with non-zero process noise to investigate its effect on the parameters of our approximation. Note that

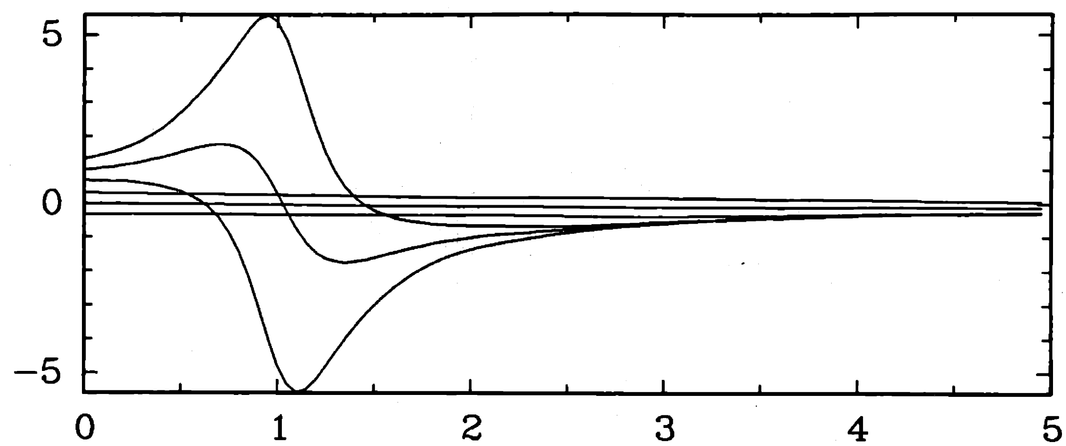


Figure 6.3.3: $\mathbf{f}(x) = x^2 \quad p(x,0) = 0.9N(0,0.1) + 0.1N(1,0.1) \quad T = 5$

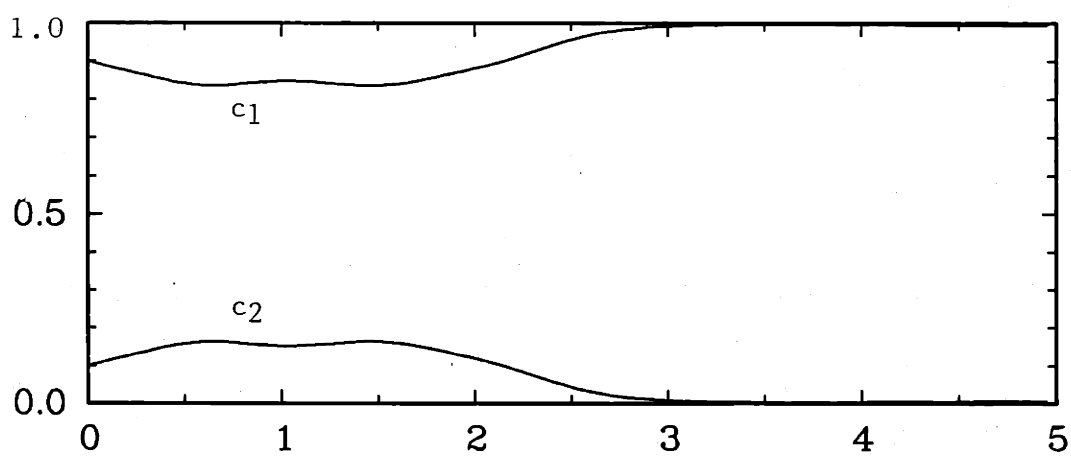


Figure 6.3.4: $\mathbf{f}(x) = x^2 \quad p(x,0) = 0.9N(0,0.1) + 0.1N(1,0.1) \quad T = 5$

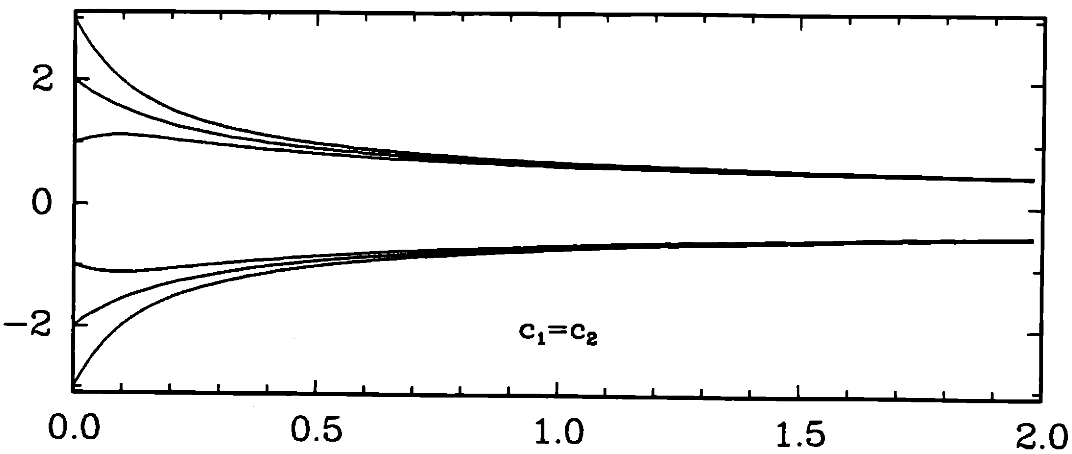


Figure 6.3.5: $f(x) = -x^3 \quad p(x,0) = 0.5N(-2,1) + 0.5N(2,1) \quad T = 2$

the scalar $\dot{x} = -x^3$ example in Chapter 5 also included process noise and displayed a stable steady state behavior which the EKF and Second Order filters did not. We see from Figure 6.3.6 a similar behavior to that of Figure 6.3.5, as would be expected. Even though the process noise has increased the steady state variances, a steady state due to the superlinear characteristic of $\dot{x} = -x^3$ is still reached.

Hence we have seen in this section that the Gaussian sum parameterization not only shows promise in handling ambiguous situations but also lends itself to a wide variety of dynamical models. In what follows we shall outline the results of a vector model for completeness.

Vector Example

In this section we shall briefly outline a vector example for completeness. We consider the PLL example of Chapter 5 [14] for the same intuitive motivations as before. Referring to Chapter 5, the PLL example basically models a system of two sinewaves, x_1 and x_2, in quadrature whose center frequency varies in a Brownian fashion. We shall first make comparisons between the results of a Chapter 5 example and the results of a 2-pdf sum filter. We limit the sum to 2 pdfs, due to the increase in complexity of the filter resulting from the consideration of vector quantities.

For clarity we have used two contour plots to display the two pdfs in x_1, x_2 (in-phase versus quadrature) space as was done with the single pdf in Chapter 5.

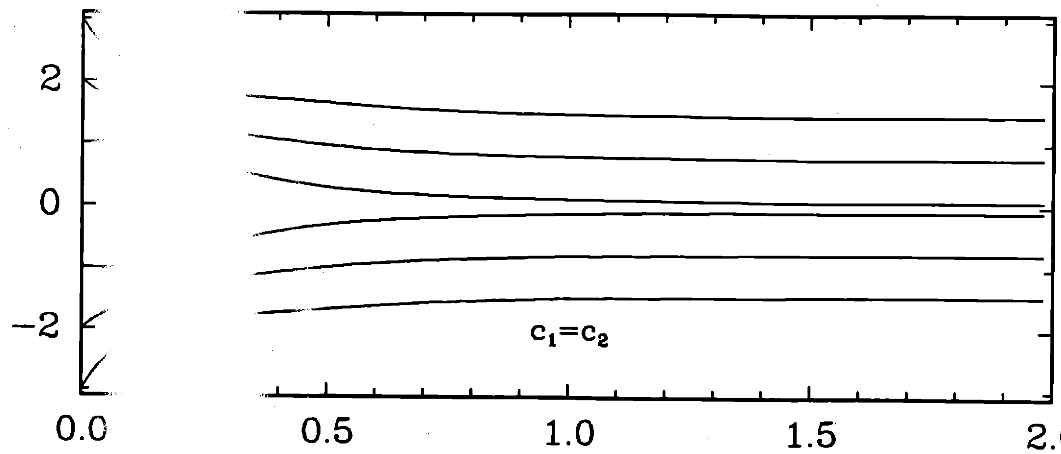


Figure 6 $-x^3 + v \quad Q = 3 \quad p(x,0) = 0.5N(-2,1) + 0.5N(2,1) \quad T = 2$

The plot e^{-1} (i.e. $2-\sigma$) contours of each pdf separately, Figure 6.3.7 for pdf 1 anc 3.8 for pdf 2. In addition, for comparison purposes, we plot the single pd Chapter 5 for the same simulation conditions as pdf 1 above in Figure 6.

In orc rly observe the movement of the individual pdf's, we selected the initia s so that the means of the two pdf's were widely spread apart (90 degre hase). For simplicity, we chose the initial variances to be equal. We note eighting coefficients, c_1 and c_2, remain equal throughout the simulatio herefore not displayed here.

It is cl igure 6.3.7 and Figure 6.3.9 that the schemes of Chapter 5 and Chapter milar results. The same behavior of constraining most of the pdf's mas unit circle is retained. However we see very little, if any, effect due to pc Both pdf's progress independently as would be expected for a non-overl io. This should not be surprising however since the weighted sum of th s, as a whole (Figure 6.3.7+6.3.8), adhere to the condition of keeping th of the pdf mass on the unit circle while progressing about it.

In the ple, we shall consider the same model and conditions as before but also lates from a noisy observation of the in-phase sinewave component. ontour plotting scheme of the previous example will be used

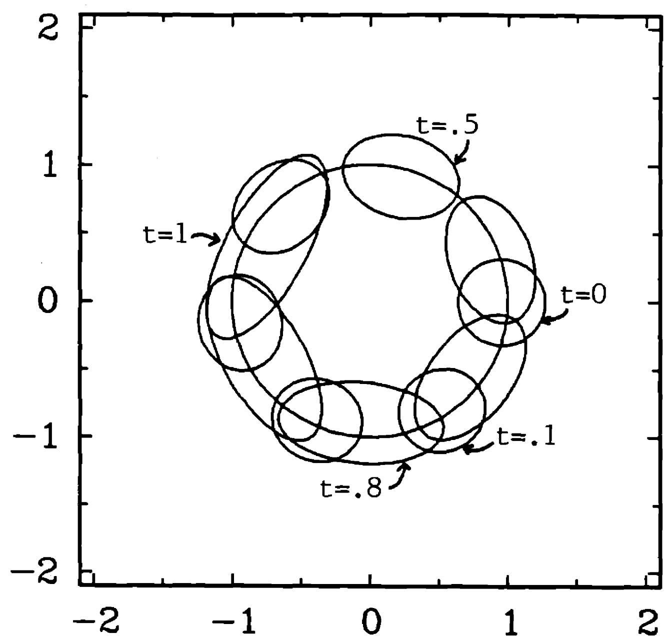


Figure 6.3.7: Plot of x_1 versus x_2 for pdf 1

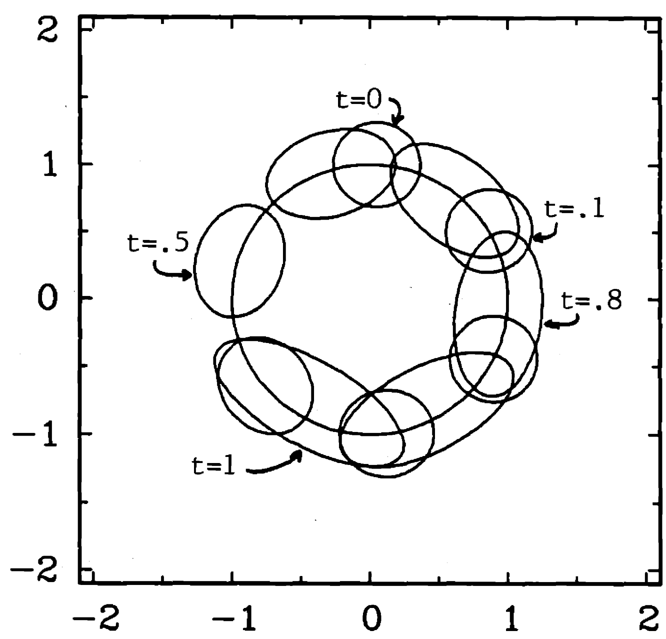


Figure 6.3.8: Plot of x_1 versus x_2 for pdf 2

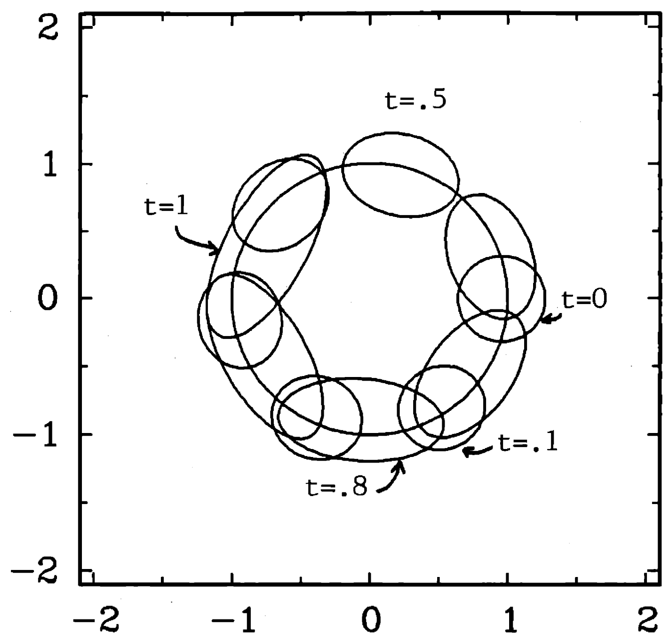


Figure 6.3.9: Plot of x_1 versus x_2 for the single pdf filter of Chapter 5

here.

In order to see more clearly the effects of the observations, we chose initial conditions so that the two pdf's had means which were widely spread apart and also spaced apart from the mean of the observation. As before, we chose the initial variances to be equal.

We see from Figures 6.3.10 and 6.3.11 that the two pdf paths eventually find the observed sinwave path. Note that even pdf 2, which started completely out of phase, locks onto the measurement.

We also see in both plots the overall reduction of the variance due to the measurements, as expected, in addition to the slight narrowing of the pdf's in the x_1 direction due to the fact that our observation is of x_1.

Finally, Figure 6.3.12 shows the change in the weighting functions versus time. Initially, pdf 2 is further away from the observations than pdf 1. Hence, we should expect that pdf 2's contibution to the total pdf approximation should be of least importance (weighted least). As pdf 2's parameters come in line with the obser-

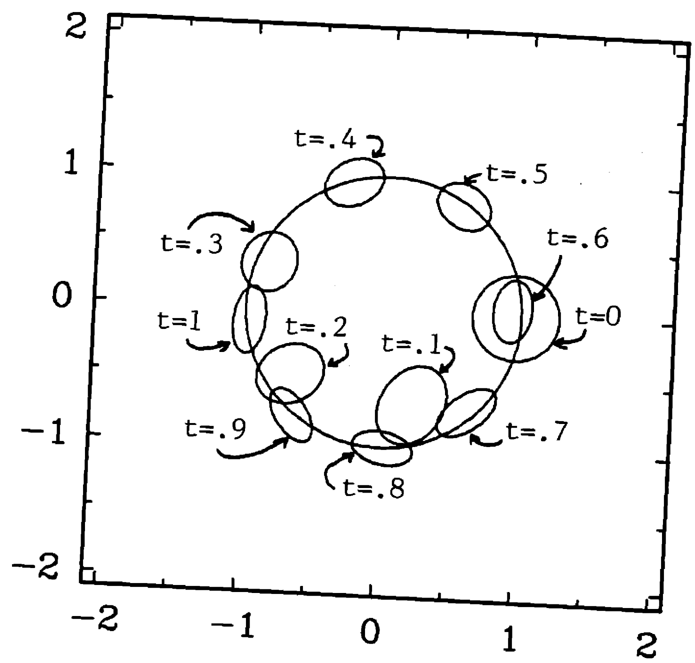


Figure 6.3.10: Plot of x_1 versus x_2 for pdf 1

vations, we would expect this importance (and hence its weighting) to increase. Encouragingly, we see exactly this behavior reflected in the weighting coefficients in Figure 6.3.12, starting out with an initial increase in c_2, corresponding to pdf 2 being close to the observation, then the sudden decrease in c_2 ($t > 0.18$), corresponding to the sudden large difference between the observations and pdf 2's mean ($t \approx 0.18$), and finally leveling off ($t > 0.3$) to its original value corresponding to pdf 2 attaining the correct phase.

Hence, what we have seen in this section is one more verification of the Gaussian sum parameterization's ability to model complex processes, such as a 3-dimensional nonlinear vector process, with and without measurements.

6.4 Summary

In this chapter we have considered a weighted sum of Gaussians as our approximating density, where the parameters are the weights, means and variances of the

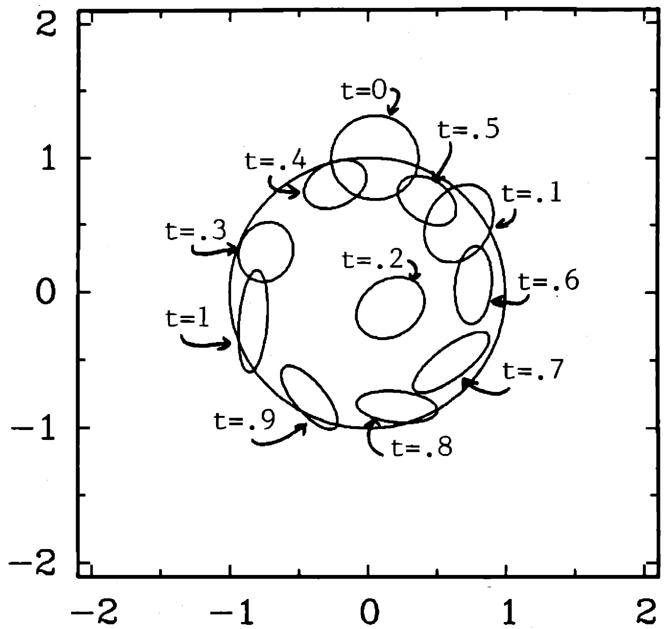


Figure 6.3.11: Plot of x_1 versus x_2 for pdf 2

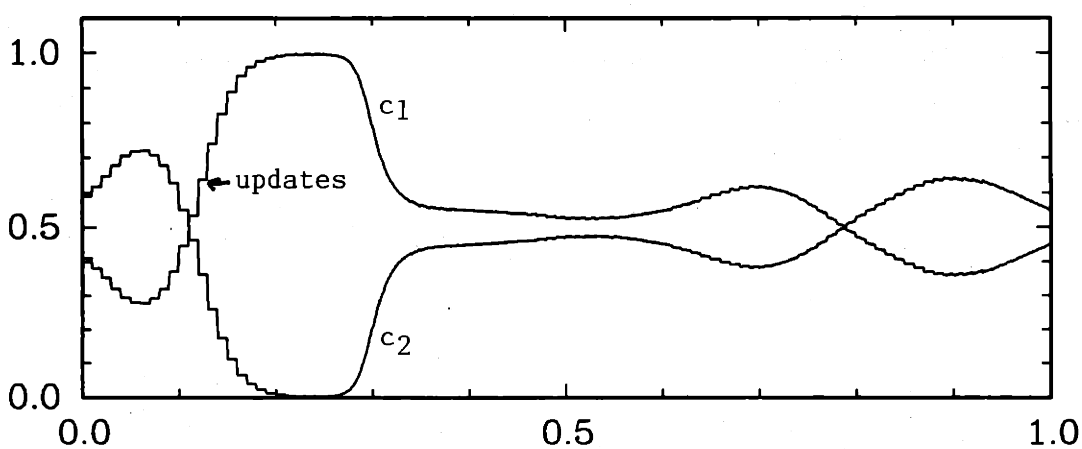


Figure 6.3.12: Plot of c_1 and c_2 versus time

Gaussians. In Section 2 we applied our fundamental approximation formulae to this parameterization and found that, although $\hat{p}(x;\underline{\alpha}(t))$ is not in exponential form, the resultant equations under the Euclidean inner product were relatively easy to evaluate. We found that by formatting the equations according to pdf index not only modularizes implementation, but also allows us to consider the n independent Gaussian implementation as a special case of the work in this chapter. This separation occurs when there is minimal overlap between the Gaussians in the sum. We find in Section 3, however, that it is just in the cases of heavy overlap that our approach to the Gaussian sum filter excels due to the many cross terms which come out of our general formulation. One drawback of the parameterization presented here becomes apparent when two pdf's in the sum cross in parameter space,i.e, become identical at some point in time. Although this may not be a major drawback, in the following chapter we shall consider an alternative parameterization.

Chapter 7

Non-linear Dynamics/ Quartic Parameterization

7.1 Introduction

In the previous chapters we considered approximations which involved second order and sum of second order parameterizations. Chapter 5 used a single-modal pdf to approximate the solution to the differential equation of concern, while Chapter 6 made use of the well known properties of a Gaussian sum to generate a multi-modal pdf. In this chapter we will consider a parameterization similar to the one discussed in Chapter 4 under the logarithmic formulation, but with a few extra terms. Specifically

$$\hat{p}(x;\underline{\alpha}(t)) = e^{\alpha_0+\alpha_1 x+\alpha_2 x^2+\alpha_3 x^3+\alpha_4 x^4} \tag{7.1.1}$$

$$\text{where } \underline{\alpha}(t) = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix}. \tag{7.1.2}$$

This form of a pdf has very little background with respect to its moments. This is due primarily to the fact that closed form expressions for integrals involving this function do not exist. As we shall see, the lack of such expressions forces us to apply our scheme using the logarithmic formulation.

The motivation for selecting such a parameterization lies in simplifications afforded by its purely exponential structure, unlike the approximation of Chapter 6, and its multi-modality, unlike Chapter 5. The same motivation led Lo [12] to his work on exponential Fourier series. Like our approach, the exponential structure of his approximating pdf makes the update step trivial. However unlike his work we are able to consider complex dynamics. This turned out to be the major drawback of Lo's formulation of the pdf approximation problem. Clearly we can consider the work on exponential Fourier series as a special case of our general approach and therefore eliminate the drawback due to the dynamical portion of the model.

Due to the increase in complexity associated with ordering for vector versions of Equation 7.1.1 and for clarity, this chapter will concern itself only with scalar processes.

Overviewing the chapter, Section 2 will derive the equations for our filter using the quartic pdf (Equation 7.1.1), initially leaving the dynamics of the process model, $\mathbf{f}(x)$, unspecified. Later, in Section 3, we will choose a specific form for $\mathbf{f}(x)$ as we did in Chapter 5 for comparison purposes. Section 4 will discuss some of the numerical issues associated with the resulting equations. Then in Section 5 we will run through a few examples using specific dynamical models, $\mathbf{f}(x)$, and discuss the results. In Section 6 we apply the results of Section 2 to a particular example which includes nonlinear measurements and compare the results with the EKF. Finally, in Section 7, we summarize the results of this chapter.

7.2 Equations

In this section we will solve the $F\underline{\dot{\alpha}} = g$ equation for $\underline{\dot{\alpha}}$. We shall first specify the parameterization then compute the F matrix followed by the g vector. We list the parameterization along with some useful functions of the parameterization here

$$\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}} \tag{7.2.1}$$

$$\hat{\varsigma} = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4 \tag{7.2.2}$$

$$\begin{aligned} \hat{\varsigma}_{\alpha_i} &= x^i \qquad 0 \le i \le 4 && (7.2.3)\\ \hat{\varsigma}_x &= \alpha_1 + 2\alpha_2 x + 3\alpha_3 x^2 + 4\alpha_4 x^3 && (7.2.4)\\ \hat{\varsigma}_{xx} &= 2\alpha_2 + 6\alpha_3 x + 12\alpha_4 x^2. && (7.2.5) \end{aligned}$$

Now, from Figure 3.3.2 in Chapter 3

$$F_{ij} = E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_{\alpha_j}\} \qquad (7.2.6)$$

with the expectation over either $\hat{p}(x;\underline{\alpha}(t))$ or $N(0,b)$,depending on the norm chosen to drive the approximation. From Equation 7.2.3 this becomes

$$F_{ij} = E\{x^{i+j}\}. \qquad (7.2.7)$$

If closed forms for the moments of $\hat{p}(x;\underline{\alpha}(t))$ existed, Equation 7.2.7 could be computed easily under either formulation. However, since closed forms do not exist for moments of $\hat{p}(x;\underline{\alpha}(t))$ in terms of $\underline{\alpha}(t)$, we must resort to working under the logarithmic formulation, since it will only involve expectations over the normal distribution, $N(0,b)$. For the quartic parameterization described by Equations 7.2.1,7.2.2, F then becomes:

$$F = \begin{pmatrix} 1 & 0 & b & 0 & 3b^2 \\ 0 & b & 0 & 3b^2 & 0 \\ b & 0 & 3b^2 & 0 & 15b^3 \\ 0 & 3b^2 & 0 & 15b^3 & 0 \\ 3b^2 & 0 & 15b^3 & 0 & 105b^4 \end{pmatrix} \qquad (7.2.8)$$

and for later convenience, its inverse is given by

$$F^{-1} = \begin{pmatrix} 15/8 & 0 & -5/(4b) & 0 & 1/(8b^2) \\ 0 & 5/(2b) & 0 & -1/(2b^2) & 0 \\ -5/(4b) & 0 & 2/b^2 & 0 & -1/(4b^3) \\ 0 & -1/(2b^2) & 0 & 1/(6b^3) & 0 \\ 1/(8b^2) & 0 & -1/(4b^3) & 0 & 1/(24b^4) \end{pmatrix}. \qquad (7.2.9)$$

Note that the F matrix is 50% sparse. This will greatly simplify later computations and is a form that is retained even when we go to higher order distributions.

