Sequential Monte Carlo Methods and their Applications

SEQUENTIAL MONTE CARLO METHODS AND THEIR APPLICATIONS

BY

DEREK YEE

AUGUST 2005

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

Master of Applied Science (2005) McMaster University
(Electrical & Computer Engineering) Hamilton, Ontario

TITLE:	Sequential Monte Carlo Methods and their Applications
AUTHOR:	Derek Yee B.Eng. (Computer Engineering) McMaster University, Hamilton, ON, Canada
SUPERVISOR:	Dr. James P. Reilly, and Dr. Thia Kirubarajan
NUMBER OF PAGES:	xiv, 130

To my parents

Abstract

In recent years, the sequential Monte Carlo method, also referred to as the particle filter has emerged as a powerful methodology for solving the generally difficult nonlinear, non-Gaussian optimal filtering problem. The underlying idea is to use a randomly weighted set of samples to recursively build in time, a point-mass approximation of the true posterior PDF. With this approximation, one can recursively estimate typically intractable posterior expectations of interest. Indeed, the PF can be applied to a very large class of models. Within the last few years, the aforementioned advantages have propelled research on particle filtering and its applications. The subject of this thesis is to the extend the theories and applications of the particle filter. The main contributions of this thesis are described as follows:

1. We consider the optimal filtering problem for a class of partially observed *non–Gaussian* dynamic state space models. In this class, the process equation consists of a combination of linear and nonlinear states, and the process noise for the nonlinear state update is a mixture of Gaussians. In order to solve this problem, we propose a novel method based on an efficient combination of the approximate conditional mean filter and the sequential importance sampling particle filter.

2. We address the problem of channel equalization and phase noise suppression in orthogonal frequency division multiplexing (OFDM) systems. For OFDM

systems, random phase noise introduced by the local oscillator causes two effects: the common phase error (CPE), and the intercarrier interference (ICI). The performance of coherent OFDM systems greatly depends on the ability to accurately estimate the *effective* dynamic channel i.e., the combined effect of the CPE and the time-varying frequency selective channel. With this in mind, we propose an algorithm that equalizes in the frequency domain, and uses a pilot tone aided *particle filter* to track/estimate the effective dynamic channel in the time domain. To increase efficiency, we implement the particle filter via a combination of sequential importance sampling, Rao-Blackwellization, and strategies stemming from the auxiliary particle filter.

Acknowledgements

I would like to express my gratitude to Dr. Reilly and Dr. Kiruba. As supervisors, they have always shown me great patience and unwavering support, in addition, to given me a high degree of freedom and encouragement in my research.

I would also like to thank Amia Zia, Michael Daly, Ramy Gohary, and Dr. William Ng for the many stimulating conversations on research. They were invaluable, motivating, thought provoking experiences that I will treasure for many years to come, particularly, the conversations over coffee that Mike and I all too often look forward too. Cheryl Gies also deserves special mention. She has always provided extraordinary assistance, and for this, I am sincerely in her debt.

Finally, I would like to thank Joanna Chong, Lucy Chong, Sen Chong, Mary Mei and my family for their continuous support in my pursuit of academic excellence. Without them, this thesis would not have been possible.

Notation and Acronyms

Symbol and Definition	
x	Scalar
$\boldsymbol{x}$	Vector
$\boldsymbol{X}$	Matrix
$\boldsymbol{X}^T$	Matrix transpose
$\boldsymbol{X}^H$	Hermitian transpose
$\lvert\boldsymbol{X}\rvert$	Determinant of $\mathbf{X}$
$diag(\boldsymbol{x})$	Diagonal matrix formed from vector $\boldsymbol{x}$
$\mathbb{E}[\cdot]$	Expectation Operator
$\delta(\cdot)$	Dirac-delta function
$\boldsymbol{I}_{n\times n}$	n $\times$ n identity matrix
$\mathbf{0}_{n\times m}$	n $\times$ m matrix of zeroes
$\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$	Normal distribution in $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$, covariance $\boldsymbol{\Sigma}$
$\mathcal{N}_c(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$	Complex Normal distribution in $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$, covariance $\boldsymbol{\Sigma}$
Bayesian Filtering	
N_p	number of particles
N_{eff}	effective sample size
Other Symbols	
Ω	Set of pilot tone locations