CHAPTER 7. NON-LINEAR/QUARTIC

Now that we have the F matrix, the next step is to compute the g vector. This is given by (from Figure 3.4.3 of Chapter 3)

$$g_i = E\{\hat{\varsigma}_{\alpha_i}\hat{\varsigma}_t\} \quad (7.2.10)$$

with the expectation over $N(0,b)$. $\hat{\varsigma}_t$ is equal to

$$\hat{\varsigma}_t = -(\mathbf{f}_x(x) + \hat{\varsigma}_x \mathbf{f}(x)) + (1/2)Q(\hat{\varsigma}_x^2 + \hat{\varsigma}_{xx}) \quad (7.2.11)$$

(since we will be dealing exclusively with the logarithmic formulation). Leaving $\mathbf{f}(x)$ unspecified for the moment, $\underline{g}$ becomes

$$\underline{g} = -E\underbrace{\begin{pmatrix} \mathbf{f}_x & \mathbf{f} & 2\mathbf{f}x & 3\mathbf{f}x^2 & 4\mathbf{f}x^3 \\ \mathbf{f}_x x & \mathbf{f}x & 2\mathbf{f}x^2 & 3\mathbf{f}x^3 & 4\mathbf{f}x^4 \\ \mathbf{f}_x x^2 & \mathbf{f}x^2 & 2\mathbf{f}x^3 & 3\mathbf{f}x^4 & 4\mathbf{f}x^5 \\ \mathbf{f}_x x^3 & \mathbf{f}x^3 & 2\mathbf{f}x^4 & 3\mathbf{f}x^5 & 4\mathbf{f}x^6 \\ \mathbf{f}_x x^4 & \mathbf{f}x^4 & 2\mathbf{f}x^5 & 3\mathbf{f}x^6 & 4\mathbf{f}x^7 \end{pmatrix}}_{\Phi}\underbrace{\begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix}}_{\underline{\tilde{\alpha}}_1} \quad (7.2.12)$$

$$+\frac{Q}{2}\underbrace{\begin{pmatrix} 0 & 15b^3 & 0 & 3b^2 & 0 & b & 0 & 1 \\ 105b^4 & 0 & 15b^3 & 0 & 3b^2 & 0 & b & 0 \\ 0 & 105b^4 & 0 & 15b^3 & 0 & 3b^2 & 0 & b \\ 945b^5 & 0 & 105b^4 & 0 & 15b^3 & 0 & 3b^2 & 0 \\ 0 & 945b^5 & 0 & 105b^4 & 0 & 15b^3 & 0 & 3b^2 \end{pmatrix}}_{\tilde{G}}\underbrace{\begin{pmatrix} 0 \\ 16\alpha_4^2 \\ 24\alpha_3\alpha_4 \\ (16\alpha_2\alpha_4 + 9\alpha_3^2) \\ (8\alpha_1\alpha_4 + 12\alpha_2\alpha_3) \\ (12\alpha_4 + 6\alpha_1\alpha_3 + 4\alpha_2^2) \\ (6\alpha_3 + 4\alpha_1\alpha_2) \\ (2\alpha_2 + \alpha_1^2) \end{pmatrix}}_{\underline{\tilde{\alpha}}_2}$$

or in compact notation

$$\underline{g} = \Phi\underline{\tilde{\alpha}}_1 + \frac{Q}{2}\tilde{G}\underline{\tilde{\alpha}}_2. \quad (7.2.13)$$

Now multiplying through by F^{-1} gives $\underline{\dot{\alpha}}$ to be

$$\underline{\dot{\alpha}} = F^{-1}\Phi\underline{\tilde{\alpha}}_1 + \frac{Q}{2}\begin{pmatrix} 0 & 15b^3 & 0 & 0 & 0 & 0 & 0 & 1 \\ -210b^3 & 0 & -15b^2 & 0 & 0 & 0 & 1 & 0 \\ 0 & -45b^2 & 0 & 0 & 0 & 1 & 0 & 0 \\ 105b^2 & 0 & 10b & 0 & 1 & 0 & 0 & 0 \\ 0 & 15b & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}\underline{\tilde{\alpha}}_2. \quad (7.2.14)$$

Note that the right hand side of Equation 7.2.14 is strictly in terms of $\underline{\alpha}$, hence closure over the parameters is obtained. This was not imediately apparent in previous chapters. In Chapter 4, for example, under the square root formulation $\dot{\underline{\alpha}}$ was a function of both $\underline{\alpha}$ and m, P (Equation 4.2.16).

7.3 Specific Dynamics

In the previous chapters we translated the parameters into means and variances for study. For the quartic parameterization of this chapter, however, the characteristics of mean and variance do not have clear meanings. In addition, retaining the original form (Equation 7.2.2) ensures that the measurement update step is kept trivial over a large class of measurement functions. An example of such an update step will be shown in Section 7.6. Because of these reasons, subsequent study and simulations will involve only the original parameterization. The only exception to this will be for display purposes where characteristics such as peak location (mode) and width may lead to a better understanding of the underlying process.

Again the form of Equation 7.2.14 leads to little insight, so, we will choose a specific $\mathbf{f}(x)$ and solve for $\dot{\underline{\alpha}}$. We choose the same $\mathbf{f}(x)$ chosen in the Section 5.2 of Chapter 5 for its clear separation of the various aspects of filter behavior. We restate it here for convenience:

$$\mathbf{f}(x) = f_0 + f_1 x + f_2 x^2 + f_3 x^3 + f_4 x^4. \tag{7.3.1}$$

In the equations for g, we find that the Φ term breaks up in the same way as the Q term in Equation 7.2.13 above:

$$\Phi = \tilde{G} \underbrace{\begin{pmatrix} 0 & 0 & 0 & 0 & -4f_4 \\ 0 & 0 & 0 & -3f_4 & -4f_3 \\ 0 & 0 & -2f_4 & -3f_3 & -4f_2 \\ 0 & -f_4 & -2f_3 & -3f_2 & -4f_1 \\ -4f_4 & -f_3 & -2f_2 & -3f_1 & -4f_0 \\ -3f_3 & -f_2 & -2f_1 & -3f_0 & 0 \\ -2f_2 & -f_1 & -2f_0 & 0 & 0 \\ -f_1 & -f_0 & 0 & 0 & 0 \end{pmatrix}}_{\theta_f} . \tag{7.3.2}$$

This allows us to write $F\underline{\dot{\alpha}} = g$ as

$$F\underline{\dot{\alpha}} = \tilde{G}\underline{\tilde{g}} \tag{7.3.3}$$

thus making

$$\underline{\dot{\alpha}} = \underbrace{\begin{pmatrix} 0 & 15b^3 & 0 & 0 & 0 & 0 & 0 & 1 \\ -210b^3 & 0 & -15b^2 & 0 & 0 & 0 & 1 & 0 \\ 0 & -45b^2 & 0 & 0 & 0 & 1 & 0 & 0 \\ 105b^2 & 0 & 10b & 0 & 1 & 0 & 0 & 0 \\ 0 & 15b & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}}_{F^{-1}\tilde{G}} \underline{\tilde{g}} \tag{7.3.4}$$

where from Equation 7.2.13 and 7.3.3, we have

$$\underline{\tilde{g}} = \theta_f \underline{\tilde{\alpha}}_1 + \frac{Q}{2}\underline{\tilde{\alpha}}_2 \tag{7.3.5}$$

Multiplying out terms

$$\underline{\dot{\alpha}} = -\begin{pmatrix} f_1 & f_0 & 0 & 45b^3 f_4 & 60b^3 f_3 \\ 2f_2 & f_1 & 2f_0 - 30b^2 f_4 & -45b^2 f_3 & -840b^3 f_4 - 60b^2 f_2 \\ 3f_3 & f_2 & 2f_1 & 3f_0 - 135b^2 f_4 & -180b^2 f_3 \\ 4f_4 & f_3 & 20bf_4 + 2f_2 & 30bf_3 + 3f_1 & 420b^2 f_4 + 40bf_2 + 4f_0 \\ 0 & f_4 & 2f_3 & 45bf_4 + 3f_2 & 60bf_3 + 4f_1 \end{pmatrix} \underline{\tilde{\alpha}}_1$$

$$+\frac{Q}{2}\begin{pmatrix} 240b^3\alpha_4^2 + 2\alpha_2 + \alpha_1^2 \\ -360b^2\alpha_3\alpha_4 + 6\alpha_3 + 4\alpha_1\alpha_2 \\ -720b^2\alpha_4^2 + 12\alpha_4 + 6\alpha_1\alpha_3 + 4\alpha_2^2 \\ 240b\alpha_3\alpha_4 + 8\alpha_1\alpha_4 + 12\alpha_2\alpha_3 \\ 240b\alpha_4^2 + 16\alpha_2\alpha_4 + 9\alpha_3^2 \end{pmatrix} \quad (7.3.6)$$

Note the structure and simplicity associated with θ_f and $F^{-1}\tilde{G}$. In the next two sections we will comment on these results and their implementations. Following this we will analyze the behavior of our result by way of examples.

7.4 Numerical Issues

Symbolically, Equation 7.3.6 can be written as

$$\underline{\dot{\alpha}} = \underbrace{F^{-1}\tilde{G}\theta_f}_{A}\,\underline{\tilde{\alpha}}_1 + \frac{Q}{2}\underbrace{F^{-1}\tilde{G}}_{B}\,\underline{\tilde{\alpha}}_2. \quad (7.4.1)$$

From the standpoint of implementation on discrete machines, Equation 7.4.1 lends some insight as to how to determine the rate of change for $\underline{\alpha}$ and thus an idea on how to choose the time steps in a simulation of Equation 7.4.1. In particular, if we have no process noise, $Q = 0$, Equation 7.4.1 has the form of a simple linear set of equations. Specifically

$$\underline{\dot{\alpha}} = \tilde{A}\underline{\alpha} + \underline{c} \quad (7.4.2)$$

where $\underline{c}$ =first column of A and $\tilde{A} = A$ with its first column replaced by 0's. Since the solution to Equation 7.4.2 is well known ($\underline{\alpha}(t) = e^{\tilde{A}t}(\underline{\alpha}(0) + k) - k$ where k is some constant), we can use it to determine the dominant modes of Equation 7.4.2 (corresponding to the largest eigenvalue of $\tilde{A}$) then from this choose a step size small enough not to overwhelm this mode in the computer implementation, i.e.,

$$\text{timestep} < 1/max(\lambda_i) \quad \lambda_i = \text{eigenvalues of } \tilde{A} \quad (7.4.3)$$

Although this scheme does not take into account the process noise, it should give us a starting value for the step size for implementation and hindsight into what to expect for $\underline{\alpha}$.

7.5 Comparisons

In this section we will try to derive some intuition into the operation of quartic pdf filters. We will accomplish this by simulating Equation 7.4.1 (or Equation 7.3.6) for various values of f_0, f_1, f_2, f_3, f_4 in $\mathbf{f}(x)$. Then, by studying the progression of the peaks (modes) of $\hat{p}(x;\underline{\alpha}(t))$ through $\underline{\alpha}$, we will both heuristically and analytically verify the correct evolution of our approximating pdf.

Before we solve for $\underline{\alpha}$, consider what might be a reasonable plot of our approximation. This will help us to get an idea for sensible values of $\underline{\alpha}(0)$, to start with, as well as of the values of $\underline{\alpha}(t)$ which result. We have the following facts: (a) a normalized pdf need not be maintained, (b) a maximum of two peaks can be expected due to the order of the approximation, (c) $\hat{\varsigma} \to -\infty$ as $x \to \pm\infty$ provided $\alpha_4 < 0$, which will be the case in any reasonable approximation. With these properties in mind, an appropriate $\hat{\varsigma}$ to be used as a vehicle to analyze our filter might be

$$\begin{aligned} \varsigma &= -\int_0^x (y-z_1)(y-z_2)(y-z_3)dy + z_0 \qquad (7.5.1) \\ &= -\frac{1}{4}x^4 + \frac{1}{3}(z_1+z_2+z_3)x^3 - \frac{1}{2}(z_1z_2+z_1z_3+z_2z_3)x^2 + z_1z_2z_3x + z_0 \end{aligned}$$

where the z_i's are the location of the two peaks and one valley. (Note that here we are restricting our attention to real z_i's. Complex z_i's will force us to interpret the pdf's in a different manner.) A plot of such a function is shown in Figure 7.5.1. From the point of view of selecting an initial value for $\underline{\alpha}(t)$ and later for tracking the movements of the pdf's along the x axis, it is useful to associate the parameters of this function, z_1, z_2, z_3, with the parameters of our filter, $\underline{\alpha}$. We do not start with this as our parameterization, i.e., $z_i = \alpha_i$, for a number of reasons. First is the lack of generality of this form as alluded to in Section 7.2. Second is the instability problem which results when two or more of the z_i's are equal or complex.

The relationships between the α_i's and the z_i's can be found by matching terms in Equation 7.5.1 and $\hat{\varsigma}$, and solving for the zero's of $\hat{\varsigma}_x$. These are summarized below:

$$\alpha_4 = -1/4 \qquad (7.5.2)$$

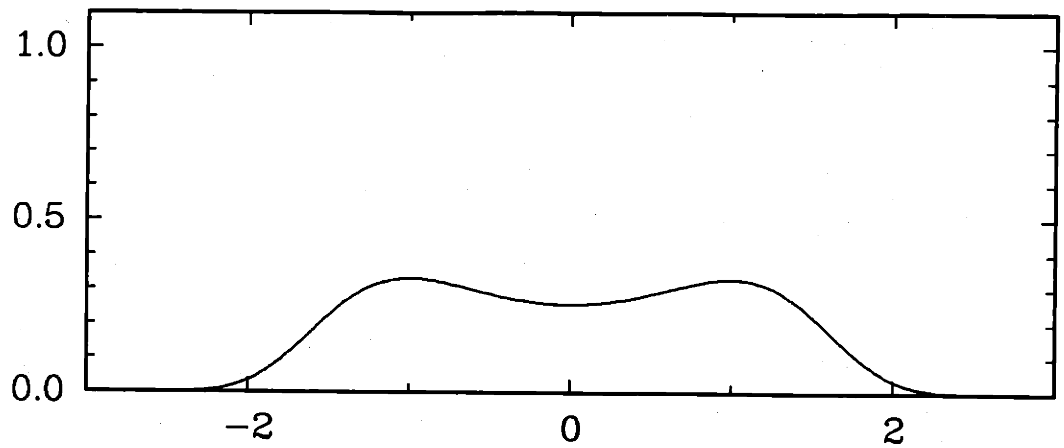


Figure 7.5.1: A 3 zero pdf

Case	f(x)	Q	I.C.	Figures
1	1	0	Gauss,quartic	7.5.2,7.5.3
2	$\pm x$	0	Gauss,quartic	7.5.4,7.5.5,7.5.6,7.5.7
3	x^2	0	Gauss,quartic	7.5.8,7.5.9
4	$\pm x^3$	0	Gauss,quartic	7.5.12,7.5.13,7.5.14,7.5.15
5	x^4	0	Gauss,quartic	7.5.10,7.5.11
6	0	1	Gauss,quartic	7.5.23,7.5.24
7	x^2	1	Gauss,quartic	7.5.25,7.5.26

Table 7.1: Various pdf test cases

$$\alpha_3 = (z_1 + z_2 + z_3)/3 \tag{7.5.3}$$

$$\alpha_2 = -(z_1 z_2 + z_1 z_3 + z_2 z_3)/2 \tag{7.5.4}$$

$$\alpha_1 = z_1 z_2 z_3 \tag{7.5.5}$$

$$\alpha_0 = z_0 \tag{7.5.6}$$

For the reverse case, finding the peak and valley locations form the α_i's, we must solve $\hat{\varsigma}_x = 0$ for x. These are just the usual polynomial solutions and will be used to analyze the evolution of the pdf as a function of its peak and valley locations.

In summary, this section will plot and analyze the pdf's resulting from our filter for each of the cases listed in Table 7.1. Note that for the first 5 cases, a closed form solution for $\underline{\alpha}(t)$ may be obtained as noted in Section 7.4. For generality and true comparison with the $Q \neq 0$ cases, we will, instead, numerically solve the differential equations themselves for all cases. We will, however, make use of the methods outlined in Section 7.4 to determine time step size for our solution. For each case, we will start with both Gaussian and quartic initial pdf's, the Gaussian for simplicity and verification, the quartic for generality. In addition we shall, make comparisons with the standard EKF results when appropriate.

As can be seen from the plots for $\mathbf{f}(x) = 1$ (Figure 7.5.2, 7.5.3), we get the expected results: a linear drift in mean for the Gaussian, and linear drift in peak locations for the quartic pdf. For this case we should expect no change in the basic shape or width of the initial pdf. This characteristic is supported by the figures. As an added check we note that the KF has identical results in the mean and $\dot{P} = 0$, thus preserving the initial shape.

Similarly, $\mathbf{f}(x) = \pm x$ (Figure 7.5.4, 7.5.5, 7.5.6, and 7.5.7) leads to just as reasonable results, with the pdfs retaining their initial shape and only the width or 'variance' changing exponentially with time; increasing for $\mathbf{f}(x) = x$ and decreasing for $\mathbf{f}(x) = -x$. Again these results are identical to those of the KF.

For $\mathbf{f}(x) = 1, \pm x$ note the simplifications in the equations for $\underline{\alpha}$ (Equation 7.3.6) and the lack of dependence on b. In fact, for these two cases we can find the exact solution to the Fokker-Planck equation for the quartic pdf at hand, i.e. given the initial pdf is described by a quartic pdf, the true pdf described by the Fokker-Planck equation will remain in the space of quartic pdf's. Substituting $\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{f}(x;\underline{\alpha}(t))}$ into the Fokker-Planck equation and solving for $\dot{\underline{\alpha}}$ we find

$$\begin{pmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \\ \dot{\alpha}_4 \end{pmatrix} = \begin{pmatrix} f_1 & f_0 & 0 & 0 & 0 \\ 0 & f_1 & 2f_0 & 0 & 0 \\ 0 & 0 & 2f_1 & 3f_0 & 0 \\ 0 & 0 & 0 & 3f_1 & 4f_0 \\ 0 & 0 & 0 & 0 & 4f_1 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} \qquad (7.5.7)$$

From Equation 7.3.6 with $Q = 0$ we find an exact match when $f_2 = 0, f_3 = 0, f_4 = 0$ covering $\mathbf{f}(x) = 1, \pm x$. Hence we find the very encouraging result that our filter yields the exact filter for linear dynamics and quartic pdf's. It does not take much insight into the derivation of Equation 7.5.7 to see that the exact solutions can be

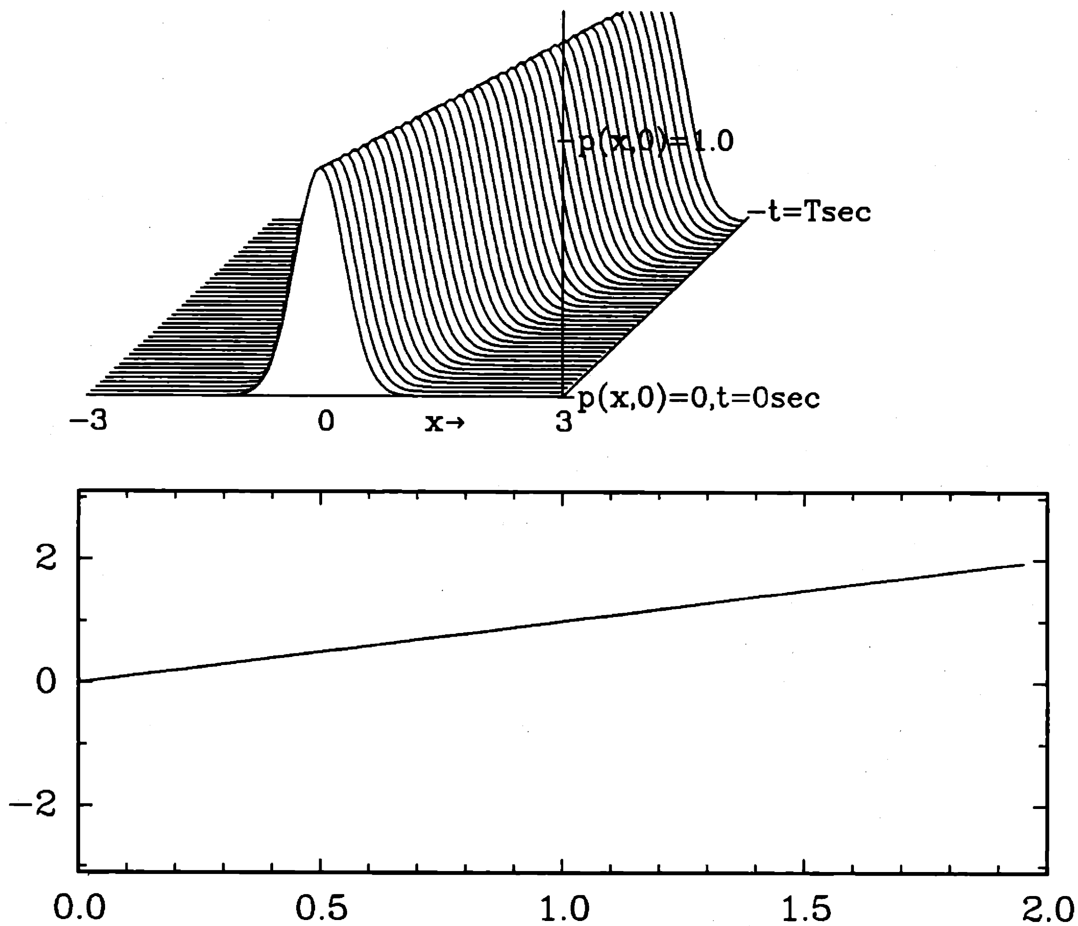


Figure 7.5.2: $\mathbf{f}(x) = 1 \quad Q = 0 \quad p(x,0) \sim N(0, 0.1) \quad b = 1 \quad T = 2sec$

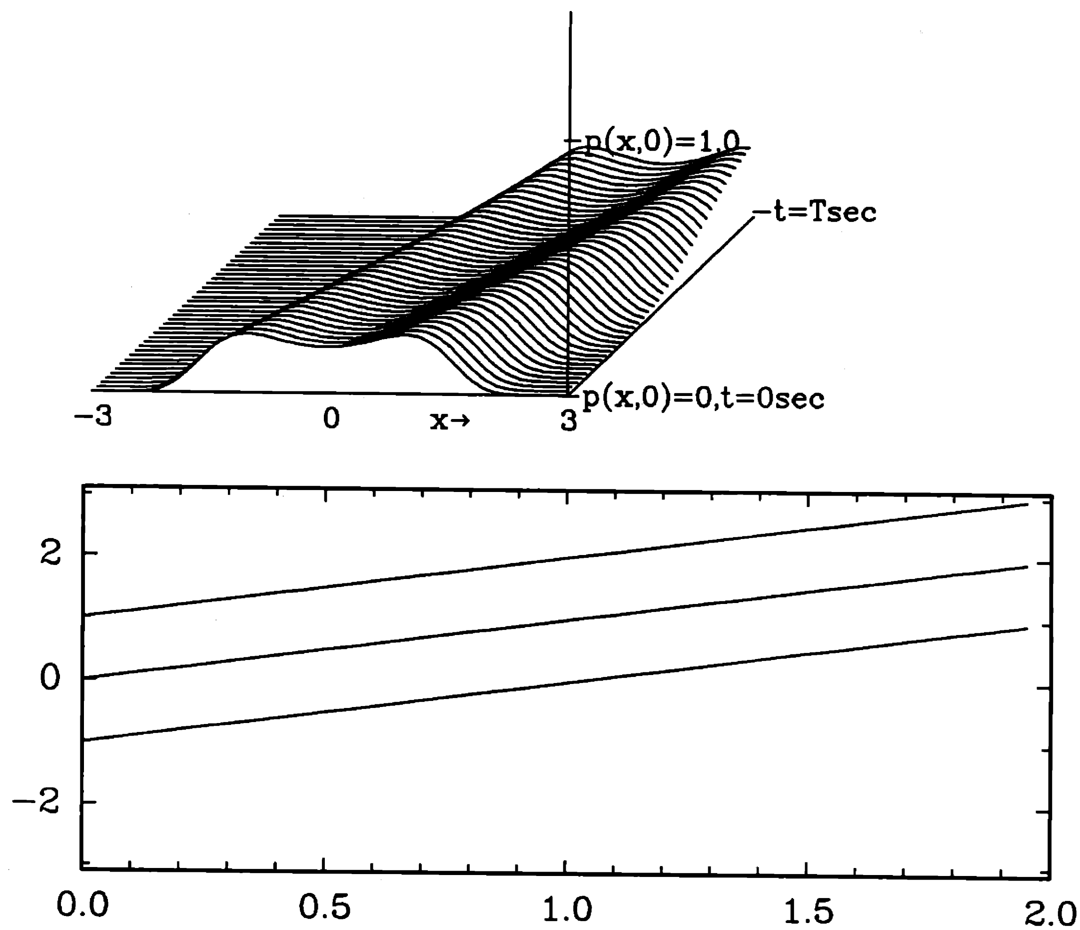


Figure 7.5.3: $\mathbf{f}(x) = 1 \quad Q = 0 \quad p(x,0)$ ~quartic(-1,0,1) $b = 1 \quad T = 2sec$

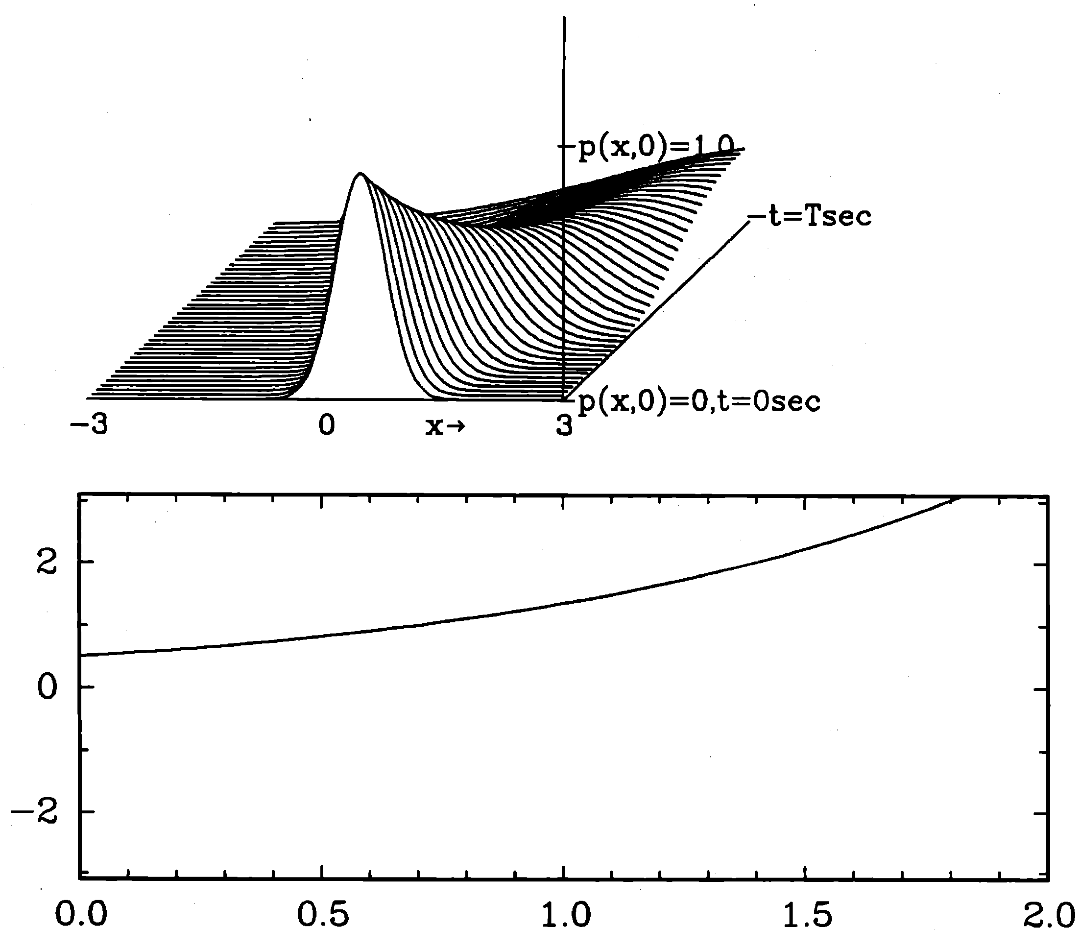


Figure 7.5.4: $\mathbf{f}(x) = x \quad Q = 0 \quad p(x,0) \sim N(0.5, 0.1) \quad b = 1 \quad T = 2sec$

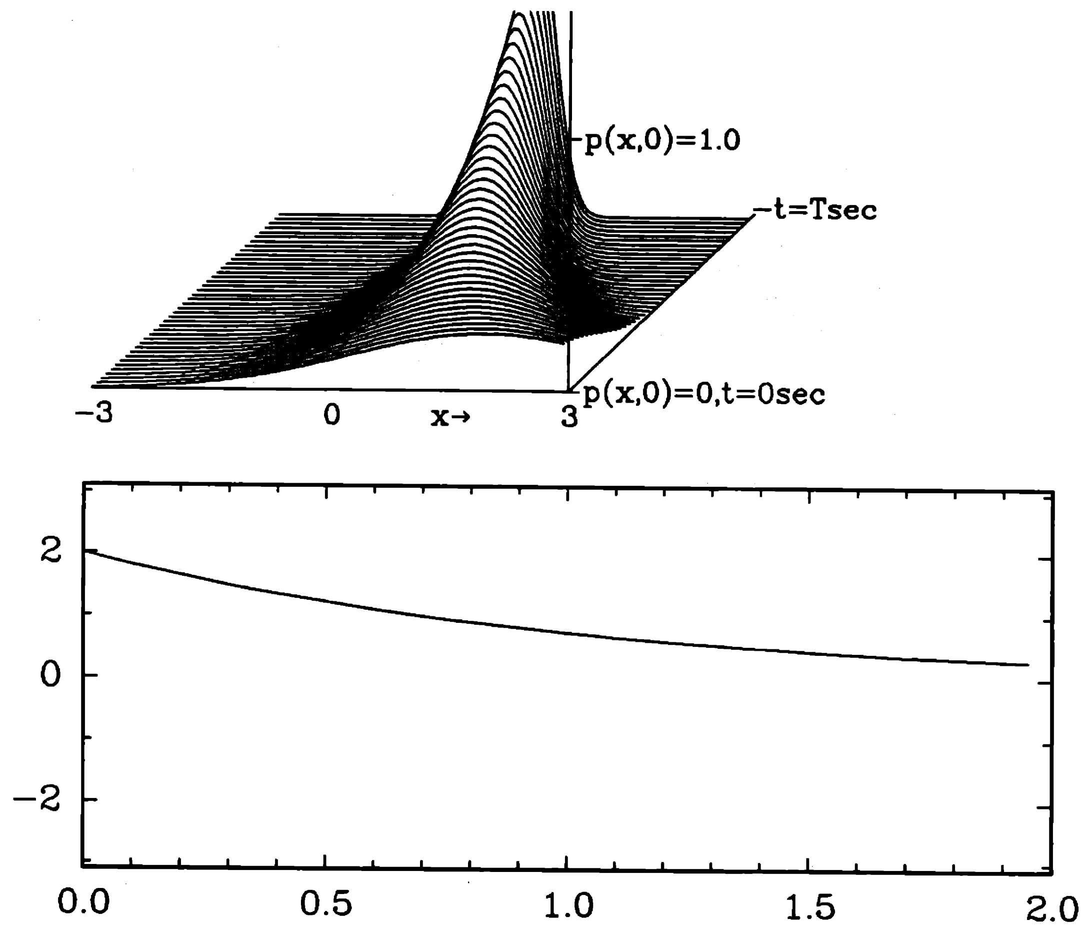


Figure 7.5.5: $\mathbf{f}(x) = -x \quad Q = 0 \quad p(x,0) \sim N(2,3) \quad b = 1 \quad T = 2sec$

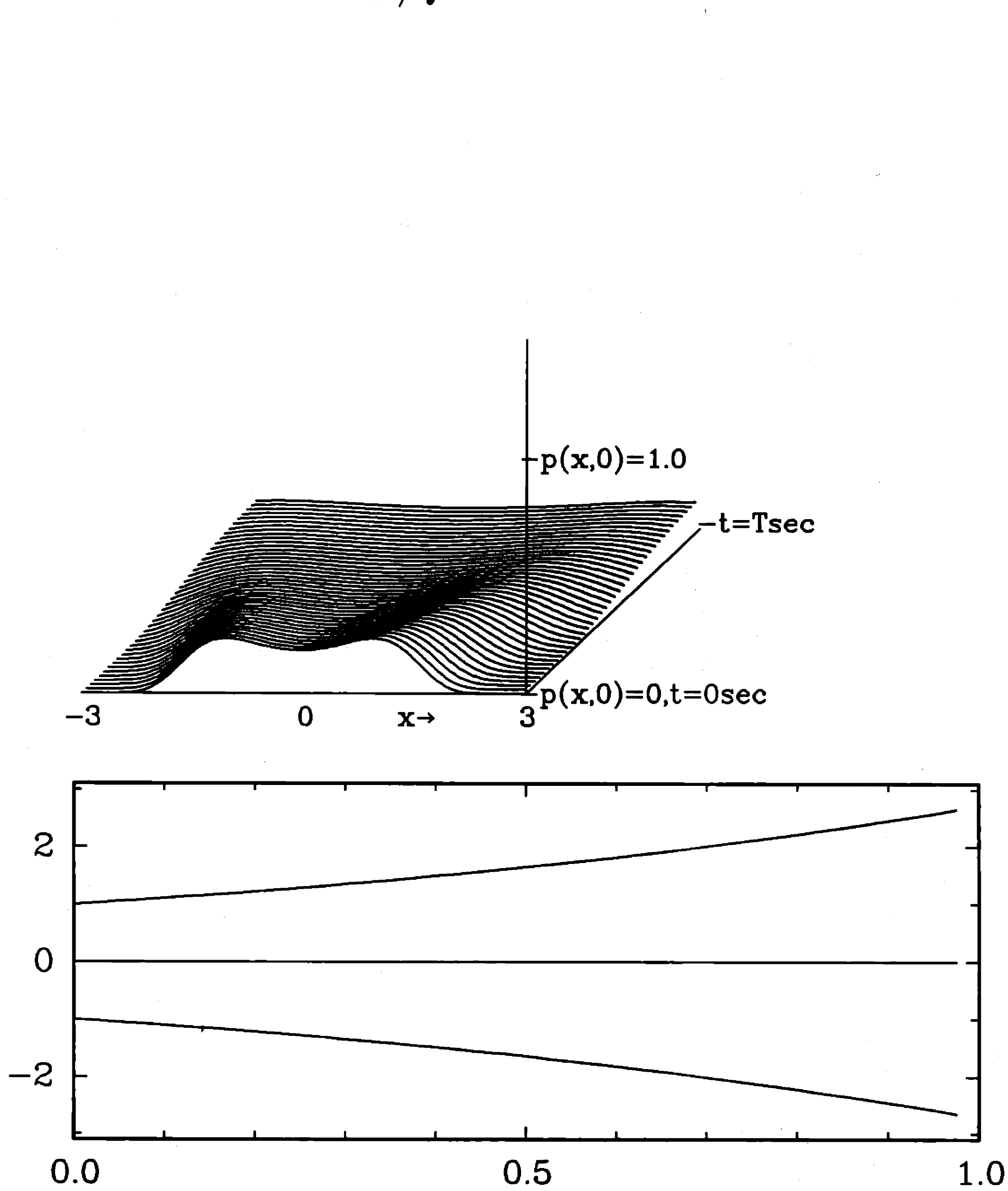


Figure 7.5.6: $\mathbf{f}(x) = x$ $\quad Q = 0$ $\quad p(x,0)$ ~quartic(-1,0,1) $b = 1$ $\quad T = 1sec$

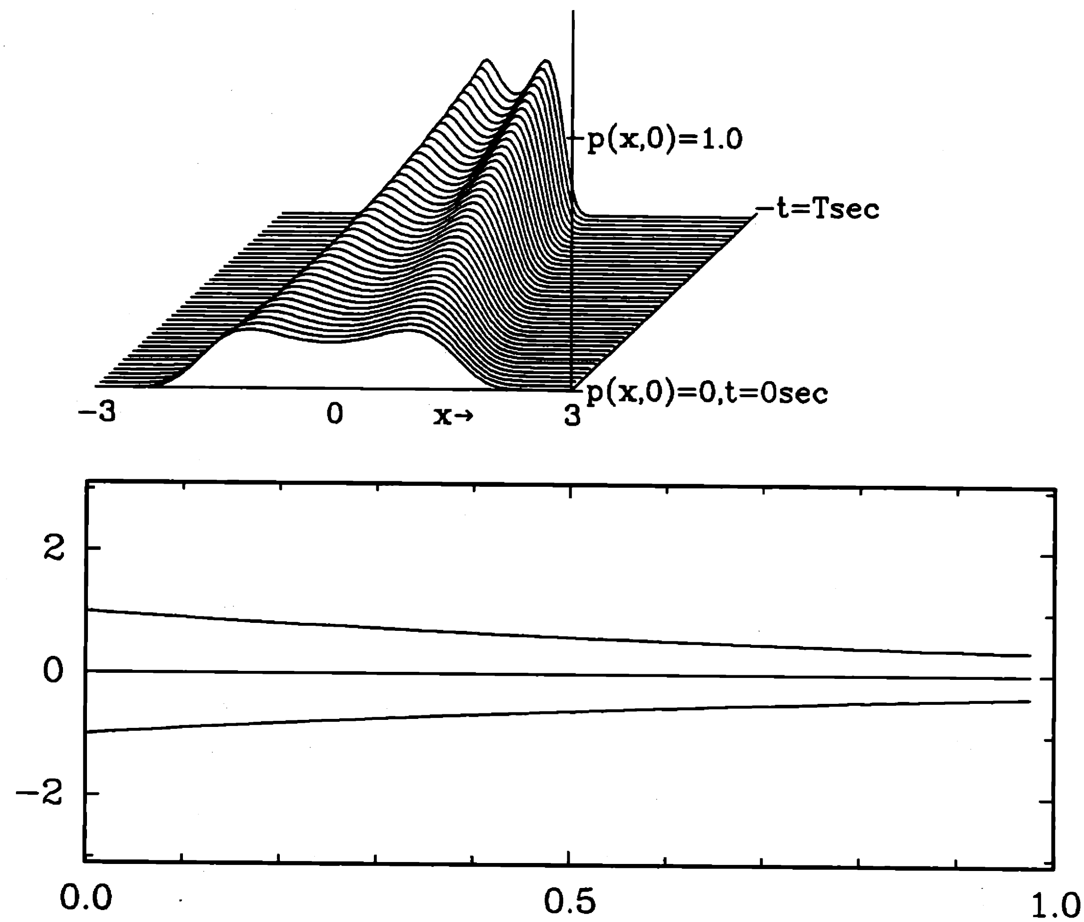


Figure 7.5.7: $\mathbf{f}(x) = -x$ $Q = 0$ $p(x,0)$ ~quartic(-1,0,1) $b = 1$ $T = 1sec$

obtained in the same way for 5th, 6th and higher order approximations. As long as the dynamics remain linear and process noise is set to zero, the exact evolution of the parametric pdf can be found. Although it is impractical to assume models with no process noise, such parameterizations could accept relatively complex polynomial measurement functions ($\mathbf{h}(x)$) at each update step while keeping calculations trivial.

Looking at the results of $\mathbf{f}(x) = x^2$ (Figures 7.5.8,7.5.9), we find the following interpretation. Since $\mathbf{f}(x)$ is always positive, we find the general trend of the pdf is to move right. However, as the negative portion of the pdf approaches zero so does $\dot{x}$. Hence we see a 'bunching up' of probability from the left, close to zero but never quite reaching zero. On the other hand, the positive portion of the pdf should have no such constraints and we see most of the probability mass heading toward large x. However, another bunching phenomenon occurs at some point $x > 2$. This, as it turns out, can be attributed to the choice of the b parameter. Intuitively, for the case shown here ($b = 1$), any pdf characteristics much above $2\sqrt{b}$ (corresponding to the 2-σ point on $N(0, b)$) would be ignored due to the weighting $N(x; 0, b)$. Hence the farthest the probability mass is considered in the approximation is around $2\sqrt{b}$. This intuitive argument can be supported by the following analysis. First we note that for the Gaussian initial condition, the lower plot of Figure 7.5.8 clearly shows the peak-splitting phenomenon necessary to account for the single modal to bi-modal pdf transition characterisitic of x^2 dynamics as descibed above. This indicates that our algorithm should not have any difficulty handling other models which could cause peak splitting and recombination.

From Section 7.4 we know that, for $Q = 0$, we should be able to solve for the α's directly. Doing so for $f_2 = 1$ with $f_i = 0$ for $i = 0, 1, 3, 4$ and keeping only the dominant terms for $t >> 0$, we have the following result for $\underline{\alpha}$

$$\underline{\alpha}(t) \approx \begin{pmatrix} - \\ -0.712369b^{\frac{3}{2}} \\ 0.0658807b \\ 0.4623583b^{\frac{1}{2}} \\ -0.1283807 \end{pmatrix} e^{10.813\sqrt{b}t} \tag{7.5.8}$$

To associate this result with Figure 7.5.8 more closely, we solve for the zeros of $\hat{\varsigma}_x$

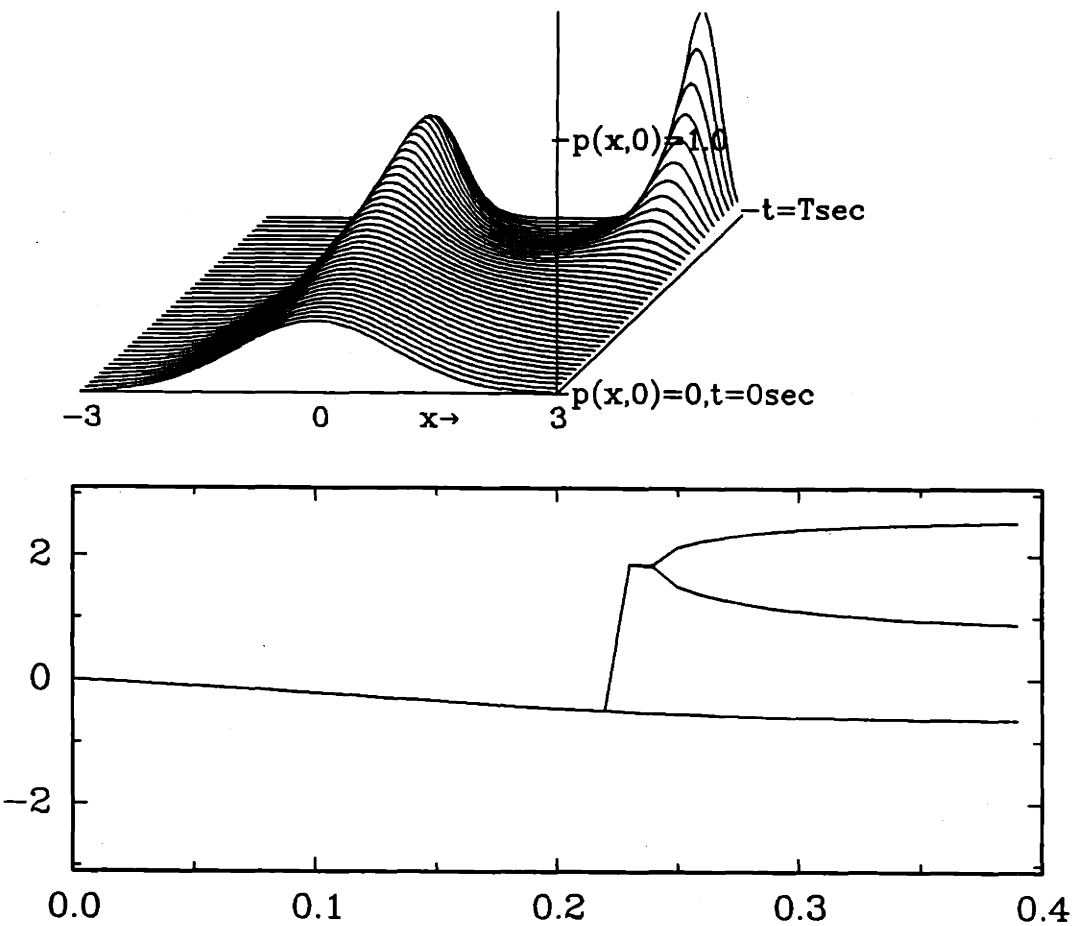


Figure 7.5.8: $\mathbf{f}(x) = x^2$ $Q = 0$ $p(x,0) \sim N(0,1)$ $b = 1$ $T = 0.4sec$

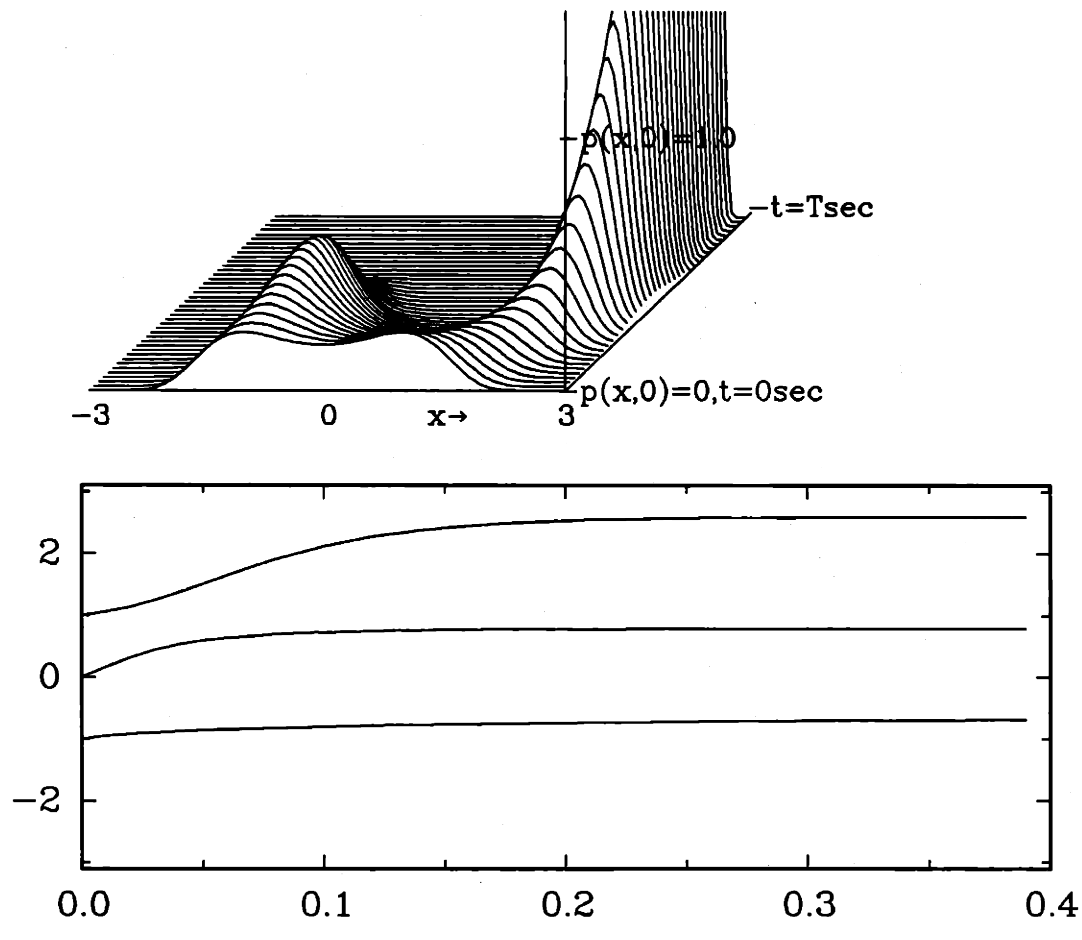


Figure 7.5.9: $\mathbf{f}(x) = x^2 \quad Q = 0 \quad p(x, 0)$ ~quartic(-1,0,1) $b = 1 \quad T = 0.4sec$

corresponding to the above α's. This should clearly identify the peak and valley locations for our pdf.

$$\hat{\varsigma}_x = 0 \rightarrow x = (2.5938288, 0.7869081, -0.6794612)\sqrt{b} \tag{7.5.9}$$

This result does indeed verify the locations of the 'bunchings' for our example. The narrowing widths of the peaks can be explained by the $e^{10.813\sqrt{b}t}$ term scaling the α's and hence the exponent of our pdf. Hence, Equation 7.5.9 suggests that we should choose b such that $2.59\sqrt{b}$ is larger than the largest excursion we expect for x as limited by external conditions such as measurement updates. The dynamics in Figure 7.5.8 are inherently unstable, and therefore there is no choice of b which would lead to the correct behavior for our pdf, i.e., for the probability mass on the right in Figure 7.5.8 to continue to go to the right as $t \rightarrow \infty$. This does not eliminate the usefulness of this result, since in many applications the range of interest is limited.

Comparing these results with the EKF for the same model, $\dot{m} = m^2, \dot{P} = 4mP$, we see a similar but only one sided behavior. If $m(0) > 0$, we see the pdf spread and drift to the right. If $m(0) < 0$, we see the bunching from the left of $x = 0$, but none of the spreading effects. Hence, we find, as expected, that the EKF does not capture the multi-modal nature of this example. In fact, for this particular example ($m(0) = 0$) the EKF fails completely by predicting no change in either the mean or variance of its pdf.

As expected, a reversed but otherwise identical behavior is obtained for $\mathbf{f}(x) = -x^2$. The change to Equation 7.5.9 involves only a change in sign.

As it turns out, the same behavior as well as the same dominant peak locations result for the quartic initial pdf case of Figure 7.5.9 for $t >> 0$, thus the same $2.59\sqrt{b}$ rule holds.