S	Spacing between each pilot tone
N	Number of orthogonal subcarriers
N_{cp}	Number of samples in cyclic prefix
P	Number of pilot tones
T_m	Channel delay spread
T	Duration of the useful portion of each OFDM symbol
T_s	Sample interval or duration of high rate data stream
T_t	Duration of one OFDM symbol
Acronyms	
ACF	Autocorrelation function
ACM	Approximate conditional mean
ACMPF	Approximate conditional mean particle filter
APF	Auxiliary particle filter
AR	Autoregressive
ARMA	Autoregressive moving–average
Aux-MKF	Auxiliary Mixture Kalman filter
CP	Cyclic prefix
CPE	Common phase error
DSSM	Dynamic state space model
EKF	Extended Kalman filter
EMKF	Extended mixture Kalman filter
ICI	Intercarrier interference
i.i.d	Independent and identically distributed
IS	Importance sampling
ISI	Intersymbol interference
KF	Kalman filter

LG	Linear Gaussian
GMM	Gaussian mixture model
MAP	Maximum a posteriori
MKF	Mixture Kalman filter
MMSE	Minimum mean square error
MSE	Mean square error
MC	Monte Carlo
OFDM	Orthogonal frequency division multiplexing
OID	Optimal importance distribution
PCRLB	Posterior Cramér-Rao lower bound
PDF	Probability density function
PF	Particle filter
PN	Phase noise
RBPF	Rao-Blackwellized particle filter
RMSE	Root mean square error
S/P	Serial to parallel
SIS	Sequential importance sampling
SMC	Sequential Monte Carlo
TVAR	Time-varying autoregressive
UVGM	Univariate nonstationary growth model

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

In many real-world applications, one requires that unknown quantities be estimated, given a set of noisy observations. Common examples include target tracking [7, 17, 21, 48], channel tracking in wireless communications [30, 36, 23, 8], and the extraction of speech signals from audio environments with contaminating noise disturbances [18, 59, 10]. Frequently, the unknowns can be characterized by a process equation and the observations by a measurement equation, which together allow the formulation of a so-called dynamic state space model (DSSM). As such, we can adopt a Bayesian filtering approach and convert the problem to one of tracking a hidden process from a set of noisy observations.

Often, observations arrive sequentially in time. Therefore, it is more appropriate to consider filters that recursively estimate the unknowns of interest. Ultimately, we aim to recursively compute the exact posterior probability density function (PDF) of interest. Indeed, within the Bayesian framework, the posterior PDF captures all the information about the unknowns. Thus, if an algorithm can recursively deduce the exact posterior PDF, we refer to it as the optimal solution of the aforementioned

Bayesian filtering problem.

For a Linear Gaussian DSSM, the celebrated Kalman filter (KF) provides the optimal solution [26]. Indeed, the KF computes the sufficient statistics at each iteration, which in turn, determines the posterior PDF for each time step k. In general, however, the optimal filter is analytically intractable for the unyielding nonlinear, non-Gaussian DSSM. Thus, a great number of researchers have introduced ingenious approximations that have resulted in mathematically tractable suboptimal filters.

Historically, the Extended Kalman filter (EKF) is the first of such filters [3]. The main idea is to invoke linearization and thereby form an approximation of the original nonlinear model that is amenable to an application of the KF. The resulting recursive formulas constitute the EKF. In a number of applications, the EKF has performed adequately. However, there also exist many scenarios in which the EKF has performed poorly, in particular for non-Gaussian distributed noise disturbances. Thus, it was suggested to consider filters that involve a collection of EKF's.