In $\mathbf{f}(x) = x^4$ (Figures 7.5.10, 7.5.11) we display the pdf evolution for $\mathbf{f}(x) = x^4$. Due to the nature of $\mathbf{f}(x)$ we expect the pdf to have a similar behavior to $\mathbf{f}(x) = x^2$. We see from Figures 7.5.10 and 7.5.11 that this is true. The similarities appear to extend all the way to the peak positions. An analysis identical to the one performed

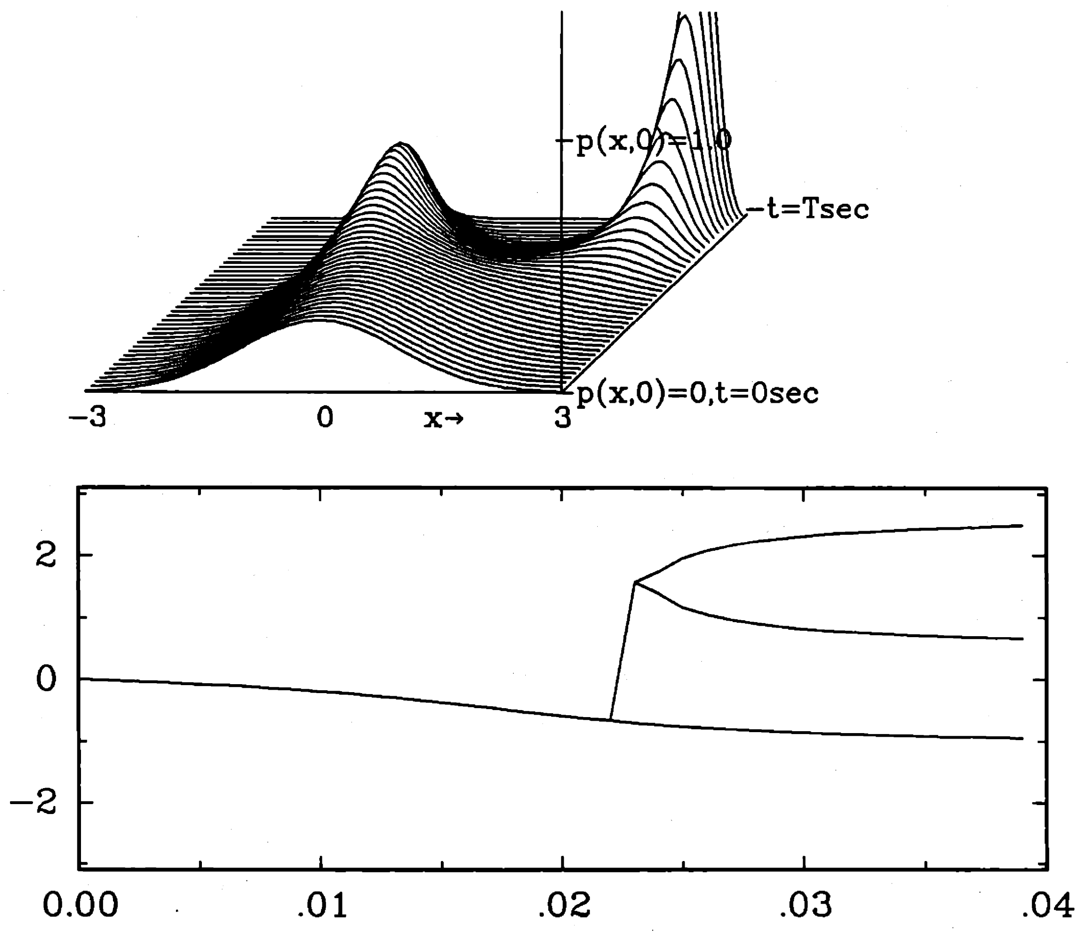


Figure 7.5.10: $\mathbf{f}(x) = x^4 \quad Q = 0 \quad p(x,0) \sim N(0,1) \quad b = 1 \quad T = 0.04sec$

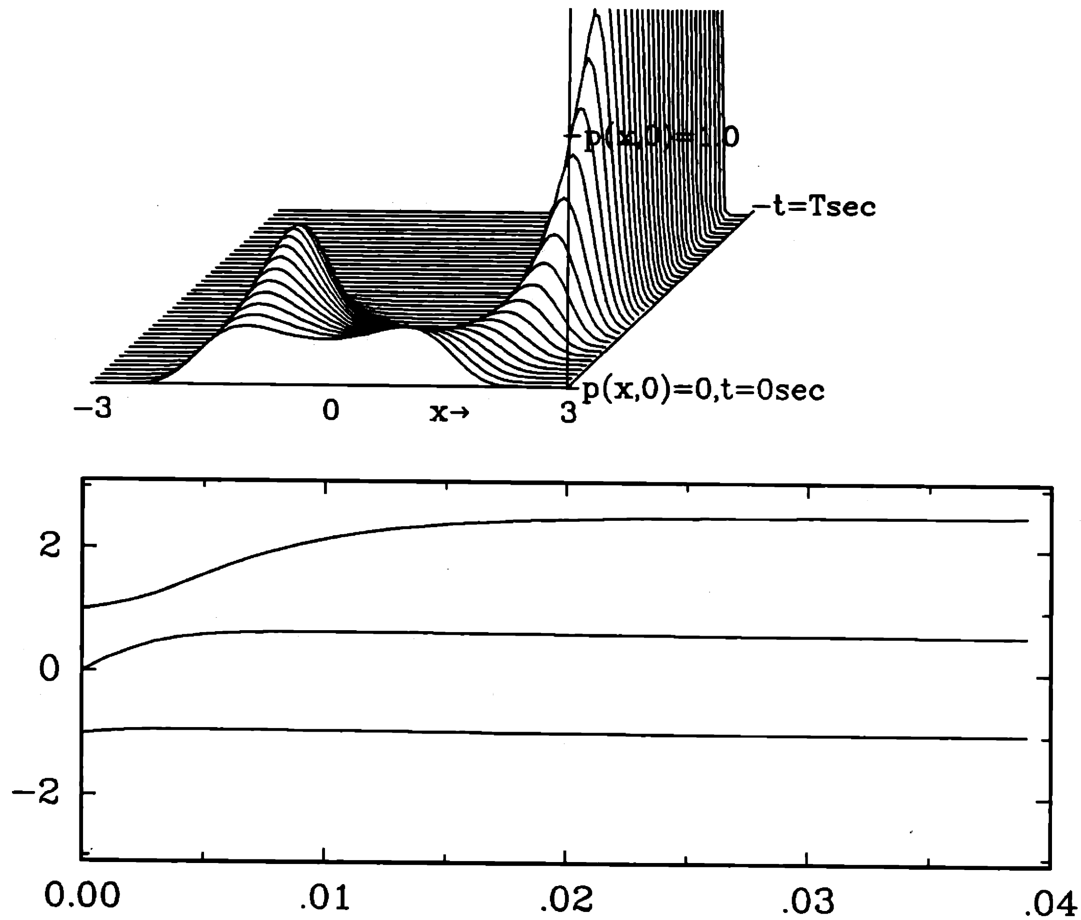


Figure 7.5.11: $f(x) = x^4$ $Q = 0$ $p(x, 0)$ ~quartic(-1,0,1) $b = 1$ $T = 0.04sec$

on $\mathbf{f}(x) = x^2$ of the dominant terms in $\underline{\alpha}$ results in the following for $t >> 0$

$$\underline{\alpha} \approx \begin{pmatrix} - \\ -0.0857120136^{\frac{1}{2}} \\ 0.04476635 \\ 0.0410477876^{-\frac{1}{2}} \\ \frac{-0.014229699}{b} \end{pmatrix} e^{123.7861b^{\frac{3}{2}}t} \tag{7.5.10}$$

which gives rise to zeros in $\hat{\varsigma}_x$ at

$$\hat{\varsigma}_x = 0 \rightarrow \quad x = (-0.985092, 0.5997431, 2.548838)\sqrt{b} \tag{7.5.11}$$

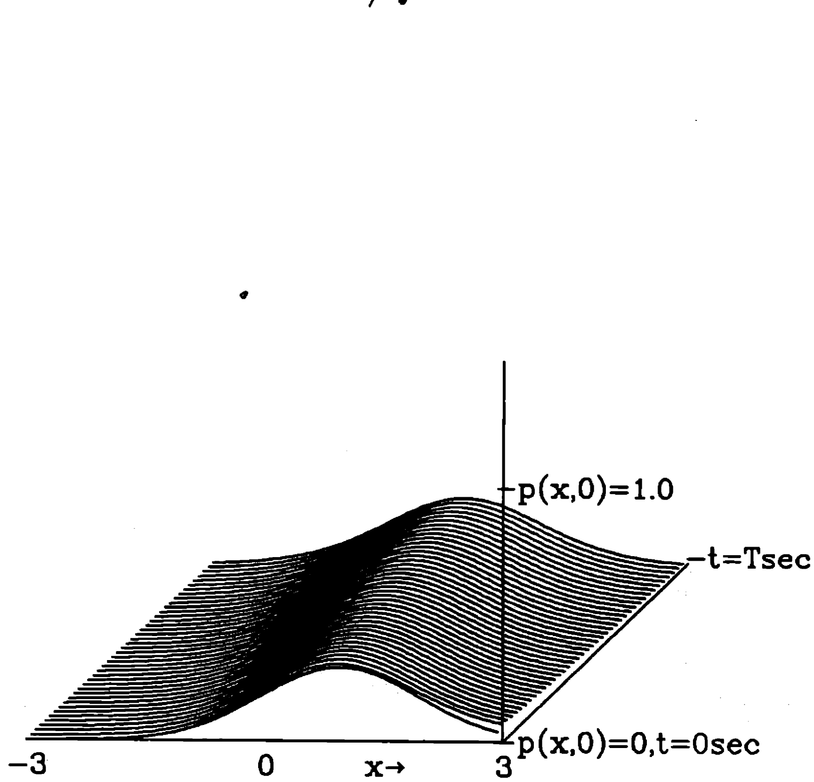
which are indeed approximately the same results as in $\mathbf{f}(x) = x^2$. The only major difference between the two cases is the rate at which the 'bunching' occurs as evidenced by the large exponent in Equation 7.5.10.

Again when we refer to the EKF results for this model, $\dot{m} = m^4, \dot{P} = 16m^3P$, we find exactly the same behavior and interpretation with the EKF describing only half of the true pdf's behavior.

Next we study the case for $\mathbf{f}(x) = x^3$ (Figures 7.5.12, 7.5.13, 7.5.14, and 7.5.15). This case turns out to result in the most problems for our approximation technique. For this example we should expect the positive half of the pdf to head toward the right and the negative half to head toward the left similar to $\mathbf{f}(x) = x$ but at a faster rate. However neither Figure 7.5.12 or 7.5.14 bear this out. The pdf's appear to reach a steady state resembling a Gaussian pdf for either initial condition. Again, since the equations for this case are a function of b, we assume that these anomalies are due to the choice of b. Doing the same type of analysis on the dominant modes of $\underline{\alpha}$ as in $\mathbf{f}(x) = x^2$ we find that for $t >> 0$ a steady state is reached,[1] with the α's given by

$$\underline{\alpha}(t) = \begin{pmatrix} - \\ 0 \\ -\frac{1}{2b} \\ 0 \\ \frac{1}{60b^2} \end{pmatrix} \tag{7.5.12}$$

[1]For $\mathbf{f}(x) = x^3$ the eigenvalues of $\tilde{A}$ from Equation 7.4.2 are negative. Hence we are left with a stable steady state solution.



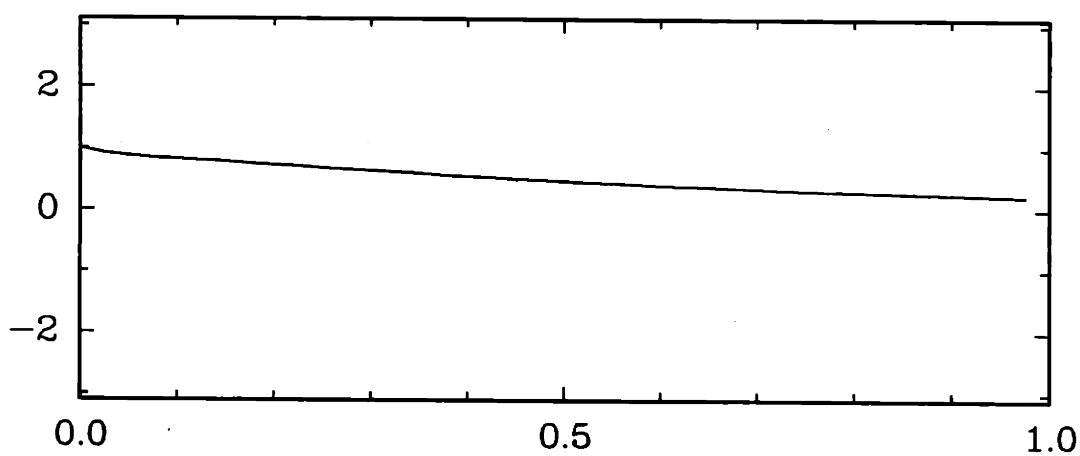


Figure 7.5.12: $\mathbf{f}(x) = x^3 \quad Q = 0 \quad p(x,0) \sim N(1,1) \quad b = 1 \quad T = 1sec$

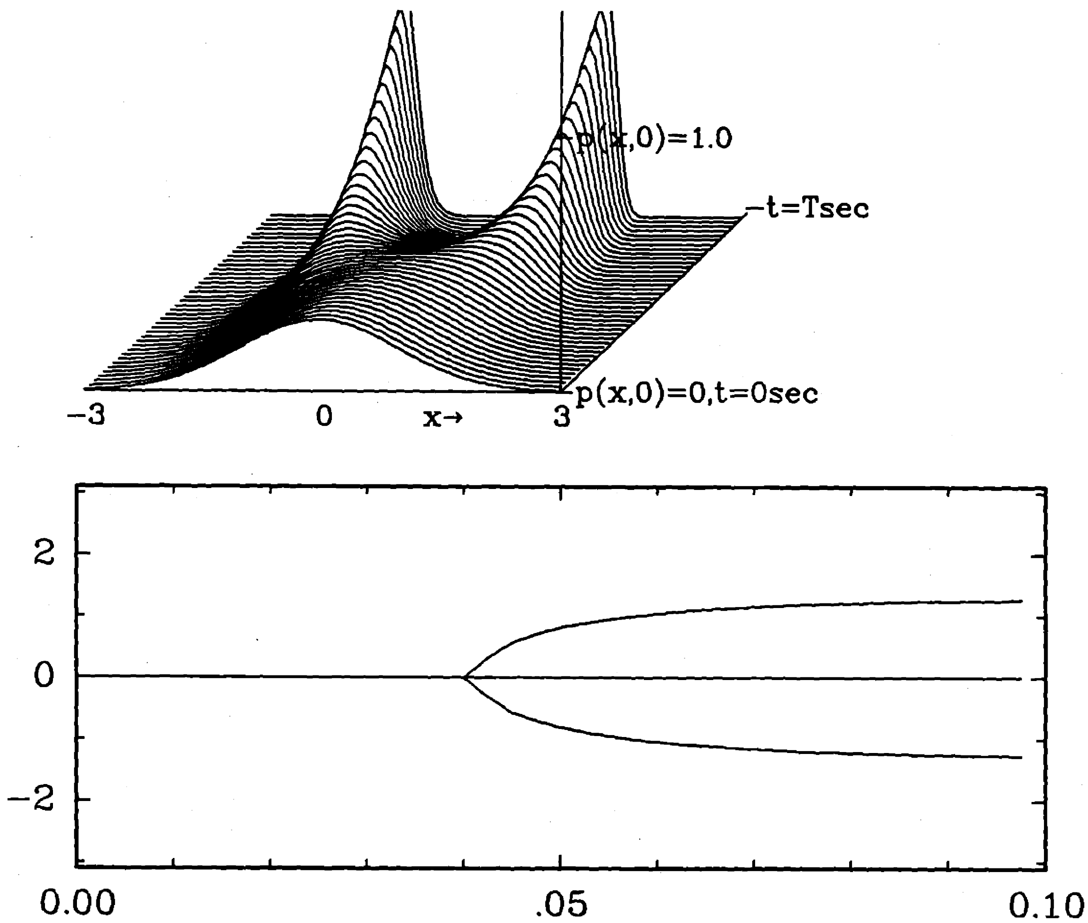
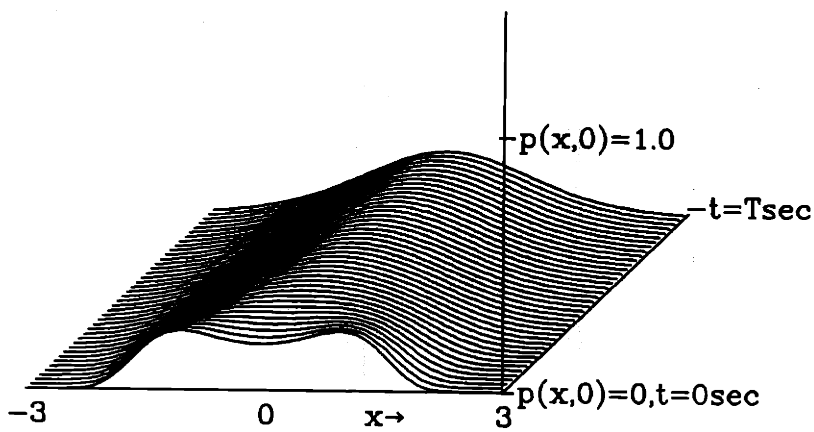


Figure 7.5.13: $\mathbf{f}(x) = -x^3$ $Q = 0$ $p(x,0) \sim N(0,1)$ $b = 1$ $T = 0.1sec$



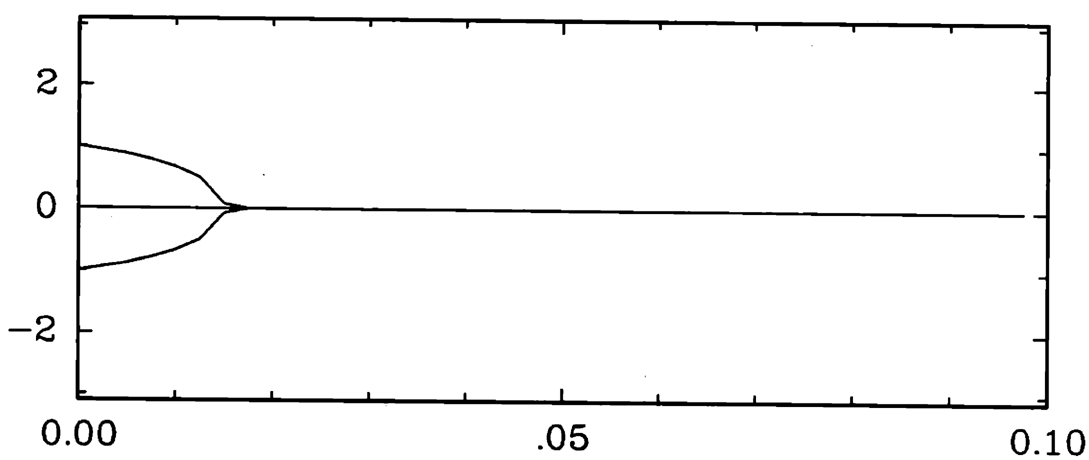


Figure 7.5.14: $\mathbf{f}(x) = x^3$ $Q = 0$ $p(x,0)$ ~quartic(-1,0,1) $b = 1$ $T = 0.1sec$

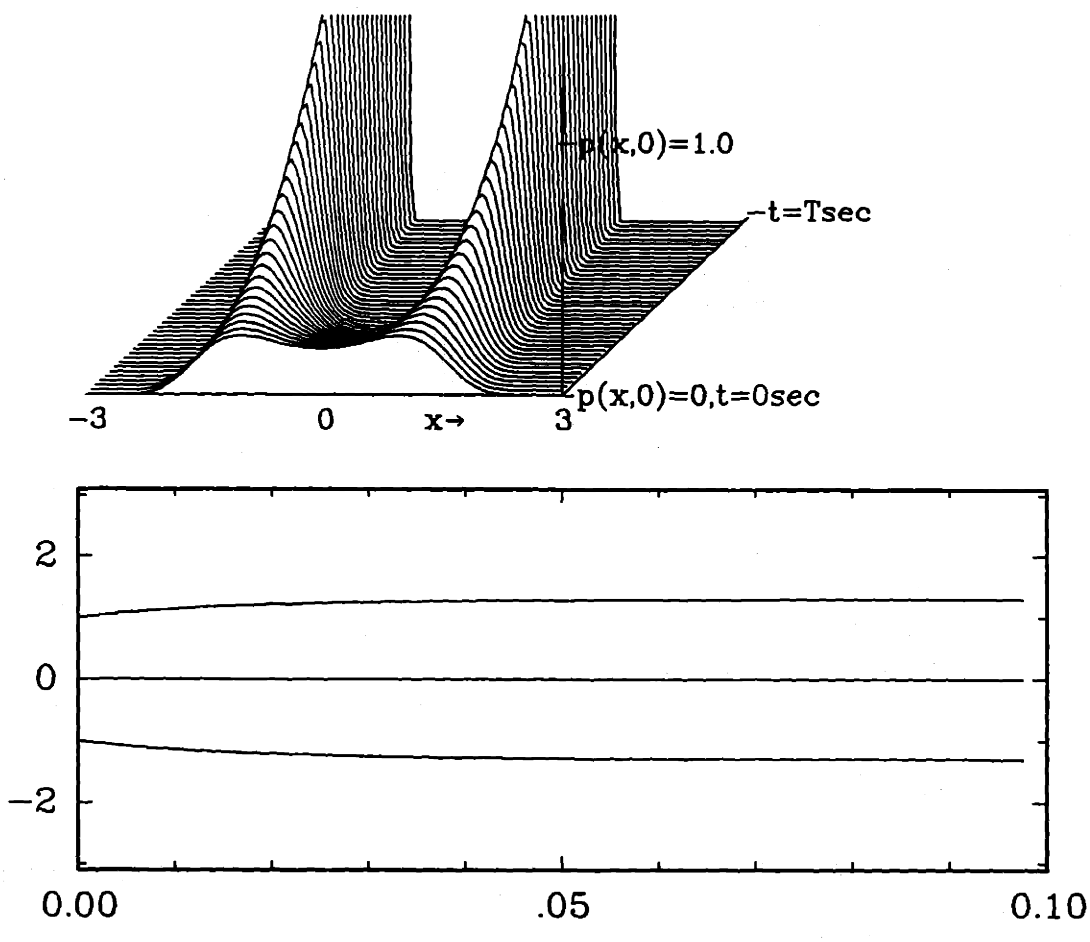


Figure 7.5.15: $\mathbf{f}(x) = -x^3$ $Q = 0$ $p(x,0)$ ~quartic(-1,0,1) $b = 1$ $T = 0.1sec$

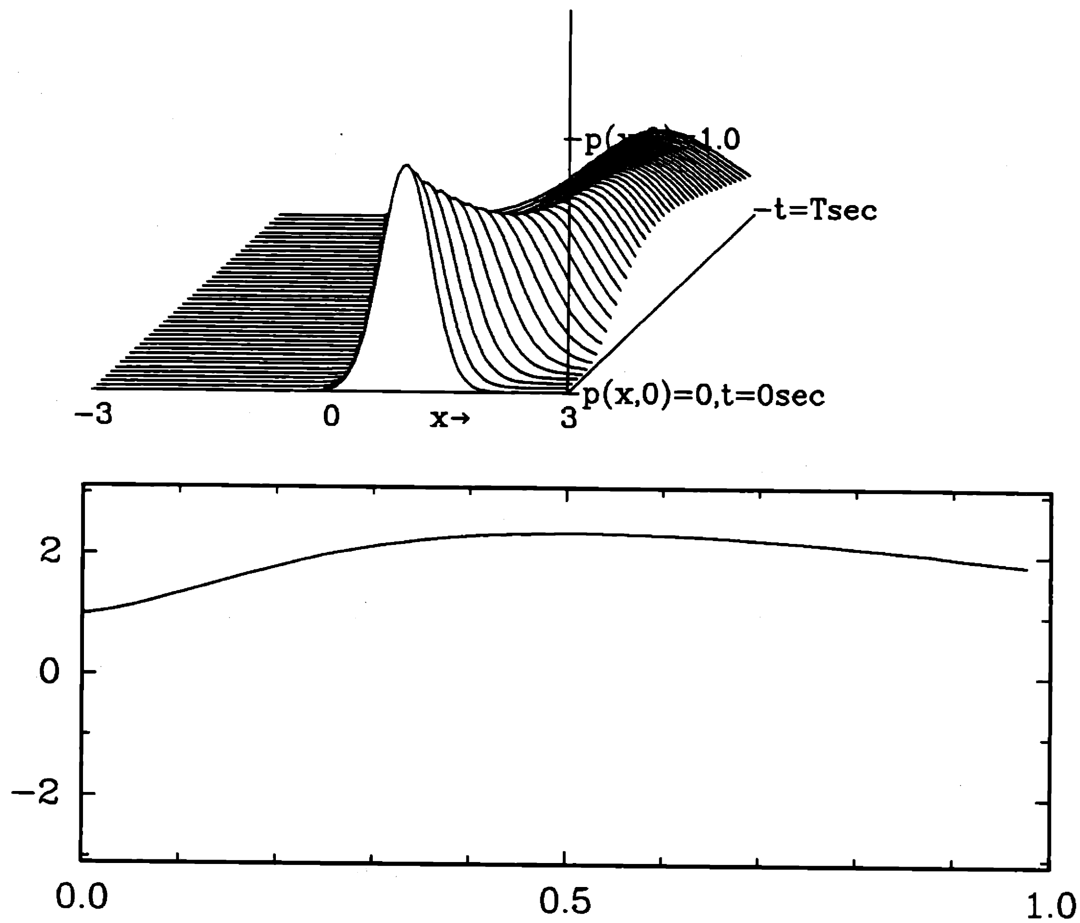


Figure 7.5.16: $\mathrm{f}(x) = x^3 \quad Q = 0 \quad p(x,0) \sim N(1, 0.1) \quad b = 1 \quad T = 1sec$

Note that for b sufficiently large (so that we may consider $\alpha_4 = 0$) the α's correspond to the coefficients of a Gaussian pdf with zero mean and variance b. Hence the steady state 'variance' of out filter equals b. Stated another way, b is the upper limit for the variance our filter can describe. This result bears out the results in Figures 7.5.12 and 7.5.14 whose steady state pdf's are $\approx N(0, b)$.

To further support the Gaussian limit behavior, we plot in Figure 7.5.16 a situation where the initial 'variance' is much less than b ($b = 1$ for our examples) and plot its evolution. This plot does indeed show the proper behavior up to the $N(0, b)$ limit.

With respect to the EKF results, $\dot{m} = m^3, \dot{P} = 6m^2P$, we see that, the EKF captures the basic behavior of the state model for $\|m(0)\| >> 0$, but fails completely when $\|m(0)\|$ is small. It can be shown that, except for the limiting behavior due to the weighting, our approximating pdf will broaden with time even if $\|m(0)\| = 0$. Of course the multi-modal pdf cannot be propagated directly by the EKF and hence there is no comparison. For this, the work of Chapter 6 may show promise.

Next we study the result for $\mathbf{f}(x) = -x^3$ (Figures 7.5.13,7.5.15). This form of dynamics is extremely stable as seen in the example in Chapter 5 Section 5. In this case we expect both negative and positive probability masses to head toward zero thus resulting in an impulse at $x = 0$ as $t \to \infty$. Referring to Figure 7.5.13 and 7.5.15, however, we see a totally different behavior.

For the Gaussian initial case, instead of the narrowing behavior which should occur, we find that the pdf splits into two humps then quickly become peaked at two points. The same steady state behavior is exhibited for the quartic initial pdf case. This steady state behavior can again be explained using a dominant eigenvalue analysis. However, the fact that the pdf mass does not tend to an impulse at zero requires deeper study. First we shall explain the steady state behavior by looking at the α's.