These filters are known as Gaussian Sum filters [3, 2], and the underlying assumption is to approximate the true posterior PDF by a Gaussian mixture approximation. Each component in the Gaussian mixture approximation, also called a mixand, is computed by a EKF or a KF. Thus, depending on the nature of the non-Gaussianity, these filters utilize a bank of EKF's or KF's to construct an approximation of the true posterior PDF. As such, these methods are more powerful, and more complex to implement. Indeed, if we do not introduce any alleviating procedure, the number of mixands exponentially increases over time. Moreover, the GSF may use a bank of EKF's and thus will still suffer from its inaccuracies. Therefore, it is of interest to consider alternative methods.

One such method is known as the approximate conditional mean (ACM) filter [37, 62]. As shown in [37], for a linear DSSM with non-Gaussian observation noise distributed in accordance to a Gaussian mixture distribution, the ACM filter yields

near optimal performance. Thus, for this scenario, the ACM filter provides an efficient alternative over the computationally expensive Gaussian sum filter.

Up to this point, it is apparent that we are without a universally effective approach for online signal processing of difficult nonlinear non-Gaussian DSSM's. Therefore, it is necessary to consider alternative solutions. Recently, *particle filtering* has emerged as a promising solution to the general nonlinear non-Gaussian filtering problem [12]. The underlying idea is to use a randomly weighted set of samples or *particles* to recursively build in time, a point-mass approximation of the true posterior PDF. Unlike traditional methods described earlier, particle filters do not make any type of approximating assumptions; rather, they build an approximation of the entire posterior PDF itself. In fact, particle filters are applicable to almost any system where signal variations are present. This is even true for nonlinear dynamics and noise distributed according to non-Gaussian distributions. Consequently, particle filters are expected to outperform popular, traditional EKF type algorithms. Indeed, in the last few years, there have been an abundant number of papers on particle filtering and their applications.

The subject of this thesis is to extend the theories and applications of the PF. In particular, we introduce a *novel* particle filter (PF) for an important class of DSSM's, and also apply the PF to difficult problems in wireless communications and nonlinear filtering. The remainder of this thesis is organized as follows.

Chapter 2 reviews various filters proposed in the literature. In particular, we introduce the theory of particle filtering and conclude this chapter with an introduction of the Posterior Cramér-Rao Lower Bound.

Chapter 3 introduces various strategies for improving the efficiency of the basic PF.

Chapter 4 introduces a *novel* PF for a class of partially observed non-Gaussian DSSM's.

Chapter 5 applies the PF to a problem arising in wireless communications. In particular, we propose a particle filtering solution for the problem of channel equalization and phase noise suppression in orthogonal frequency division multiplexing (OFDM) systems.

Chapter 6 concludes this thesis, and provides some suggestions for future research.

Chapter 2

Bayesian Filtering

In this chapter we introduce the problem of sequential Bayesian state estimation. Then, we move on to a brief review of the Kalman filter (KF), approximate conditional mean (ACM) filter, extended Kalman filter (EKF), and a introduction to the particle filter (PF). Finally, we conclude this chapter with an introduction of the Posterior Cramér-Rao Lower Bound (PCRLB) which determines a lower bound on the mean square error (MSE) of the optimal algorithm.