With $f_3 = -1$ and $f_i = 0$ for $i = 0, 1, 2, 4$ we find the eigenvalues of the matrix of Equation 7.3.6 to be

$$(1.5836, 28.4164, 6.7621, 53.2379)b \tag{7.5.13}$$

Keeping the largest eigenvalue and setting all others to zero, we solve Equation 7.3.6 with $Q = 0$ for $\underline{\alpha}(t)$ at $t >> 0$. This results in

$$\underline{\alpha}(t) \approx \begin{pmatrix} - \\ 0 \\ \frac{-0.008198895}{b} \\ 0 \\ \frac{-0.0024249547}{b^2} \end{pmatrix} e^{53.2379bt}. \tag{7.5.14}$$

Solving $\hat{\varsigma}_x = 0$ to find the peak and valley locations of $\hat{\varsigma}$ we find

$$x = (-1.3, 0, 1.3)\sqrt{b} \tag{7.5.15}$$

(which is not a function of t) corresponding to the steady state peak locations in our figures. This verifies that the odd nature of the pdf's of Figures 7.5.13 and 7.5.15 is not a result of numerical sensitivity errors in implementing $F\dot{\underline{\alpha}} = g$ but one inherent to our approximation scheme.

Conceptually, our approach is to propagate the approximating density, at some time t, one infinitesimal time step forward using the Fokker-Planck equation. Since this takes us off our parametric pdf's manifold, we must project the result back onto our manifold to retain closure. The projection step is where the actual approximation takes place. Hence a Fokker-Planck propagated pdf which has many if not all of its basis orthogonal to the parametric pdf manifold will yield a poor approximation.

To formally determine if this is the cause of our poor results we must develop an error measure between the propagated pdf and the projected pdf. This will be discussed in more detail in the following chapter. Here, however we shall simply examine the change due to the Fokker-Planck equation and the same change projected on our parametric manifold and see if they 'make sense'. Plotted in Figures 7.5.17, 7.5.18, 7.5.19 are the approximating pdf, the pdf derivative propagated off the manifold (desired direction), and the same pdf derivative projected back onto the manifold (actual direction), respectively. Due to symmetry, we examine only the right half of $\hat{p}$ and $\frac{\partial \hat{p}}{\partial t}$. We see from the figures a definite loss due to the projection step, since the propagated pdf derivative shows a peak at a point closer to zero than the projected pdf derivative, thus indicating that the true pdf evolves towards the origin faster than the approximating pdf. Upon closer examination of $\frac{\partial \hat{p}}{\partial t}$ (in the lower plots), we see that before projection, $\frac{\partial \hat{p}}{\partial t}$ shows a strong tendanncy to remove mass from the right side of $\hat{p}$. This feature is removed by the projection.

Up to this point we have considered the special case of $Q = 0$ and have discovered problems for the case of $\mathbf{f}(x) = -x^3$. Next we shall consider the inclusion of process noise in a few examples. We will begin with the $\mathbf{f}(x) = -x^3 + v$ case with $Q \neq 0$. Doing this resulted in the propagated-projected-pdf plots shown in Figures 7.5.20, 7.5.21, and 7.5.22. As can be seen, the $\mathbf{f}(x) = -x^3$ filter now behaves as would be expected for $-x^3$ dynamics. This is due to the effect $Q > 0$ has on the projection

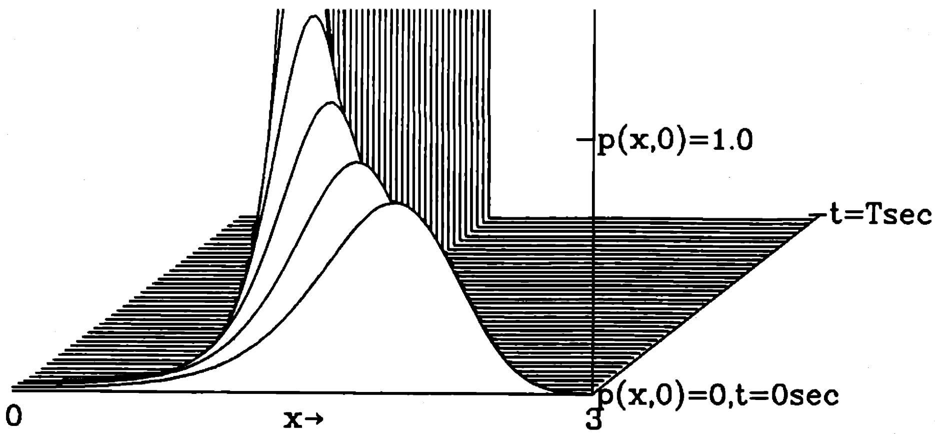


Figure 7.5.17: Plot of approximating pdf. $Q = 0$ $p(x,0)$ ~quartic(-2,0,2) $T = 0.5sec$

step as evidenced in Figures 7.5.21 and 7.5.22. Note the small amount of noise ($Q = 0.001$) needed to make the filter behave properly.

Although the $Q = 0$ results show a definite flaw in our approximation scheme, most real applications will not involve zero process noise, and hence we still feel justified in presenting our scheme as an alternative to other filtering methods. Note that the EKF still performs very poorly in the face of these dynamics with instabilities (see Chapter 5). It is also clear from the above examples that there is a strong need for a companion set of tools to evaluate the errors, so that we may determine whether a particular model leads to poor performance under a given parameterization. We shall derive such a tool in Chapter 8.

Now to determine the effects of the noise itself, we shall look at $\mathbf{f}(x) = 0, Q = 1$ (Figures 7.5.23 and 7.5.24) then study $\mathbf{f}(x) = x^2, Q = 1$ (Figures 7.5.25,7.5.26) to study the effects of noise and $\mathbf{f}(x) = x^2$ dynamics. As can be seen for either initial condition for $\mathbf{f}(x) = 0$, setting $Q = 1$ causes the pdf to diffuse out. For the initial quartic pdf plot (Figure 7.5.24), the initial pdf evolves into a Gaussian form. For the Gaussian initial pdf plot (Figure 7.5.23), it simply increases the variance of the initial Gaussian. In this case, the variance, P, can be shown analytically to vary as $\dot{P} = Q$, hence verifying the accuracy of our scheme once more.

For $\mathbf{f}(x) = x^2, Q = 1$ (Figures 7.5.25, and 7.5.26), we see the smoothing char-

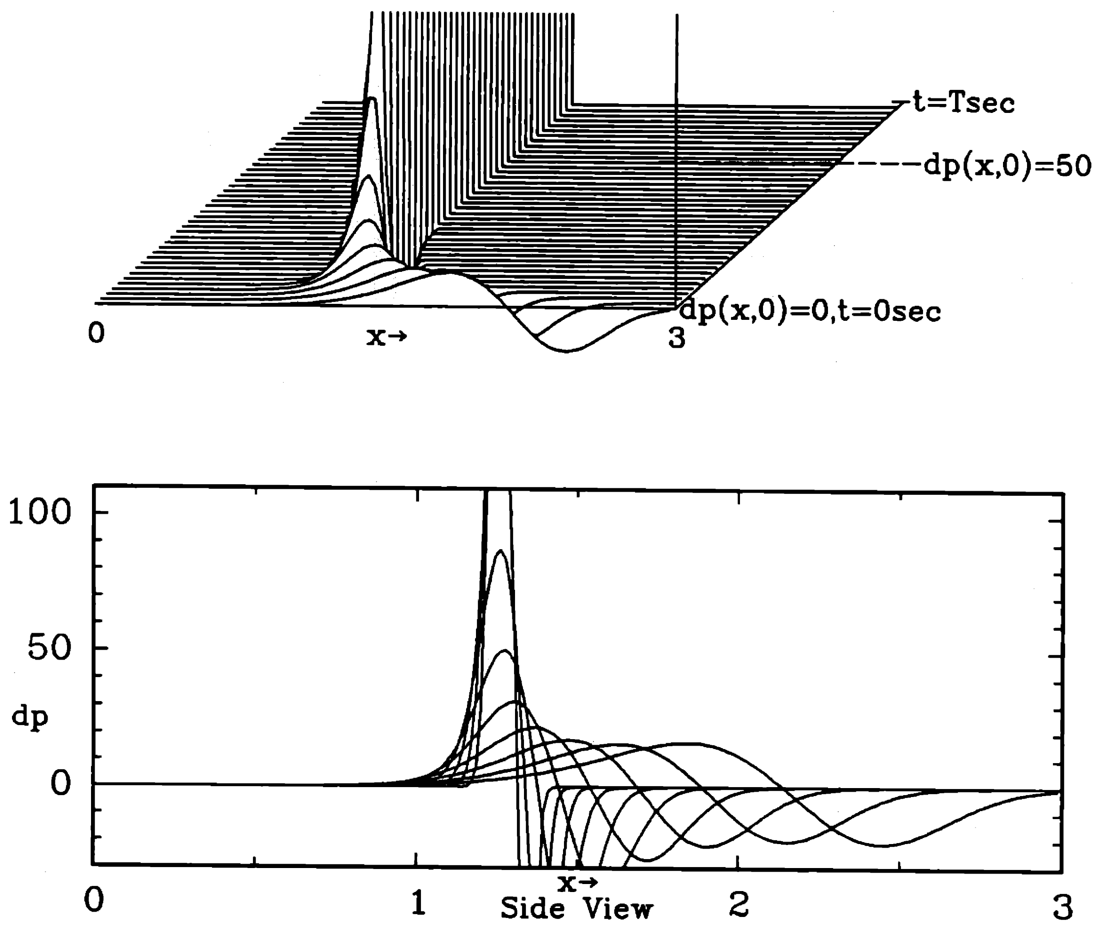


Figure 7.5.18: Plot of $\frac{\partial p}{\partial t}$ using the Fokker-Planck equation

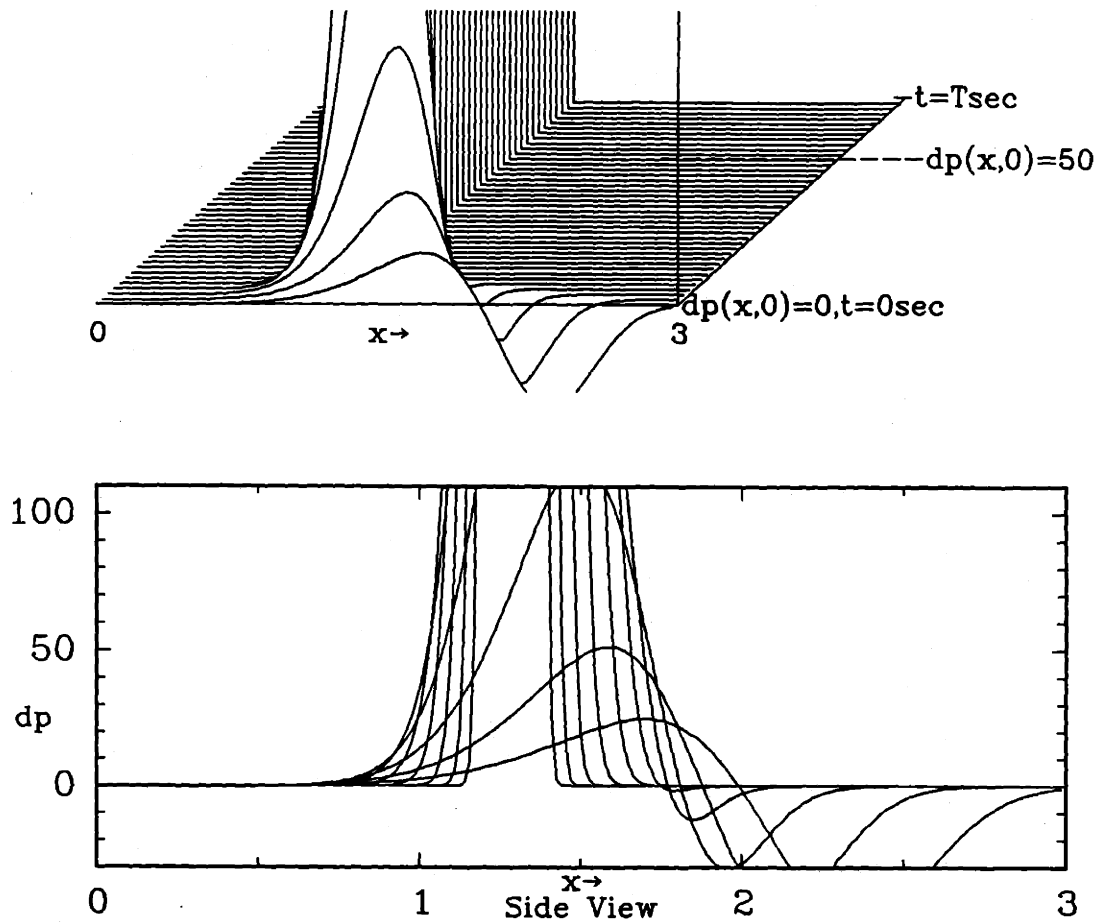


Figure 7.5.19: Plot of $\frac{\partial p}{\partial t}$ projected onto the manifold

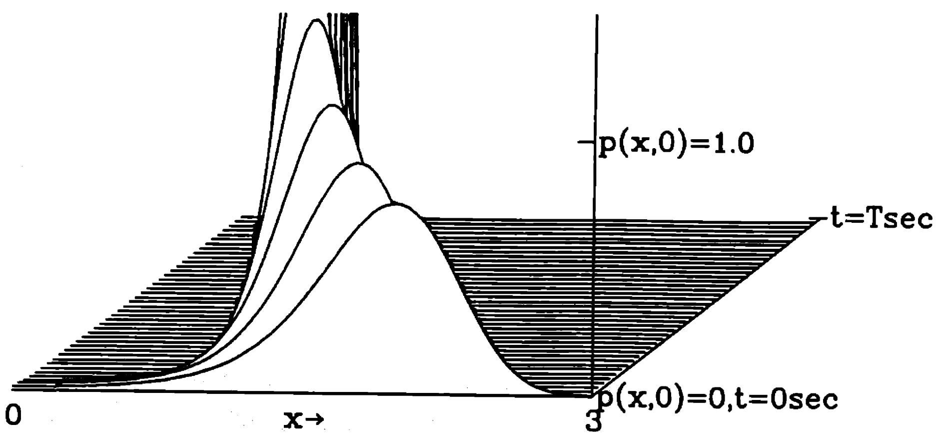


Figure 7.5.20: Plot of approximating pdf. $Q = 0.001$ $p(x,0)$ ~quartic(-2,0,2) $T = 0.5sec$

acteristic applied to the results of $\mathbf{f}(x) = x^2$. Again the pdf behaves with the appropriate bunching about $x < 0$ and spreading for $x > 0$ but, due to the process noise, the minimum widths of the left and right hand peaks is limited. In fact, we find that eventually the positive peak flattens out completely. This is also evidenced from the peak plot in the lower half of Figure 7.5.26 where the valley and upper peak locations combine to form a plateau which heads toward large x, although this limit again appears to be effected by our choice of b.

In the next section we will apply our scheme to an example which includes measurements for direct comparison with the EKF.

7.6 Example

In this section we will present another application of the results derived in this chapter. From Chapter 1 we know that if we consider discrete measurement updates, the application of these updates can be affected by the use of Bayes' rule. In addition, we know that if the form of the approximating conditional pdf includes, as a subset, the form of the pdf describing the measurement, that Bayes' rule can be affected by a simple addition of coefficients. It is the purpose of this section to

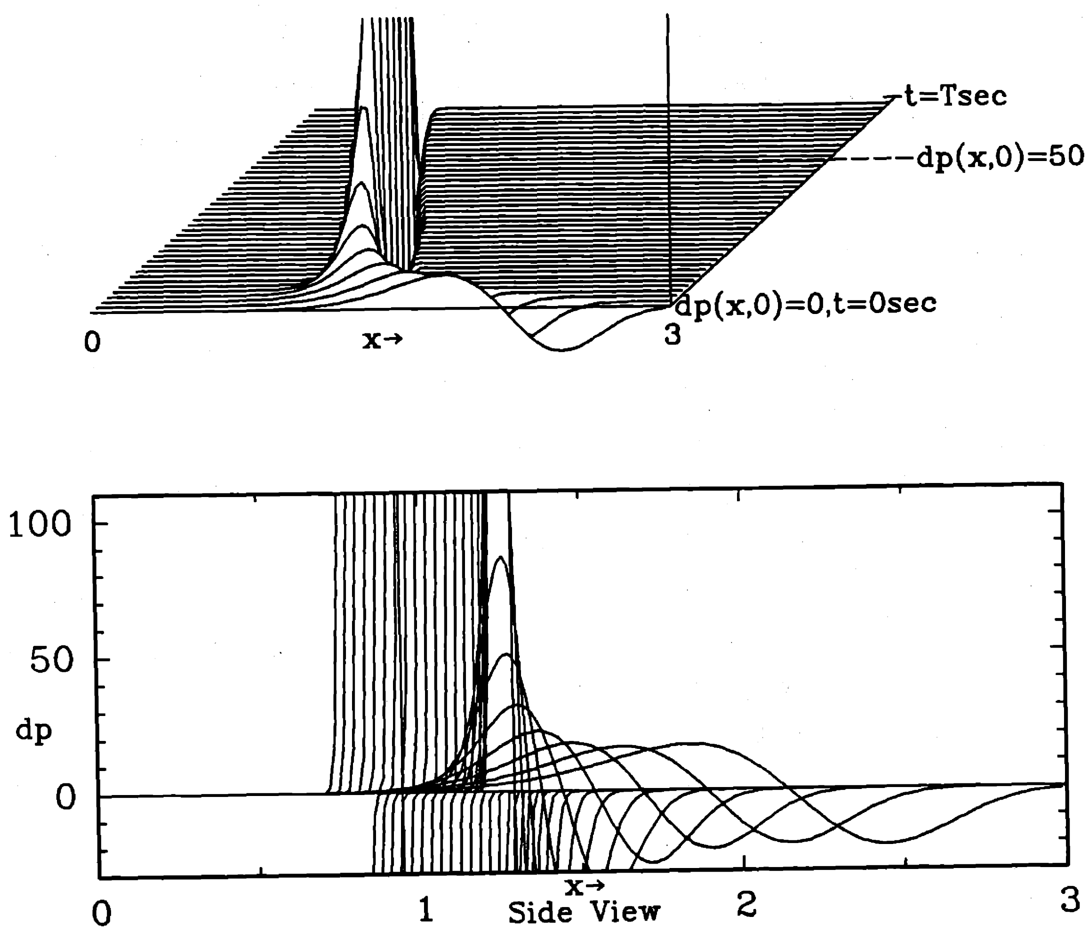


Figure 7.5.21: Plot of $\frac{\partial \hat{p}}{\partial t}$ using the Fokker-Planck equation

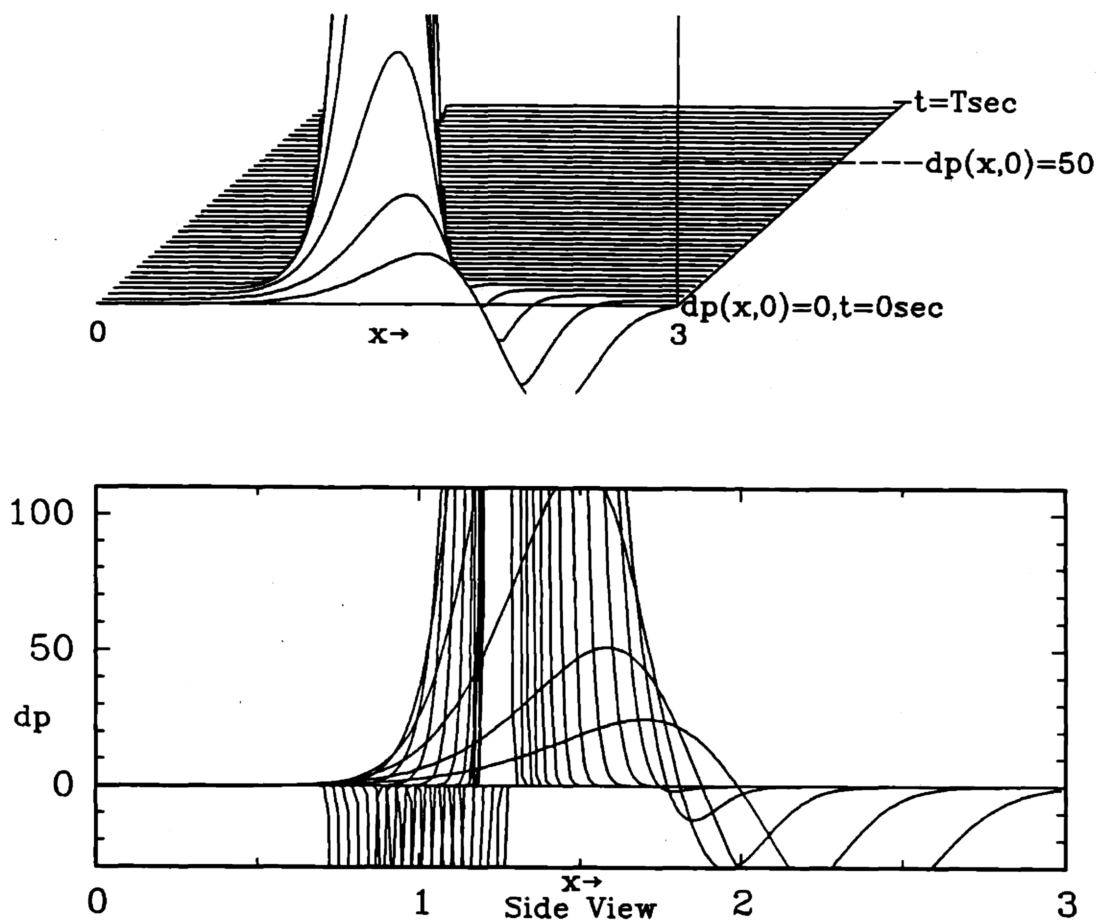


Figure 7.5.22: Plot of $\frac{\partial \hat{p}}{\partial t}$ projected onto the manifold

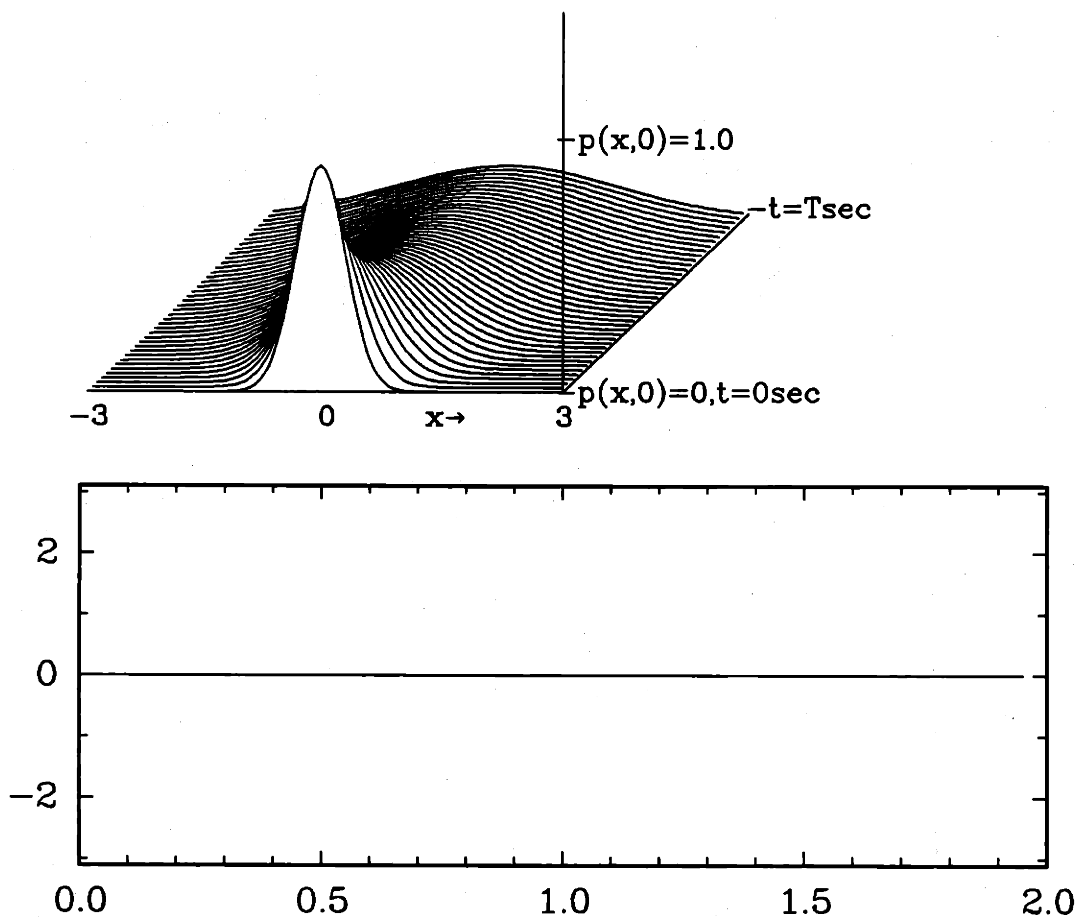


Figure 7.5.23: $\mathbf{f}(x) = 0 \quad Q = 1 \quad p(x,0) \sim N(0,0.1) \quad b = 1 \quad T = 2sec$