2.1 Sequential Bayesian State Estimation

In this thesis, we consider the problem of using a sequence of noisy observations $\boldsymbol{y}_{1:k} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k\}$ to estimate a state of a system $\boldsymbol{x}_k$ that changes over time. Generally, we will assume that $\boldsymbol{x}_k$ changes in accordance to a process equation:

$$\boldsymbol{x}_k = \boldsymbol{F}_k(\boldsymbol{x}_{k-n:k-1}, \boldsymbol{w}_k) \tag{2.1}$$

where $n \in \mathbb{N}$, the notation $(\cdot)_{l:m}$, indicates all the elements from time l to time m, $\boldsymbol{F}_k(\cdot, \cdot)$ is a possibly nonlinear function that may vary in time, and $\boldsymbol{w}_k$ is the possibly non-Gaussian process noise. It is a well known fact that dynamic state estimation

also requires a measurement equation. That is, we aim to estimate $\boldsymbol{x}_k$ from some noisy observation $\boldsymbol{y}_k$:

$$\boldsymbol{y}_k = \boldsymbol{H}_k(\boldsymbol{x}_k, \boldsymbol{e}_k) \tag{2.2}$$

where $\boldsymbol{H}_k(\cdot, \cdot)$ is a possibly nonlinear measurement function that may vary in time, and $\boldsymbol{e}_k$ is the possibly non-Gaussian observation noise. In this work, the noise sources are assumed to be independent and identically distributed ($i.i.d$) random variables.

Equation (2.1) together with (2.2) form the so-called dynamic state space model (DSSM). Finally, we point out that (2.1) is an n-th order Markov process, and that for $n = 1$, (2.1) reduces to a first order Markov process [3].

2.2 Bayesian Filtering

In this work, we adopt a Bayesian approach to dynamic state estimation, and our general objective is to recursively build in time the posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ from $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1})$. Mathematically, this procedure can be described in two stages. In the first stage, called the *prediction* stage, we use the Chapman-Kolmogorov equation [40] to obtain the predicted posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_{k-1} \tag{2.3}$$

where $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ is the transition prior density that can be determined from (2.1). In the following stage, known as the *update* stage, we obtain $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ by using Bayes' rule [40]. That is, we update $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ via

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) = \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})} \tag{2.4}$$

where the likelihood $p(\boldsymbol{y}_k|\boldsymbol{x}_k)$ can be determined from (2.2), and the normalization density $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$ from

$$p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_k. \tag{2.5}$$

In principle, we can use the posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ to obtain the minimum mean square error (MMSE) estimate

$$\boldsymbol{x}_{k|k} = \mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k] = \int \boldsymbol{x}_k p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k, \tag{2.6}$$

the maximum *a posteriori* (MAP) estimate

$$\boldsymbol{x}_{k|k}^{MAP} = \arg\max_{\boldsymbol{x}_k} \left[p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})\right], \tag{2.7}$$

and any other estimate of interest (i.e. covariance). Thus, we refer to the *exact* recursive solution of (2.3)-(2.5) as the *optimal* solution for Bayesian filtering. However, the computation of (2.3)-(2.6) generally requires the ability to evaluate complex intractable multidimensional integrals. Except for a few special cases, we must make approximations that yield practical suboptimal algorithms. In the following, we review the optimal KF and a selection of these suboptimal algorithms.

2.2.1 The Kalman Filter

When the DSSM satisfies the linear Gaussian (LG) assumption [7], (2.1)-(2.2) can be written as

$$\boldsymbol{x}_k = \boldsymbol{F}_k \boldsymbol{x}_{k-1} + \boldsymbol{u}_k + \boldsymbol{w}_k \tag{2.8}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{e}_k \tag{2.9}$$

where $\boldsymbol{F}_k$ and $\boldsymbol{H}_k$ are known matrices, $\boldsymbol{u}_k$ is a known input vector, $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k)$, $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$ and $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0)$. As usual, the notation $\mathcal{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

For this model, the analytical solutions to (2.3)-(2.5) take the form of a Gaussian distribution. Therefore, if we set $\boldsymbol{x}_{0|0} = \hat{\boldsymbol{x}}_0$, and $\boldsymbol{P}_{0|0} = \hat{\boldsymbol{P}}_0$, it can be shown that for $k = 1, \ldots, n$ the posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$, the predicted posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$,

and the normalization density $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$ satisfy (see [26, 7] or Appendix A):