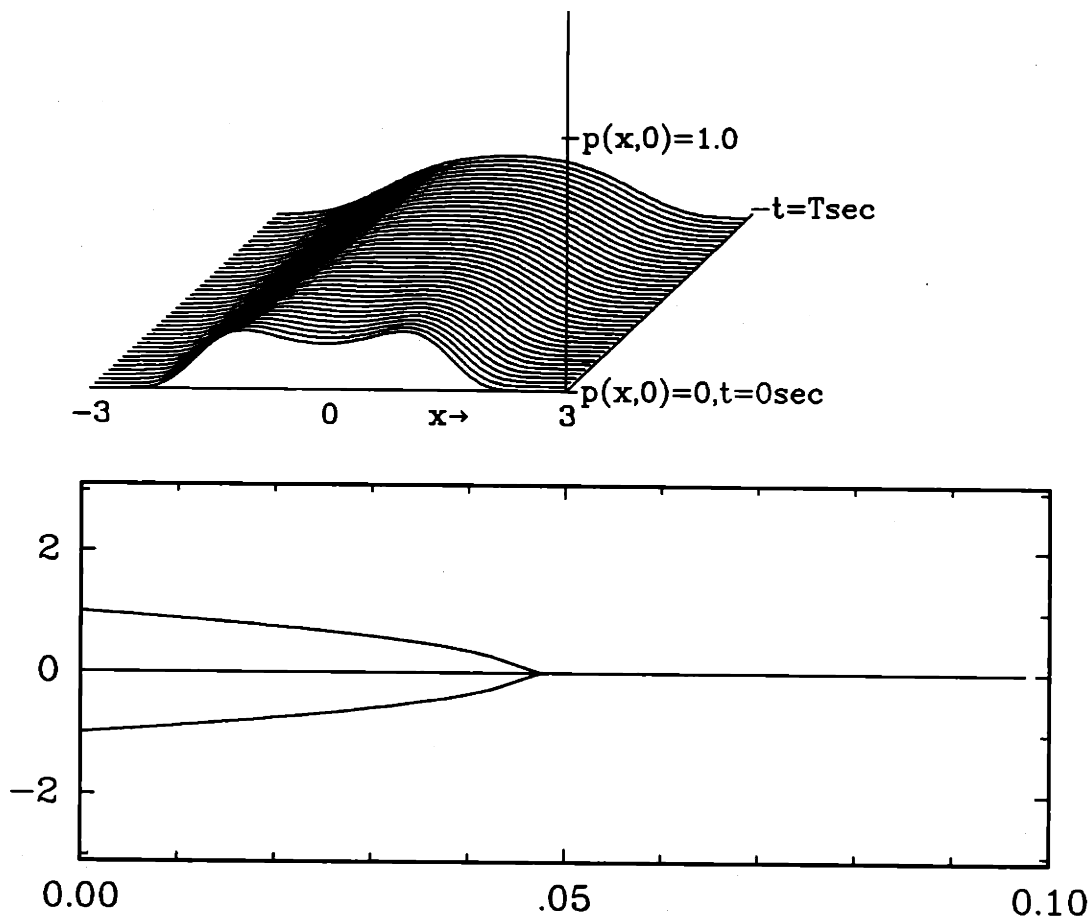


Figure 7.5.24: $\mathbf{f}(x) = 0 \quad Q = 1 \quad p(x,0)$ ~quartic(-1,0,1) $b = 1 \quad T = 0.1sec$

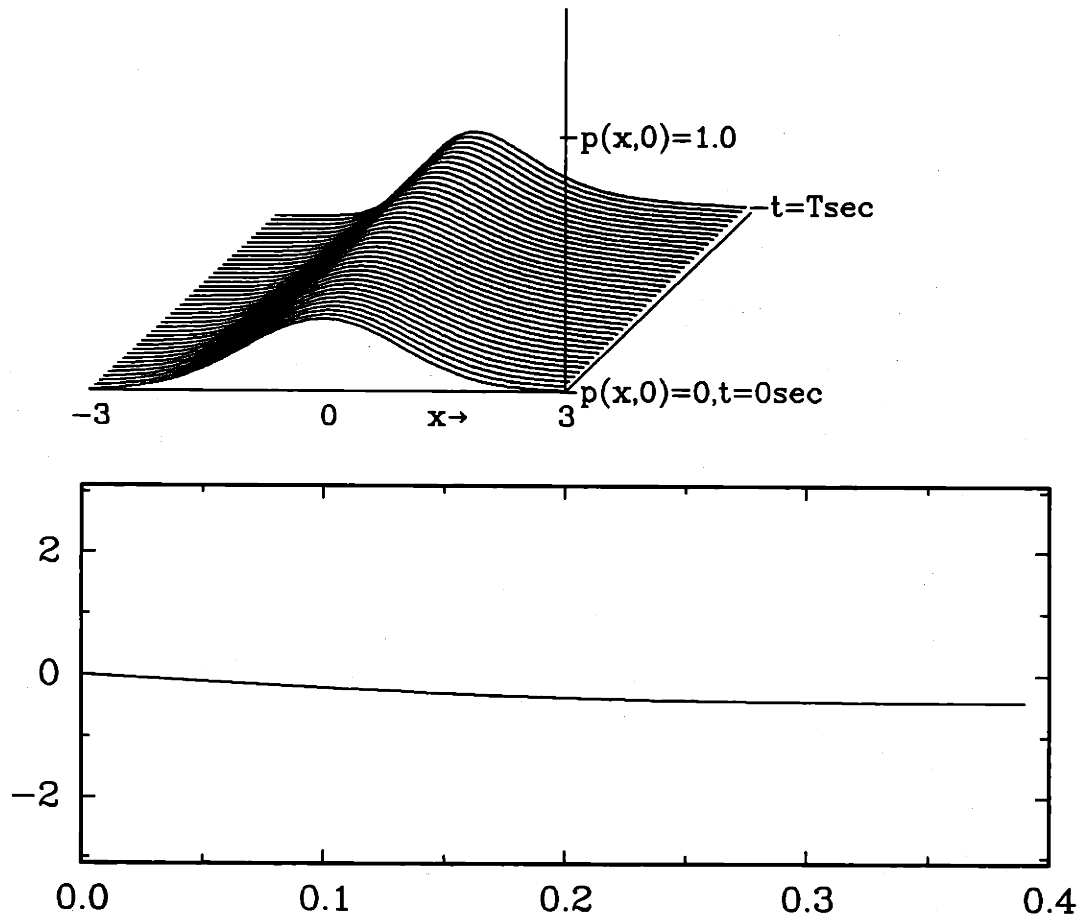
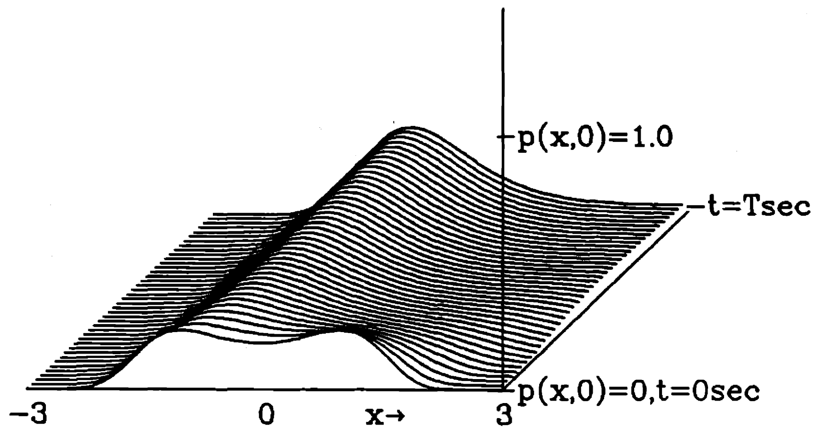


Figure 7.5.25: $\mathbf{f}(x) = x^2$ $Q = 1$ $p(x,0) \sim N(0,1)$ $b = 1$ $T = 0.4sec$



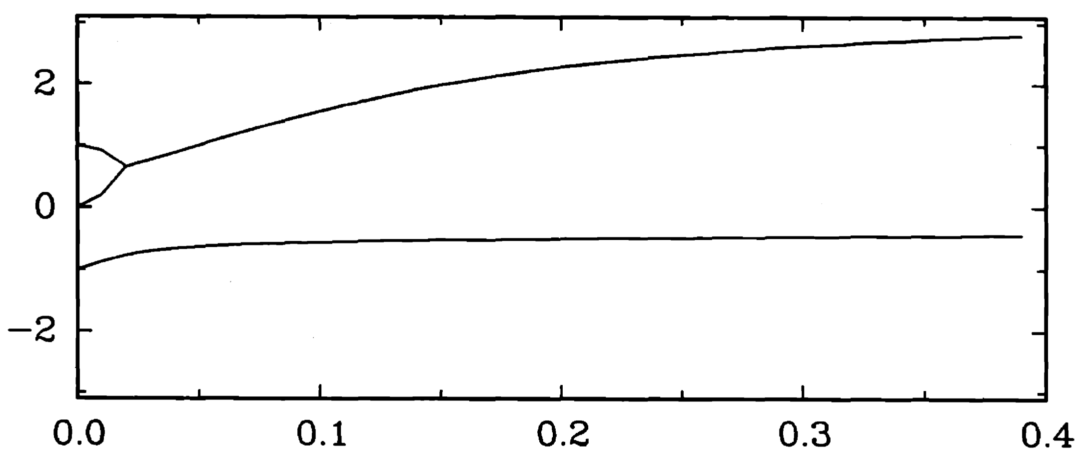


Figure 7.5.26: f$(x) = x^2$ $Q = 1$ $p(x,0)$ ~quartic(-1,0,1) $b = 1$ $T = 0.4sec$

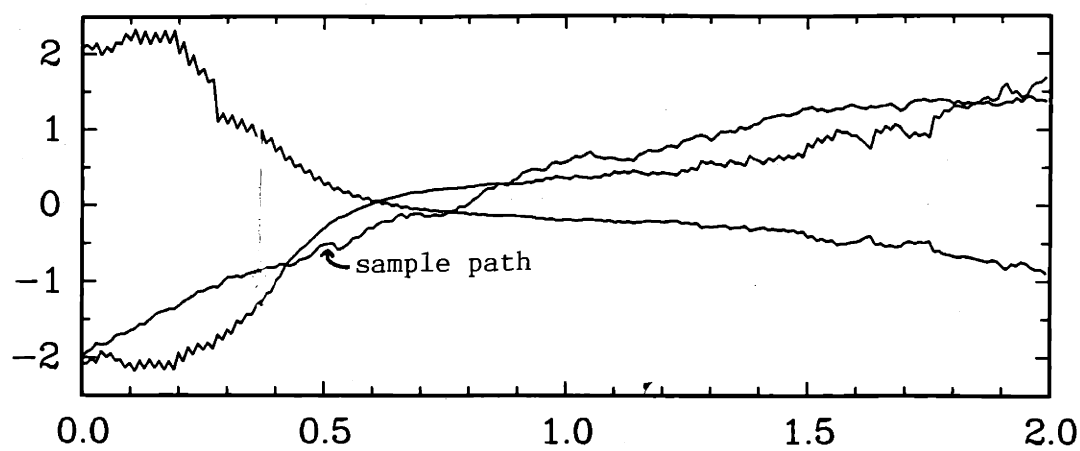


Figure 7.6.27: Peak locations of pdf and sample path for Equation 7.6.2

describe such an example. We choose the following state model for our example:

$$dx = (1-x)dt + dv \tag{7.6.1}$$

$$z_k = hx^2(t_k) + w_k \tag{7.6.2}$$

where dv is a Brownian processes with intensity Q, and w_k is an uncorrelated sequence with unit variance.

The result is an update step which is given by:

$$\alpha_4 = \alpha_4 - \frac{h^2}{2} \tag{7.6.3}$$

$$\alpha_2 = \alpha_2 + hz_k \tag{7.6.4}$$

$$\alpha_0 = \alpha_0 - \frac{z_k^2}{2}. \tag{7.6.5}$$

where α_i are the approximating pdf parameters. Applying these equations along with the results of the previous section we find the results displayed in Figures 7.6.27 and 7.6.28. These figures plot the peak locations (Figure 7.6.27) and corresponding amplitudes (Figure 7.6.28) versus time. For comparison we also plot the results of the standard EKF in Figure 7.6.29. The initial conditions for both examples was a Gaussian pdf with mean 0 and variance 100. The process noise Q and measurment amplitude h were set equal 0.1 and 2.0 respectively.

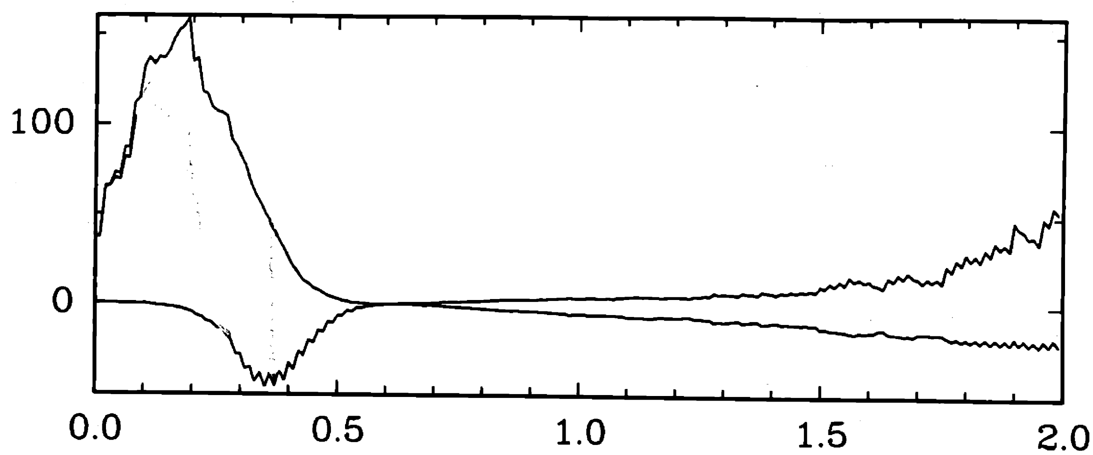


Figure 7.6.28: Amplitudes at the corresponding pdf peak locations

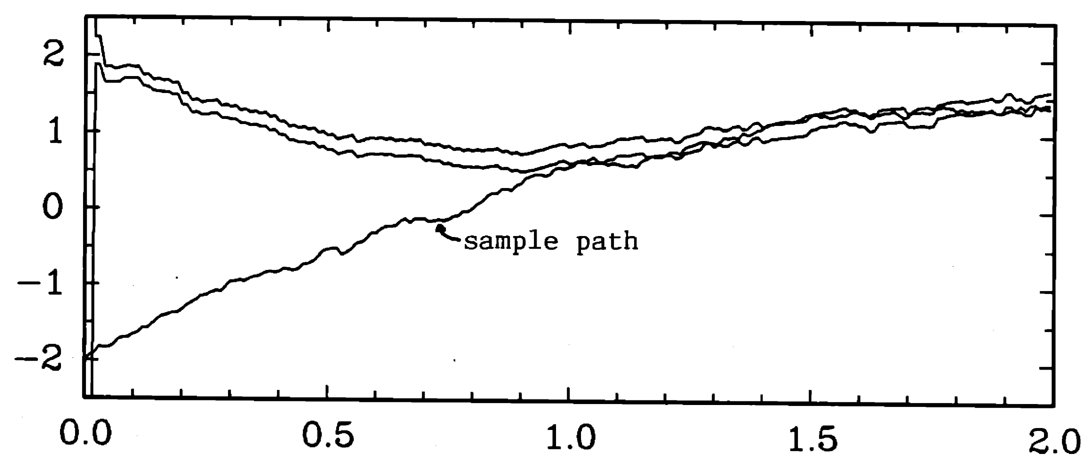


Figure 7.6.29: mean +- $\sqrt{\text{variance}}$ and sample path for the EKF

By studying the model (Equations 7.6.2) we would expect the following behavior. Due to the high signal to noise ratio of our measurement (2/1), we would expect initially for the $z_k = x^2$ measurement term to dominate our estimate of the state. This implies that the pdf should appear double peaked at first and symetrical about zero. This is precisely what we see in Figure 7.6.27 about $t = 0$ where the initial single Gaussian peak has split into two points symetrical about zero.

As time progresses, we should expect that the dynamical portion of the model ($\dot{x} = 1 - x$) should start taking effect by modifying the pdf to place more emphasis on the positive portion of the pdf, which should concentrate about $x = 1$. We see exactly this behavior after $t = 0.5$ in Figure 7.6.28. We attribute the transient behavior before $t = 0.5$ to the initial large uncertainty (variance) given in this simulation.

Compare these results with those of the EKF in Figure 7.6.29. Obviously the multi-modal nature of the model of concern here cannot be captured by the EKF. As shown in Figure 7.6.29 for $t < 1$, the EKF may in fact choose the wrong path as its estimate and then indicate a very small error variance in this estimate.

This section has demonstrated by way of example the effects of measurements on our approximation scheme and found the results to be encouraging as well as easy to implement. In the next section we shall summarize the results of this chapter.

7.7 Summary

In this chapter the goal has been to familiarize ourselves with the approximation scheme as applied to the quartic distribution. We began by deriving the equations which describe the evolution of the parameters. We found here that due to the lack of a closed form relationship between the parameters and the moments of the quartic distribution, only the logarithmic formulation could be considered. And as it turns out, the resultant $\dot{\underline{\alpha}}$ equation is second order in the parameters, and has much structure. Next we specialized these results to particular polynomial dynamics and analyzed the numerical issues associated with the propagation of the parameters. This resulted in further simplifications in the $\dot{\underline{\alpha}}$ equation and in fact gave a closed form solution for $\underline{\alpha}(t)$ when the process noise is zero ($Q = 0$). Following this in Section 5 we evaluate the $F\dot{\underline{\alpha}} = g$ equation numerically for various orders in the dynamics and analyze the results. We find encouragingly that the quartic approximation yields results which behave perfectly for linear dynamics, and within 'loose' limits for $\pm x^2, +x^3, \pm x^4$ dynamics. The limitations are due to the finite width of the weighting function within the inner products and basically limits the usefulness of the approximation to $\|x\| < 2\sqrt{b}$. For the $-x^3$ case we unfortunately found that our approximation behaved poorly. This was due not only to the finite width of the weighting function but also to losses in the projection step. We find, however, in a later example that the inclusion of process noise to the $-x^3$ dynamics results in reasonable filter performance due to fewer losses in the projection step. Finally, we describe an example which takes advantage of the polynomial structure of the quartic exponent to simplify measurement updates. We see here that the update step becomes a trivial addition step.

Chapter 8

Factors In Choosing a Parameterization

8.1 Introduction

In this chapter we shall address the performance issues associated with our filtering scheme. We shall do this by determining the rate at which the error between the true pdf and approximating pdf's increases from a starting point on the manifold of parameterized pdf's. Using this result we then discuss the general implications of singularity of the F matrix, zeros in the g vector, and parametric transformations on filter performance. It is not the intent of this chapter to present a thorough error analysis of our filtering scheme; this we feel is beyond the scope of this thesis. It is the intent, however, to present some simple results which should prove useful in choosing good parameterizations with respect to filter perfomance.

Section 2 will derive the rate of increase in an error measure at any point on the manifold using some of the results from Chapter 2. Section 3 will to use these results to make statements on the effects of the $\underline{g}$ vector and parameter transformations, on the error rate (and hence on performance). Section 4 and 5 will evaluate the error rate for the specific parameterizations considered in this thesis. Section 6 will finally compare these results. In Section 7 we shall consider the effects of a measurement update step on the true error between the underlying pdf and our approximation. Finally, Section 8 will summarize the results of this chapter.

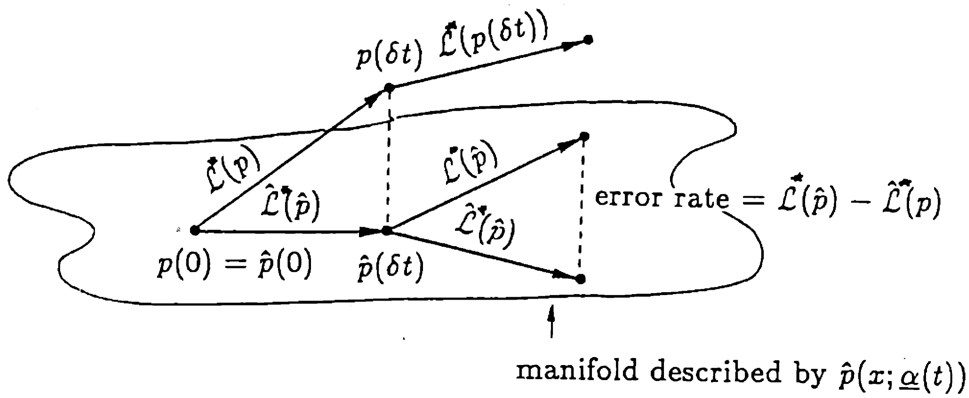


Figure 8.2.1: Error Rate Description

8.2 Error Rate

In this section we shall further evaluate the results from Section 2.5 on error rate under the square root and logarithmic formulations. We study the error rate in place of the true error between our approximating pdf and the true pdf, as a matter of necessity since we do not have available to us the true pdf at any point in time. The error rate as defined in this chapter is the time derivative of the true error but with the true pdf substituted by our approximating pdf, in other words, it is the rate of increase of the true error from any point on the pdf manifold. For comparison, we shall actually study the norm of the error rate.

We start by expanding the error rate equation, now in pdf notation,

$$\|\mathcal{L}^*(\hat{p}) - \hat{\mathcal{L}}^*(\hat{p})\| \tag{8.2.1}$$

where $\mathcal{L}^*(\cdot)$ is the Fokker-Planck (F-P) operator, $\hat{\mathcal{L}}^*(\cdot)$ is the F-P operator re-projected onto the manifold, and $\hat{p}$ is the pdf on that manifold (Refer to Figure 8.2.1). The re-projected F-P operator is given by

$$\hat{\mathcal{L}}^*(\hat{p}) = \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T \dot{\underline{\alpha}}, \tag{8.2.2}$$

describing the differential change of our approximation on the manifold. Since $\underline{\dot{\alpha}}$ is completely described by the $F\underline{\dot{\alpha}} = g$ equation, we can further write

$$\hat{\mathcal{L}}^*(\hat{p}) = \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g}. \tag{8.2.3}$$

Substituting this into Equation 8.2.1 we find

$$\|\mathcal{L}^*(\hat{p}) - \hat{\mathcal{L}}^*(\hat{p})\| =< (\mathcal{L}^*(\hat{p}) - \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g}), (\mathcal{L}^*(\hat{p}) - \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g}) > \tag{8.2.4}$$

Expanding we get

$$=< \mathcal{L}^*(\hat{p}), \mathcal{L}^*(\hat{p}) > -2 < \mathcal{L}^*(\hat{p}), \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g} > + < \underline{g}^T F^{-T} \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}, \begin{pmatrix} \vdots \\ \frac{\partial \hat{p}}{\partial \alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g} > \tag{8.2.5}$$

Since the inner products are assumed over x we may further expand to

$$\begin{aligned} =< \mathcal{L}^*(\hat{p}), \mathcal{L}^*(\hat{p}) > -2 &\begin{pmatrix} \vdots \\ < \mathcal{L}^*(\hat{p}), \frac{\partial \hat{p}}{\partial \alpha_i} > \\ \vdots \end{pmatrix}^T F^{-1}\underline{g} \\ &+\underline{g}^T F^{-T} \begin{pmatrix} \cdots & \\ \vdots & < \frac{\partial \hat{p}}{\partial \alpha_i}, \frac{\partial \hat{p}}{\partial \alpha_j}^T > \end{pmatrix} F^{-1}\underline{g}. \end{aligned} \tag{8.2.6}$$

Recalling the definitions for F and $\underline{g}$

$$F = \begin{pmatrix} \cdots & \\ \vdots & < \frac{\partial \hat{p}}{\partial \alpha_i}, \frac{\partial \hat{p}}{\partial \alpha_j}^T > \end{pmatrix} \quad (= F^T) \tag{8.2.7}$$

$$\underline{g} = \begin{pmatrix} \vdots \\ < \mathcal{L}^*(\hat{p}), \frac{\partial \hat{p}}{\partial \alpha_i} > \\ \vdots \end{pmatrix} \tag{8.2.8}$$

we find

$$\|\epsilon(t)\| =< \mathcal{L}^*(\hat{p}), \mathcal{L}^*(\hat{p}) > -\underline{g}^T F^{-1}\underline{g} \tag{8.2.9}$$

where we have defined the error rate as $\epsilon(t)$. This form will allow us to better understand the ramifications of various parameterizations on the performance of our filter. The following section will explore some of these effects.

8.3 Effects of the Parameterization

In this section we shall use the error rate equation (Equation 8.2.9) to determine the effects of the choice of parameterization on performance. First we look at how the error rate is effected by the g vector. From the definition of $\underline{g}$, having a $\underline{g}$ vector with zero magnitude indicates that all the components of $\mathcal{L}^*(\hat{p})$ have directions that are orthogonal to the bases of the parametric manifold we have chosen. This implies that the chosen parameterization cannot capture any of the characteristics of the optimal nonlinear filtering solution. This would clearly lead to a poor approximation and is verified by Equation 8.2.9.

Next we describe the effects of the F matrix on the error rate. From the definition of the F matrix, its singularity ($det|F^{-1}| \to \infty$) implies that our particular parameterization is redundant, i.e., that at least a pair of bases cover an identical component of the parametric manifold. From a practical standpoint, this inefficiency is not a desirable feature of a particular parameterization and the error rate equation bears this out.

Now we consider the effects of a transformation of the parameters. Define the transformation from parameters $\underline{\alpha}$ to $\underline{\beta}$ as

$$\underline{\alpha} = \gamma(\underline{\beta}). \tag{8.3.1}$$

The key quantity in deriving our approximations is $\frac{\partial \hat{p}}{\partial \underline{\alpha}}$. This can be written in terms of $\underline{\beta}$ using the following transformation

$$\frac{\partial \hat{p}}{\partial \underline{\beta}} = S\frac{\partial \hat{p}}{\partial \underline{\alpha}} \quad S_{ij} = \frac{\partial \alpha_j}{\partial \beta_i} = \frac{\partial \gamma_j(\underline{\beta})}{\partial \beta_i} \tag{8.3.2}$$

where the transformation S is invertable over the range of parameter values of interest.

Now, using the definitions for F and $\underline{g}$ we find

$$F_{\underline{\alpha}} =< \hat{p}_{\underline{\alpha}}, \hat{p}_{\underline{\alpha}} >= S^{-1} < \hat{p}_{\underline{\beta}}, \hat{p}_{\underline{\beta}} > S^{-T} = S^{-1} F_{\underline{\beta}} S^{-T} \tag{8.3.3}$$

$$F_{\underline{\alpha}}^{-1} = S^T F_{\underline{\beta}}^{-1} S \tag{8.3.4}$$

where we have introduced the notation $\hat{p}_{\underline{\alpha}}$ and $\hat{p}_{\underline{\beta}}$ for the vector of partial derivatives with respect to the components of $\underline{\alpha}$ and $\underline{\beta}$, repectively for brevity. Similarly for $\underline{g}$

$$\underline{g}_{\underline{\alpha}} =< \hat{p}_{\underline{\alpha}}, \mathcal{L}^*(\hat{p}) >= S^{-1} < \hat{p}_{\underline{\beta}}, \mathcal{L}^*(\hat{p}) > . \tag{8.3.5}$$

Although $\mathcal{L}^*(\hat{p})$ is a function of $\underline{\alpha}$, it is not a function of $\frac{\partial \cdot}{\partial \underline{\alpha}}$, hence we may simply replace $\underline{\alpha}$ with $\gamma(\underline{\beta})$ in $\mathcal{L}^*(\hat{p})$ to put it in terms of $\underline{\beta}$, allowing us to write

$$\underline{g}_{\underline{\alpha}} = S^{-1} \underline{g}_{\underline{\beta}} \tag{8.3.6}$$

and hence

$$\|\epsilon(t)_{\underline{\alpha}}\| =< \mathcal{L}^*(\hat{p}), \mathcal{L}^*(\hat{p}) > - \underline{g}_{\underline{\beta}}^T F_{\underline{\beta}}^{-1} \underline{g}_{\underline{\beta}} = \|\epsilon(t)_{\underline{\beta}}\|. \tag{8.3.7}$$

This result implies that the error rate is invariant under parametric transformations. This should not be surprising in view of the fact that our approximation scheme is based upon projections onto the manifold described by the parameters. If the parameterization resulting from a transformation span the same space (as in a polar to rectangular coordinate transformation), the position on the new manifold corresponding to the minimum error will be identical to that obtained under the original parameterization. For example, the two parameterizations:

$$\hat{\varsigma}(x; \underline{\alpha}(t)) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 \text{ and} \tag{8.3.8}$$

$$\hat{\varsigma}(x; \underline{\alpha}(t)) = -\frac{1}{2\beta_2}(x - \beta_1)^2 + \beta_0 \tag{8.3.9}$$

would yield identical performance with respect to the error rate for a certain range of parameters, i.e. $\beta_2 > 0$. This coordinate invariant property gives the designer some leeway in choosing a parameterization, allowing him to base his selection more on the computational issues instead of worrying about minimizing error.

The next two sections we will evaluate the error rate for the two formulations of interest in this thesis. Then in the following section we shall apply these results to the examples from previous chapters and make comparisons.

8.4 Square Root Formultation

Under the square root formulation we deal with the distance between the square roots of pdf's. Hence we replace $\hat{p}(x,t)$ in Equation 8.2.9 with $\sqrt{\hat{p}(x,t)} = \hat{q}$. For further simplification we make the substitution $\hat{p}(x,t) = e^{2\hat{\varsigma}(x,t)}$. We then find the following

$$\hat{\mathcal{L}}^*(\hat{q}) = \hat{q}(x;\underline{\alpha}(t)) \begin{pmatrix} \vdots \\ \hat{\varsigma}_{\alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g}. \tag{8.4.1}$$

Applying the same substitution $\mathcal{L}^*(\hat{p})$ yields (refering to Equation 3.4.15 for details)

$$\mathcal{L}^*(\hat{q}) = \hat{q}(x;\underline{\alpha}(t))\hat{\varsigma}_t \tag{8.4.2}$$

and thus

$$\epsilon(t) = (\hat{\varsigma}_t - \begin{pmatrix} \vdots \\ \hat{\varsigma}_{\alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g})\hat{q}(x;\underline{\alpha}(t)). \tag{8.4.3}$$

At this juncture we assume that $\hat{\varsigma}(x;\underline{\alpha}(t))$ and $\mathbf{f}(x)$ are polynomial functions of x as we did in Chapters 4,5,7. Although this limits the applicability of subsequent work, it does allow us to obtain firm results that we may later use for comparison. Under these assumptions $\hat{\varsigma}_t$ and $\hat{\varsigma}_{\underline{\alpha}}$ also become polynomial in x. This allows us to write the error rate as the difference between two polynomials whose coefficents are a function of the coefficients of x in $\hat{\varsigma}_t$, and the F and $\underline{g}$ terms (and hence $\underline{\alpha}(t)$). Each polynomial is also multiplied by the common factor $\hat{q}(x;\underline{\alpha}(t))$ which may be placed outside the error expression. Using these facts and grouping the coefficients corresponding to $\hat{\varsigma}_t$ into a vector $\underline{fp}$ and terms corresponding to $\begin{pmatrix} \vdots \\ \hat{\varsigma}_{\alpha_i} \\ \vdots \end{pmatrix}^T F^{-1}\underline{g}$ into a vector $\underline{fg}$. Then by factoring out the various coefficients, we may force the error rate into the form

$$\epsilon(t) = \begin{pmatrix} 1 & (x-m) & (x-m)^2 & (x-m)^3 & (x-m)^4 & \cdots \end{pmatrix} (\underline{fp} - \underline{fg})\hat{q}(x;\underline{\alpha}(t)). \tag{8.4.4}$$

Note that we have chosen to write our polynomial in terms of powers of $(x - m)$ instead of x. This is due to the simplifications that come from considering terms centered about the mean, m, of the approximating density. When we form the norm of this error using the square root formulation, we get

$$\|\epsilon(t)\| = (\underline{fp} - \underline{fg})^T \underbrace{[\int (x-m)^{i+j}\hat{p}(x;\underline{\alpha}(t))dx]}_{R_{ij}} (\underline{fp} - \underline{fg}). \tag{8.4.5}$$

Specialize to Gaussian

Since for the square root formulation we have considered only the Gaussian pdf as the approximating density (Chapters 4 and 5), we specialize to it now.

$$\hat{p} = e^{2\hat{\varsigma}(x;\underline{\alpha}(t))}\hat{\varsigma}(x;\underline{\alpha}(t)) = -\frac{V}{2}(x-m)^2 + \frac{c}{2} \quad (P = V^{-1}) \tag{8.4.6}$$

So, R becomes

$$R = \begin{pmatrix} 1 & 0 & P & 0 & 3P^2 & 0 \\ 0 & P & 0 & 3P^2 & 0 & 15P^3 \\ P & 0 & 3P^2 & 0 & 15P^3 & 0 \\ 0 & 3P^2 & 0 & 15P^3 & 0 & 105P^4 \\ 3P^2 & 0 & 15P^3 & 0 & 105P^4 & 0 \\ 0 & 15P^3 & 0 & 105P^4 & 0 & 945P^5 \end{pmatrix}. \tag{8.4.7}$$

Note the sparse structure of R. This will simplify calculations later.

The next step is to find $\underline{fp}$ and $\underline{fg}$. To find $\underline{fp}$ we use the specific polynomial

$$\mathbf{f}(x) = f_0 f_1 x + f_2 x^2 + f_3 x^3 + f_4 x^4 \tag{8.4.8}$$

into the $\hat{\varsigma}_t$ equation given in Chapter 3. Then expand and re-write $\hat{\varsigma}_t$ in the form

$$\hat{\varsigma}_t = \sum_{i=0}^{n} fp_i (x-m)^i \tag{8.4.9}$$

where n is as large as need be (here $n = 5$) to accomodate $\hat{\varsigma}_t$. After some manipulation in solving for the fp_i's, we find

$$\underline{fp} = \frac{1}{2}\begin{pmatrix} -(4f_4m^3+3f_3m^2+2f_2m+f_1) \\ V\sum_{i=0}^{4} f_i m^i - (12f_4m^2+6f_3m+2f_2) \\ (4f_4m^3+3f_3m^2+2f_2m+f_1)V - 12f_4m - 3f_3 \\ (6f_4m^2+3f_3m+f_2)V - 4f_4 \\ (4f_4m+f_3)V \\ f_4V \end{pmatrix} + \frac{Q}{2}\begin{pmatrix} -\frac{V}{2} \\ 0 \\ \frac{V^2}{2} \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (8.4.10)$$

For $\underline{fg}$, things are much easier since $(\hat{\varsigma}_{\alpha_i})^T F^{-1}\underline{g}$ is already essentially in the desired form. This is due to $(\hat{\varsigma}_{\alpha_i})$, where the parameters, $\underline{\alpha}$, are (c, m, V).

$$(\hat{\varsigma}_{\alpha_i}) = \begin{pmatrix} \frac{1}{2} \\ \frac{V}{2}(x-m) \\ -\frac{1}{4}(x-m)^2 \end{pmatrix} \quad (8.4.11)$$

Hence $\underline{fg}$ after some manipulation becomes

$$\underline{fg} = \frac{1}{2}\begin{pmatrix} \dot{c} \\ V\dot{m} \\ -\frac{1}{2}\dot{V} \\ 0 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} -(4f_4m^3+3f_3m^2+2f_2m+f_1) - (12f_4m+3f_3) \\ V\sum_{i=0}^{4} f_i m^i + 6f_4m^2+3f_3m+f_2+3f_4P \\ (4f_4m^3+3f_3m^2+2f_2m+f_1)V + 12f_4m + 3f_3 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \frac{Q}{2}\begin{pmatrix} -\frac{V}{2} \\ 0 \\ \frac{V^2}{2} \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (8.4.12)$$

where we have augmented $\underline{fg}$ with zeros to clarify comparisons with $\underline{fp}$ which in general will have more non-zero terms.

Now forming the difference $(\underline{fp} - \underline{fg})$, we see that many terms cancel, yielding

$$(\underline{fp} - \underline{fg}) = \frac{1}{2}\begin{pmatrix} 12f_4m+3f_3 \\ -(18f_4m^2+9f_3m+3f_2+3f_4P) \\ -(24f_4m+6f_3) \\ (6f_4m^2+3f_3m+f_2)V - 4f_4 \\ (4f_4m+f_3)V \\ f_4V \end{pmatrix}. \quad (8.4.13)$$

So that the error rate is given by

$$\|\epsilon(t)\| = (\underline{fp} - \underline{fg})^T R (\underline{fp} - \underline{fg}). \quad (8.4.14)$$

Note that the cancelations have left only terms corresponding to the nonlinear part of the dynamics. Linear and process noise (Q) terms are gone. This means that, for this particular parameterization, the only contributions to the error rate are dynamical terms higher than first order. Conversely, we know that for linear dynamical models, our scheme yields the optimal filter and hence no loss in performance. The error rate obtained here encouragingly verifies this fact. In a later section we shall actually calculate the above quantity and make comparisons. However, in the next section we shall first evaluate the error rate under the logarithmic formulation.

8.5 Logarithmic Formulation

In this section we evaluate the error rate under the logarithmic formulation. We shall do this for two parameterizations, first the second order parameterization of Chapters 4 and 5, then the 4th order paramterization of Chapter 7. For the logarithmic formulation, $\epsilon(t)$ can be put in the same form as Equation 8.4.4, except without $\hat{q}(x;\underline{\alpha}(t))$. Also, $\underline{fp}$ and $\underline{fg}$ now must be redetermined to reflect the fact that we are now dealing with the log of the pdf, $\hat{\varsigma}(x;\underline{\alpha}(t))$, instead of the pdf itself (here $\hat{p}(x;\underline{\alpha}(t)) = e^{\hat{\varsigma}(x;\underline{\alpha}(t))}$). Factoring out the coefficents of $\underline{fg}$ and $\underline{fp}$ we can force the error rate into the form

$$\epsilon(t) = (1 \quad x \quad x^2 \quad x^3 \quad x^4 \quad \cdots)\,(\underline{fp} - \underline{fg}). \tag{8.5.1}$$

Now applying the inner product associated with the logarithmic formulation, we find

$$\|\epsilon(t)\| = (\underline{fp} - \underline{fg})^T \underbrace{[\int x^{i+j} w(x) dx]}_{R_{ij}} (\underline{fp} - \underline{fg}). \tag{8.5.2}$$

2nd Order Parameterization

Next, finding $\underline{fp}$ using the same $\mathbf{f}(x)$ and the 2nd order parameterization

$$\hat{\varsigma}(x;\underline{\alpha}(t)) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 \tag{8.5.3}$$

into $\hat{\varsigma}_t$ for the logarithmic formulation (see Chapter 3 Table 3.4.3) and putting it into the form

$$\hat{\varsigma}_t = \sum_{i=0}^{n} fp_i x^i \tag{8.5.4}$$

we find

$$\underline{fp} = \begin{pmatrix} -f_1 & -f_0 & 0 \\ -2f_2 & -f_1 & -2f_0 \\ -3f_3 & -f_2 & -2f_1 \\ -4f_4 & -f_3 & -2f_2 \\ 0 & -f_4 & -2f_3 \\ 0 & 0 & -2f_4 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} + \frac{Q}{2} \begin{pmatrix} \alpha_1^2 + 2\alpha_2 \\ 4\alpha_1\alpha_2 \\ 4\alpha_2^2 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{8.5.5}$$

As in the square root case, $\underline{fg}$ is easier to obtain due to the form of $(\hat{\varsigma}_{\alpha_i})$

$$(\hat{\varsigma}_{\alpha_i}) = \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix}. \tag{8.5.6}$$

Therefore, $\underline{fg} = F^{-1}\underline{g}$ in this case or

$$\underline{fg} = \begin{pmatrix} -f_1 & -f_0 + 3b^2 f_4 & 6f_3b^2 \\ -2f_2 - 12f_4b & -f_1 - 3f_3b & -2f_0 - 6f_2b - 30f_4b^2 \\ -3f_3 & -f_2 - 6f_4b & -2f_1 - 12f_3b \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} + \frac{Q}{2} \begin{pmatrix} \alpha_1^2 + 2\alpha_2 \\ 4\alpha_1\alpha_2 \\ 4\alpha_2^2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{8.5.7}$$

Again we have augmented $\underline{fg}$ with zeros to simplify comparison. Now forming the difference $(\underline{fp} - \underline{fg})$ we find

$$(\underline{fp} - \underline{fg}) = \begin{pmatrix} 0 & -3b^2 f_4 & -6f_3 b^2 \\ 12f_4 b & 3f_3 b & 6f_2 b + 30 f_4 b^2 \\ 0 & 6f_4 b & 12 f_3 b \\ -4f_4 & -f_3 & -2f_2 \\ 0 & -f_4 & -2f_3 \\ 0 & 0 & -2f_4 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} \tag{8.5.8}$$

So that the error rate is given by

$$\|\epsilon(t)\| = (\underline{fp} - \underline{fg})^T R (\underline{fp} - \underline{fg}) \tag{8.5.9}$$

where R is given by

$$R = \begin{pmatrix} 1 & 0 & b & 0 & 3b^2 & 0 \\ 0 & b & 0 & 3b^2 & 0 & 15b^3 \\ b & 0 & 3b^2 & 0 & 15b^3 & 0 \\ 0 & 3b^2 & 0 & 15b^3 & 0 & 105b^4 \\ 3b^2 & 0 & 15b^3 & 0 & 105b^4 & 0 \\ 0 & 15b^3 & 0 & 105b^4 & 0 & 945b^5 \end{pmatrix}. \tag{8.5.10}$$

Again we see that the linear and noise terms have canceled out, leaving only higher than first order dynamical terms to contribute to the error. Since our scheme under the logarithmic formulation also leads to the optimal solution for linear dynamics, once more we have supported the validity of our error rate norm in predicting 'good' performance.

Quartic Nonlinear

For further comparison, we evaluate the error rate for the 4th order parameterization $(\hat{\varsigma}(x;\underline{\alpha}(t)) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \alpha_4 x^4)$ as well. Following the same steps as before, we find

$$\underline{fg} = \begin{pmatrix} -f_1 & -f_0 & 0 & -45b^3f_4 & -60b^3f_3 \\ -2f_2 & -f_1 & -2f_0+30b^2f_4 & 45b^2f_3 & 840b^3f_4+60b^2f_2 \\ -3f_3 & -f_2 & -2f_1 & -3f_0+135b^2f_4 & 180b^2f_3 \\ -4f_4 & -f_3 & -20bf_4-2f_2 & -30bf_3-3f_1 & -420b^2f_4-40bf_2-4f_0 \\ 0 & -f_4 & -2f_3 & -45bf_4-3f_2 & -60bf_3-4f_1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} + \frac{Q}{2} \begin{pmatrix} 240b^3\alpha_4^2+2\alpha_2+\alpha_1^2 \\ -360b^2\alpha_3\alpha_4+6\alpha_3+4\alpha_1\alpha_2 \\ -720b^2\alpha_4^2+12\alpha_4+6\alpha_1\alpha_3+4\alpha_2^2 \\ 240b\alpha_3\alpha_4+8\alpha_1\alpha_4+12\alpha_2\alpha_3 \\ 240b\alpha_4^2+16\alpha_2\alpha_4+9\alpha_3^2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{8.5.11}$$

and

$$\underline{fp} = \begin{pmatrix} -f_1 & -f_0 & 0 & 0 & 0 \\ -2f_2 & -f_1 & -2f_0 & 0 & 0 \\ -3f_3 & -f_2 & -2f_1 & -3f_0 & 0 \\ -4f_4 & -f_3 & -2f_2 & -3f_1 & -4f_0 \\ 0 & -f_4 & -2f_3 & -3f_2 & -4f_1 \\ 0 & 0 & -2f_4 & -3f_3 & -4f_2 \\ 0 & 0 & 0 & -3f_4 & -4f_3 \\ 0 & 0 & 0 & 0 & -4f_4 \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix}$$

$$+\frac{Q}{2}\begin{pmatrix} \alpha_1^2+2\alpha_2 \\ 4\alpha_1\alpha_2+6\alpha_3 \\ 6\alpha_1\alpha_3+4\alpha_2^2+12\alpha_4 \\ 8\alpha_1\alpha_4+12\alpha_2\alpha_3 \\ 16\alpha_2\alpha_4+9\alpha_3^2 \\ 24\alpha_3\alpha_4 \\ 16\alpha_4^2 \\ 0 \end{pmatrix}. \tag{8.5.12}$$

Forming the difference we find

$$(\underline{fp}-\underline{fg}) =$$

$$\begin{pmatrix} 0 & 45b^3f_4 & 60b^3f_3 \\ -30b^2f_4 & -45b^2f_3 & -840b^3f_4-60b^2f_2 \\ 0 & -135b^2f_4 & -180b^2f_3 \\ 20bf_4 & 30bf_3 & 420b^2f_4+40bf_2 \\ 0 & 45bf_4 & 60bf_3 \\ -2f_4 & -3f_3 & -4f_2 \\ 0 & -3f_4 & -4f_3 \\ 0 & 0 & -4f_4 \end{pmatrix}\begin{pmatrix} \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix}+\frac{Q}{2}\begin{pmatrix} -240b^3\alpha_4^2 \\ 360b^2\alpha_3\alpha_4 \\ 720b^2\alpha_4^2 \\ -240b\alpha_3\alpha_4 \\ -240b\alpha_4^2 \\ 24\alpha_3\alpha_4 \\ 16\alpha_4^2 \\ 0 \end{pmatrix} \tag{8.5.13}$$

So that the error rate is given by

$$\|\epsilon(t)\| = (\underline{fp}-\underline{fg})^T R(\underline{fp}-\underline{fg}) \tag{8.5.14}$$

where R is a matrix of moments of $w(x) \sim N(x;0,b)$.

$$R = \begin{pmatrix} 1 & 0 & b & 0 & 3b^2 & 0 & 15b^3 & 0 \\ 0 & b & 0 & 3b^2 & 0 & 15b^3 & 0 & 105b^4 \\ b & 0 & 3b^2 & 0 & 15b^3 & 0 & 105b^4 & 0 \\ 0 & 3b^2 & 0 & 15b^3 & 0 & 105b^4 & 0 & 945b^5 \\ 3b^2 & 0 & 15b^3 & 0 & 105b^4 & 0 & 945b^5 & 0 \\ 0 & 15b^3 & 0 & 105b^4 & 0 & 945b^5 & 0 & 10395b^6 \\ 15b^3 & 0 & 105b^4 & 0 & 945b^5 & 0 & 10395b^6 & 0 \\ 0 & 105b^4 & 0 & 945b^5 & 0 & 10395b^6 & 0 & 135135b^7 \end{pmatrix} \tag{8.5.15}$$

model parameterization	linear ($f_0, f_1 \neq 0$)	2nd order ($f_2 \neq 0$)	3rd order ($f_3 \neq 0$)
Gaussian	0	$\frac{3Pf_2^2}{2}$	$\frac{f_3^2}{4}(54Pm^2 + 33P^2 - 18P + 9)$

model parameterization	4th order ($f_4 \neq 0$)
Gaussian	$\frac{f_4^2}{4}(216Pm^4 + (960P^2 - 288P + 144)m^2 + 336P^3)$

Table 8.1: $\|\epsilon(t)\|$ under the square root formulation

Note that when we form the difference between $\underline{fp}$ and $\underline{fg}$ in Equation 8.5.13, many of the terms cancel out. However, we see that the effects of process noise (the vector multiplied by Q in Equation 8.5.13) remain. In addition, terms corresponding to α_0 and α_1 also cancel out leaving only $\underline{\alpha}$ and $\mathbf{f}(x)$ components higher than first order. Finally, not the dominant role played by b, the variance of the weighting function, in the error rate. In fact the error rate norm is a function of b^6. This should come as no surprise since $2\sqrt{b}$ sets effective range of the error between true and approximating pdf's to be considered for approximation. Hence the residual error should increase as b is increased. In the following section we shall compare the various $\|\epsilon(t)\|$ results obtained in this section.

8.6 Comparison

In this section we shall compute the error rates for the various examples presented thus far (Chapters 4,5 and 7) and make comparisons between them. Tables 8.1 and 8.2 sumarize the particular results we will be analyzing.

General Characteristics

First we shall make some general observations on the error rate equations obtained in the previous section.

model parameterization	linear ($f_0, f_1 \neq 0$)	2nd order ($f_2 \neq 0$)	3rd order ($f_3 \neq 0$)
2nd order	0	$24b^3\alpha_2^2 f_2^2$	$(6\alpha_1^2 + 96b\alpha_2^2)b^3 f_3^2$
4th order	0	$1920b^5\alpha_4^2 f_2^2$	$(1080\alpha_3^2 + 11520b\alpha_4^2)b^5 f_3^2$

model parameterization	4th order ($f_4 \neq 0$)
2nd order	$(96 + 960b\alpha_2 + 24b\alpha_1^2 + 2880b^2\alpha_2^2)b^3 f_4^2$
4th order	$(480\alpha_2^2 + 40320b\alpha_2\alpha_4 + 6480b\alpha_3^2 + 927360b^2\alpha_4^2)b^5 f_4^2$

Table 8.2: $\|\epsilon(t)\|$ under the logarithmic formulation

Recall that by definition $\|\epsilon(t)\|$ is always positive and that for the work of the previous two sections $\|\epsilon(t)\|$ has a minimum at $\underline{\alpha} = \underline{0}$. Although this may lead one to conclude that the minimum error or best parameterization is achieved when $\underline{\alpha} = 0$, one must be careful to look at the definition of $\|\epsilon(t)\|$. It is the 'rate' at which the true pdf and the approximating pdf will differ given that both the true and approximating pdfs equal a parametric pdf described by $\underline{\alpha}$. Hence it is not surprising that as $\underline{\alpha}$ approaches zero, corresponding to a very wide pdf in these examples, the error rate also approaches zero.

This feature, however, does not detract from the usefulness of the error rate measure, since, for the higher order parameterizations where our work has the greatest application, $\|err(t)\|$ will be a 4th order function of $\underline{\alpha}$. It does say, however, that for the case of no process noise, $Q = 0$ (referring to Equation 8.5.13), our error rate is only 2nd order and therefore we shall have only one minimum at $\underline{\alpha} = 0$. For this case we may only use these results as a relative measure between different parameterizations.

CHAPTER 8. FACTORS IN CHOOSING A PARAMETERIZATION

Comparisons

In the next steps we shall analyze the results presented at the start of this section in Tables 8.1 and 8.2. Under the two formulations, we will make comparisons to once again show the merit of the use of the error rate as a performance measure.

The first case to consider is the linear case. For either formulation we find that since $f_i = 0$ for $i = 2, 3, 4$, the $(\underline{fp} - \underline{fg})$ vector and hence the error rate is zero. This result is by no means unexpected since our filter has been found to yield the exact solutions to the linear examples of Chapter 4 and 7. The value of this result lies in the fact that an exact filter yields a zero error rate thus upholding our performance measure.

The next case to consider is for 2nd order dynamics, i.e., $\dot{x} = f_2 x^2$. The key point to notice here is that not all the parameters contribute to the error rate. In fact, only the highest order parameter (e.g., α_4 for the 4th order parameterization) is of any consequence. As we go to higher order dynamics, we see more parameters come into play. This is in line with what we would expect as the complexity of the state model increases. All this suggests that a rough estimate of filter performance could be had by observing the highest weighted parameter during operation. Conversely, it suggests that a good filter would keep this particular parameter small.

Since simply examining the formulae for $\|err(t)\|$, even for the specific models considered here, does not provide a clear comparison, we shall resort to a numerical example. In deciding what to compare, we may apply the result of Equation 8.3.7 to immediately rule out comparisons between parameterizations that can be written as invertible transformations of one another. This eliminates the comparison between 2nd order and Gaussian pdf under either formulation. (Here the transform is $\alpha_1 = m/P \quad \alpha_2 = -1/(2P)$.) Therefore, we shall make a comparison between the 2nd order and 4th order parameterizations under the logarithmic formulation. We choose the model $\dot{x} = -x^3 + v$, due to its familiarity (Chapters 5,6 and 7). Note that we shall make comparisons between the actual filtering results, i.e., based on the $\underline{\alpha}$ obtained from $F\dot{\underline{\alpha}} = g$. The 'potential' performance for the 4th order parameterization shall always be better than the 2nd order due to the extra degrees of freedom given the 4th order parameterization. This can be seen by considering paths through the 2nd order manifold, which will always be contained within the 4th

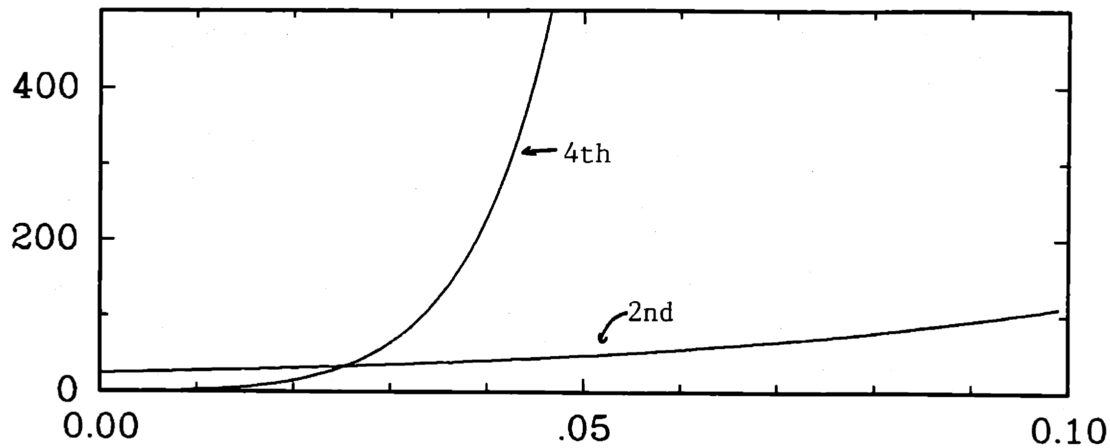


Figure 8.6.2: $f(x) = -x^3 \quad Q = 0 \quad p(x,0) \sim N(0,1) \quad b = 1$

order manifold. We see from Table 8.2 that the 4th order error rate, for $\dot{x} = x^2$ for example, will always be zero. Hence from looking at only the error rate equations, the 4th order parameterization shall always be capable of performing better than the 2nd order parameterization.

Three plots of $\|err(t)\|$ versus time are presented in Figures 8.6.2, 8.6.3, and 8.6.4. These correspond to various process noise values of $Q = 0, 0.1, 3.0$, respectively. We would expect that the higher order parameterization would always perform at least as well as a lower order parameterization simply due to the extra degrees of freedom associated with including more parameters. However, we see from the first plot that for $Q = 0$, the 4th order parameterization has a higher error rate than the 2nd order one. This should come as no surprize when we consider the actual performance of this filter configuration as seen in Chapter 7. As we increase Q in the next plots we see that eventually the 4th order error rate becomes less than the 2nd order result. This characteristic is also verified by the true filter performance as was shown in Chapter 7 (with $Q = .001$). As an additional note we find, by careful study of Figure 8.6.2 and Figure 7.5.13 from Chapter 7, that the location of the first anomally in the 4th order error rate corresponds to the point in time where the true pdf experiences stability problems.

Summarizing, we have seen in this section that the error rate cannot be used in

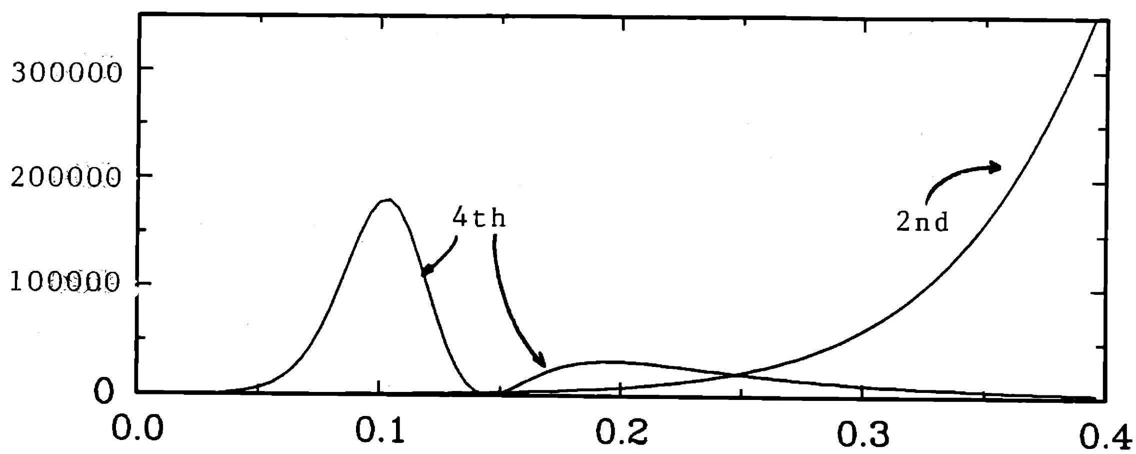


Figure 8.6.3: $\mathbf{f}(x) = -x^3 \quad Q = 0.1 \quad p(x,0) \sim N(0,1) \quad b = 1$

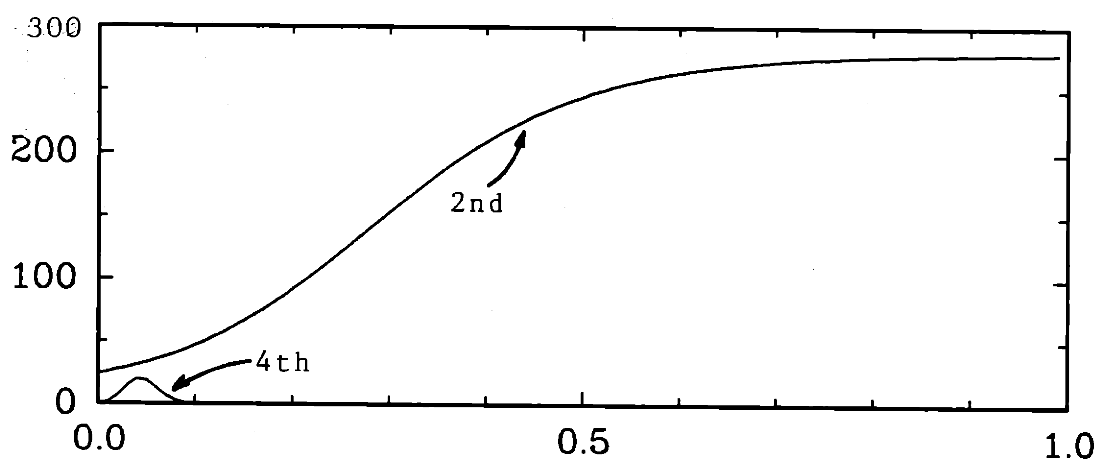


Figure 8.6.4: $\mathbf{f}(x) = -x^3 \quad Q = 3 \quad p(x,0) \sim N(0,1) \quad b = 1$

lieu of a true error, but may be used to determine the relative performance between different model/parameterization combinations. Next, we saw that for the linear case, the error rate figure is zero, thus verifying what we would expect. Then we find that, depending on the dynamical model, only certain parameters contribute to the error rate and therefore should be examined more carefully than the others. Finally, the application of the error rate for 2nd and 4th order parameterizations to a model with known behavior shows an equivalent behavior in the error rate thus supporting the figure's usefulness. In the following section we will turn our attention to the effects of the measurement update step on the true error between the actual underlying pdf and our approximation.

8.7 Study of Update Effects

In this section we will attempt to determine the effects of an update step on the error between true and approximated densities. The hope is to find a consistent relationship which shows that the error after an update is somehow less than the error before the update. We shall first try to determine this relationship under the square root formulation then under the logarithmic formulation.

The error of concern in this section will be defined as $\|p_1 - p_2\|$. Defining the update operation as $\mathbf{h}(\cdot)$, we may then write the error after an update step as $\|\mathbf{h}(p_1)-\mathbf{h}(p_2)\|$. Our goal is then to determine if there exists a consistent relationship between $\|p_1 - p_2\|$ and $\|\mathbf{h}(p_1) - \mathbf{h}(p_2)\|$ under the formulations used in this thesis.

First we consider the square root formulation which corresponds to the L_2 norm with the p_i's replaced with the square root densities, here defined as q_i. Expanding $\|p_1 - p_2\|$ under these conditions yields

$$\|q_1 - q_2\| = \int (q_1 - q_2)^2 dx = \int q_1^2 dx + \int q_2^2 dx - 2\int q_1 q_2 dx \tag{8.7.1}$$

$$= \int 2(1 - \int q_1 q_2 dx). \tag{8.7.2}$$

A similar expansion of $\|\mathbf{h}(q_1) - \mathbf{h}(q_2)\|$, where $\mathbf{h}(\cdot)$ has been modified to satisfy $\int(\mathbf{h}(q_i))^2 dx = 1$, yields

$$\|\mathbf{h}(q_1) - \mathbf{h}(q_2)\| = 2(1 - \int \mathbf{h}(q_1)\mathbf{h}(q_2)dx). \tag{8.7.3}$$

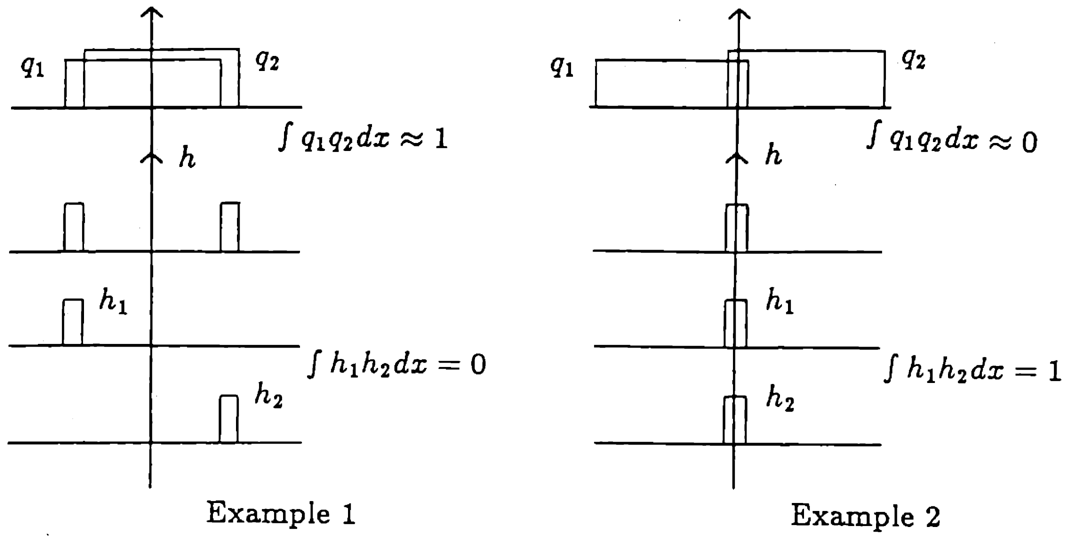


Figure 8.7.5: Examples which show that $\int q_1 q_2 dx$ can be greater than or less than $\int \mathbf{h}(q_1)\mathbf{h}(q_2)dx$

We note that it is equivalent to study the relationship between $\int q_1 q_2 dx$ and $\int \mathbf{h}(q_1)\mathbf{h}(q_2)dx$ instead of the errors themselves since

$$\int q_1 q_2 dx \lessgtr \int \mathbf{h}(q_1)\mathbf{h}(q_2)dx \rightarrow \|\mathbf{h}(q_1) - \mathbf{h}(q_2)\| \lessgtr \|q_1 - q_2\|. \tag{8.7.4}$$

When we go to deterimine this relationship, however, we quickly find examples that satisfy either $\leq$ or $\geq$, thus eliminating the hope for any possible bounding theorems under this formulation. We pictorially outline two examples in Figure 8.7.5.

Next we consider the logarithmic formulation. Under this formulation we replace the p_i's with ς_i's corresponding to the logarithm of the densities. Also note that since we are dealing with the logarithms, $\mathbf{h}(\cdot)$ may be applied to the ς_i's through an addition. Referring to Bayes' rule, $\mathbf{h}(\varsigma_i) = \varsigma_i + \hbar + c_i$, where $\hbar$ =log of the x dependant portion of $\mathbf{h}(\cdot)$,i.e., $p_{y/x}$ and $c_i = -ln(\int e^{\varsigma_i} e^{\hbar} dx)$ log of the normalization constant. Now, expanding $\|p_1 - p_2\|$ under these assumptions and applying the associated inner product we find

$$\|\mathbf{h}(\varsigma_1) - \mathbf{h}(\varsigma_2)\| = \|(\varsigma_1 - \varsigma_2) + (c_1 - c_2)\| \tag{8.7.5}$$

$$= \|\varsigma_1 - \varsigma_2\| + 2(c_1 - c_2)\int(\varsigma_1 - \varsigma_2)w(x)dx + (c_1 - c_2)^2. \tag{8.7.6}$$

So in order to find the relationship between $\|\varsigma_1 - \varsigma_2\|$ and $\|\mathbf{h}(\varsigma_1) - \mathbf{h}(\varsigma_2)\|$ we need only look at the second term.

$$2(c_1 - c_2)\int(\varsigma_1 - \varsigma_2)w(x)dx + (c_1 - c_2)^2 \lessgtr 0 \rightarrow \|\mathbf{h}(\varsigma_1) - \mathbf{h}(\varsigma_2)\| \lessgtr \|\varsigma_1 - \varsigma_2\| \quad (8.7.7)$$

Again when we go to determine the relationship we find that examples for both directions are readily found. One example is simply to use the Gaussian pdf for $p_1, p_2, \mathbf{h}()$. Replacing ς_i with $-\frac{1}{2P}(x - m_i)^2 + \kappa$ results in the following

$$\Delta = (c_1 - c_2) = log(\frac{\int N(x; m_1, P)N(x; m_h, P)dx}{\int N(x; m_2, P)N(x; m_h, P)dx}) \quad (8.7.8)$$

$$2\Delta\int(\varsigma_1 - \varsigma_2)w(x)dx + \Delta^2 = 2\Delta(m_2^2 - m_1^2) + \Delta^2 \quad (8.7.9)$$

We see that the last term can be made either greater or less than zero by selecting m_1, m_2, m_h appropriately. In particular, if $m_2 > m_1$ and $m_h = m_1$ then it is greater than zero. If $m_2 > m_1$ and $m_h = m_2$ then it is less than zero.

Hence we have found that under the formulations considered in this thesis, no statements can be made about the effects of the update step. In particular we found by way of counterexample that in general the error does not decrease after an update step as we had hoped. Note that this is not to say our algorithm shall always diverge.

8.8 Summary

In this chapter we have made an attempt at trying to create tools to help the designer choose a parameterization. In the first section we defined such a tool as the error rate. The error rate is defined as the difference between the true direction of the pdf, given by the F-P equation, and the direction given by our approximation. In order to make the difference useful we choose the starting point for the two directions as equal and on the parametric manifold. Then in order to validate its usefulness as a performance measure we look at how the error rate is effected by various conditions on both the parameterization and the model. We encouragingly find the following

- A linear state model leads to a zero error rate

- Parameterizations with components orthoganal to the true pdf direction as given by the F-P operator ($\underline{g} = \underline{0}$) lead to high error rates
- The choice of units used in the state model has no effect on the error rate
- Invertible transformations of the parameters (e.g., 2nd order to Gaussian) have no effect on the error rate

To further validate the error rate we calculated it for an example with known results from a previous chapter. We again find an agreement between the error rate figure and actual performance.

Finally, in Section 7 we attempt to study the effects of an update step on the true error between underlying and approximate pdf's. We find the results of this section not so encouraging. Under the formulations considered in this thesis, which were made with motivations toward implementation, we found no general conditions for the effects of the measurement update. This may be a goal of future research on this subject.

Chapter 9

Conclusions

9.1 Outline

In this work we have considered the nonlinear filtering problem from a practical viewpoint. The goal of this work has been to find a solution to the filtering problem which provides a probabilistic description of the state, for generality, and which can be calculated from a finite number of recursive (vector) equations. This later goal was driven by the desire to take advantage of the existence of high speed digital computers on which a set of recursive equations could be easily implemented. As a result of our efforts toward this goal, we found a procedure which turns out to be useful, not only for a large class of filtering problems, but also to many problems outside the field.

To summarize, the work of this thesis has lead to two main contributions:

- A technique for approximating solutions to a partial differential equation with the solution to a set of ordinary differential equations.
- A systematic approach to approximating the pdf associated with the optimal nonlinear filter with a finitely parameterized function (resulting from the application of the above).

The first contribution is a technique for approximating partial differential equations of the form

$$\frac{\partial y(x,t)}{\partial t} = f(x,t,y(x,t)) \quad y(x,0) = y_0 \tag{9.1.1}$$

with a set of ordinary differential equations of the form

$$F(\underline{\alpha}(t))\frac{d\underline{\alpha}(t)}{dt} = \underline{g}(\underline{\alpha}(t)) \quad \underline{\alpha}(0) = \underline{\alpha}_0 \tag{9.1.2}$$

$$\text{where } F_{ij} =< \frac{\partial \hat{y}(x;\underline{\alpha}(t))}{\partial \alpha_i}, \frac{\partial \hat{y}(x;\underline{\alpha}(t))}{\partial \alpha_j} >$$

$$\text{and } g_i =< \frac{\partial \hat{y}(x;\underline{\alpha}(t))}{\partial \alpha_i}, f(x,t,\hat{y}(x;\underline{\alpha}(t))) >$$

where $\hat{y}(x;\underline{\alpha}(t))$ is the function approximating the solution to Equation 9.1.1, e.g., $y(x,t) \approx \hat{y}(x;\underline{\alpha}(t))$, and $\underline{\alpha}(t)$ is the vector of parameters which describes the functional approximation. The parametric function $\hat{y}(x;\underline{\alpha}(t))$ is chosen by the filter designer based on any a priori information about the true function's $(y(x,t))$ behavior, and/or on implementation issues. It was found that although the precise form is not critical, a function which cannot capture the key qualitative characteristics of $y(x,t)$ will lead to a poor approximation.

After the form of $\hat{y}(x;\underline{\alpha}(t))$ has been determined, our technique determines the 'optimal' F and $\underline{g}$ terms as a function of this parameterization and $f(,,)$ from the original state model.

This result came about as a byproduct of our work toward finding an efficient nonlinear filtering algorithm, but clearly can be applied to problems outside of filtering theory.

The second contribution of this work comes from the application of our approximation technique to nonlinear filtering problems. Here the differential equation we wish to approximate is the Fokker-Planck (F-P) or the Zakai equation describing the evolution of the conditional density function for a specific stochastic model.

$$\frac{\partial p(x,t)}{\partial t} = \mathcal{L}^*(p(x,t)) \quad p(x,0) = p_0 \Rightarrow \begin{cases} p(x,t) \approx \hat{p}(x;\underline{\alpha}(t)) \\ F(\underline{\alpha}(t))\frac{d\underline{\alpha}(t)}{dt} = \underline{g}(\underline{\alpha}(t)) \quad \underline{\alpha}(0) = \underline{\alpha}_0 \end{cases} \tag{9.1.3}$$

where $\mathcal{L}^*(\cdot)$ is the F-P operator. For the Zakai equation, $\mathcal{L}^*(\cdot)$ is the F-P operator plus an additional measurement term. Using the properties associated with pdf's, the forms of F and g simplify and can be interpreted as functions of moments. These simplifications eliminate the need for any complicated integral evaluations, numerical or otherwise, to determine F or g. Hence, the result is a filter which can be applied to an arbitrary state model and which retains the recursive implementation advantages of the Kalman filter.

In order to demonstrate our approximation technique and to describe the form of F and $\underline{g}$, as well as to illustrate the practical aspects of implementing such a filter, we studied the nonlinear filter by way of example, applying Equation 9.1.3 to various parameterization-model combinations. The results of each of the examples was very encouraging. In particular, we found that (a) many of the presently available filtering schemes are just special cases of our generalized approach to filtering, and (b) that our approach could be applied to complicated pdf approximations with very favorable results. To sumarize, we found that

- For linear dynamics and Gaussian parameterizations, our approach results in the Kalman filter.

- For nonlinear dynamics and Gaussian parameterizations, our approach leads to a filter which takes into account not only all the terms of the EKF and second order filters, but also the additional higher order terms due to nonlinearities.

- For a weighted sum of Gaussians parameterization, our scheme results in a filter which takes into account all of the cross term effects between the elements of the sum, unlike any past work done using this parameterization. This leads to an improvement in filter performance for the specific examples studied.

- Finally, for the quartic parameterization, the applicability and advantages of our scheme are demonstrated by considering an arbitrary 4th order pdf approximation.

In the final section of our work we derived a figure of merit with which we could determine the relative quality of one parameterization over another.

In summary, we find the contributions of this thesis to be a major step toward the practical applications of nonlinear filtering theory to real world problems, and would like to see further applications of this work. Next we describe further work that can be done in the area.

9.2 Further Research

As we see it, further work related to this thesis that should pursued can be broken down into two catagories: applications and theory.

From the application point of view, we would like to see our filtering scheme applied to specific problems whose structure could be used to simplify the formulations to ones that could be processed in real time. This includes the selection of new error criteria and parameterizations which would further simplify processing. In addition we would like to study the performance of our approximation scheme when applied to problems outside of filtering theory.

From the theoretical point of view, we would like to see developed techniques for determining the true error between the approximating and true functions, or at least for determining bounds on the error.

In addition we would like to make connections between our work and that of the mathematical community concerning, in particular, the approximation step. In this thesis we have resorted to the usual linear algebraic techniques in deriving an approximation to the PDE describing pdf evolution. The Lie algebraic techniques used by the mathematical community have been used in an attempt to derive transformations which would reduce the dimensionality of the filtering problem, and have thus far proven useful only in determining the existence or non-existence of such transformations. It would be interesting to exploit the parallelism that exists between the two algebras to see where our approach stands with respect to such transformations, i.e., what does our approach translate to under the Lie algebraic framework. Such work may lead to a clarification of the deficiencies of our approach and hope-fully corrections of these, and in addition may provide a better understanding of the general nonlinear filtering problem.

References

1. S. K. Mitter, Lectures on Nonlinear Filtering and Stochastic Control, *Nonlinear Filtering and Stochastic Control*, Eds A. Moro and S. K. Mitter, Springer Lecture Note in Mathematics, 1983.

2. A. H. Jazwinski, 'Stochastic Processes and Filtering Theory.' Academic Press, New York 1970.

3. V. E. Benes, Exact Finite Dimensional Filters for Certain Diffusions with Nonlinear Drift, *Stochastics,* **5**, pp65-92 (1981).

4. A. Gelb,'Applied Optimal Estimation.' MIT Press, Cambridge MA 1974.

5. G. L. Blankenship, J. S. Baras, Existence, Uniqueness, and Asymptotic Behavior of Solutions to a Class of Zakai Equations with Unbounded Coefficients, *IEEE Transactions on Automatic Control,* Vol AC 28 No **2**, Feb 1983 pp203-214.

6. M. Zakai, On the Optimal Filtering of Diffusion Processes, *Z. Wahr. Verw. Gebiete,* 11, 1969, pp230-243.

7. H. J. Kushner, Dynamical Equations for Optimal Nonlinear Filtering, *J. Differential Equations* **3**, 179-190 (1967).

8. Y. Bar-Shalom, Tracking Methods in a multitarget Environment, *IEEE Transactions on Automatic Control,* Vol AC-23, No **4**, Aug 1978 pp618-626.

9. H. W. Sorenson and A. R. Stubberud, Non-Linear filtering by Approximation of the aposteriori density,*Int. J. of Control Vol* **8**, No **1** pp33-51 (1968).

10. D. L. Alspach and H. W. Sorenson, Nonlinear Baysian Estimation using Gaussian Sum Approximations, *IEEE Transactions on Automatic Control,* Vol AC-17 , No **4**, Aug 1972 pp439-448.

11. P.J.Buxbaum,R.A.Haddad,Recursive Optimal Estimation for a class of Non-Gaussian Processes,Proceedings of the Symposium on Computer Processing in Communications,pp375-399,Polytechnic Institute of Brooklyn,Apr 1969.

12. J.T. Lo, Exponential Fourier Densities and Optimal Estimation and Detection on the Circle, *IEEE Transactions on Information Theory,* Vol IT-23, No 1, Jan 1977 pp110-116.

13. R. W. Brockett, Nonlinear Systems and Nonlinear Estimation Theory, *Stochastic Systems: The Mathematics of Filtering and Identification and Applications*, M. H. Hazewinkel (ed.), 441-477, D. Reidel Publishing Co. 1981.

14. D. E. Gustafson, Linear Minimum-Variance Filters Applied to Phase-Lock Loops, *Proc. Second Symp. on Nonlinear Estimation Theory and its Applications,* San Diego, CA, Sept. 13-15, 1971.

Biography

Born in Washington D.C. on October 2, 1958, Richard Lamb began his life in the usual messy way. Using bribes of scientific toys, his mother enticed him to wander around and mix chemicals, catch bugs, oh and of course build things. After a number of trips to Japan's Akibahara district in Tokyo he decided all he wanted to do was build circuits to tap into phones, break into alarms, and transmit kilowatts of RF power all the while being a general all around brat. While avoiding stab wounds in the DC public school system, he realized what the future held for this sort of individual was at most becoming a TV repairman. Since by this time his habits had become much too expensive for such a career, he decided to become a lawyer by day and a closet hacker by night. Well by 17, in his junior year at George Washington University, he realized that learning how to read and write English for a native would be much too difficult, so the lawyer idea was scrapped, besides 'all he could do was hack' (hack, hack). So he decided to get all the awards that GWU had to offer and go to MIT and get a Masters degree building fancy RF circuitry. Well, after discovering that MIT did no such circuit building except in theory, he decided to stick it out and maybe learn some math (must be something to it, why would all these people be here?) while getting a Masters and Engineers degree in EE. After completing a stint at Draper, consulting at various companies, and taking numerous vacations to far away lands, he realizes that graduate student life isn't so bad; why ever graduate? This was his first mistake. While withering away 6 years of his life in ignorant bliss doing stochastic signal processing he finds out that he's heading for a record stay, so he decides to go on a monetary and circuit building fast and madly race to finish his PhD thesis. Now he stands before you, dazed, bewildered, and confused with the strange urge to go out and break into some DOD computer facility and hack.