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{x}_{k|k}, \boldsymbol{P}_{k|k}) \tag{2.10}$$

$$p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{y}_{k|k-1}, \boldsymbol{S}_{k|k-1}) \tag{2.11}$$

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{x}_{k|k-1}, \boldsymbol{P}_{k|k-1}) \tag{2.12}$$

where

$$\boldsymbol{x}_{k|k} = \boldsymbol{x}_{k|k-1} + \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T \boldsymbol{S}_{k|k-1}^{-1}(\boldsymbol{y}_k - \boldsymbol{y}_{k|k-1}) \tag{2.13}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T \boldsymbol{S}_{k|k-1}^{-1}\boldsymbol{H}_k\boldsymbol{P}_{k|k-1} \tag{2.14}$$

$$\boldsymbol{y}_{k|k-1} = \boldsymbol{H}_k\boldsymbol{x}_{k|k-1} \tag{2.15}$$

$$\boldsymbol{S}_{k|k-1} = \boldsymbol{H}_k\boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T + \boldsymbol{R}_k \tag{2.16}$$

$$\boldsymbol{x}_{k|k-1} = \boldsymbol{F}_k\boldsymbol{x}_{k-1|k-1} + \boldsymbol{u}_k \tag{2.17}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_k\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_k^T + \boldsymbol{Q}_k. \tag{2.18}$$

Notice that (2.13)-(2.18) which represents one iteration of the Kalman filter (KF) completely characterizes (2.10)-(2.12), the exact analytical solution to (2.3)-(2.5). Therefore, we arrive at the well known fact that the KF is the *optimal* filter for the highly restrictive LG DSSM. In the following sections, we consider several suboptimal filters for various nonlinear, possibly non-Gaussian DSSM's.

2.2.2 The Approximate Conditional Mean filter

Here, we make the assumption that (2.1)-(2.2) can be written as

$$\boldsymbol{x}_k = \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k + \boldsymbol{w}_k \tag{2.19}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{e}_k \tag{2.20}$$

where $\boldsymbol{F}_k$ and $\boldsymbol{H}_k$ are known matrices, $\boldsymbol{u}_k$ is a known input vector, $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k)$ and $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0)$. However, unlike the LG assumption made in the

previous section, we assume that $p(\boldsymbol{e}_k)$ instead satisfies a Gaussian Mixture Model (GMM):

$$\boldsymbol{e}_k \sim \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{e}_k; \mu_k^j, \boldsymbol{R}_k^j). \tag{2.21}$$

where $\sum_{j=1}^{N} p_j = 1$. Under this assumption, the KF loses its optimality, and may even significantly degrade in performance. In [2], the *optimal* filter is derived, and it is shown that $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ is given by a GMM with an exponentially increasing number of mixands[1]. In practice, the increasing number of mixands results in a prohibitively expensive filter. Evidently, the *optimal* filter is infeasible for real time applications. Thus, with the aim of designing a computationally attractive algorithm, [37] suggested the surprisingly effective assumption that for all time instances k, $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ can be well approximated by a single Gaussian distribution $\mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \hat{\boldsymbol{P}}_{k|k-1})$, that is

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \hat{\boldsymbol{P}}_{k|k-1}). \tag{2.22}$$

Under this assumption, which is known as the *Masreliez approximation* [62], the derived *suboptimal* filter results in the so-called ACM filter [62, 45] and the estimate of the true posterior mean $\boldsymbol{x}_{k|k}$, and covariance $\boldsymbol{P}_{k|k}$ satisfies (see Appendix B for a detailed proof, or [37] for an outline of the derivation)

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T g_k(\boldsymbol{y}_k) \tag{2.23}$$

$$\hat{\boldsymbol{P}}_{k|k} = \hat{\boldsymbol{P}}_{k|k-1} - \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T G_k(\boldsymbol{y}_k)\boldsymbol{H}_k\hat{\boldsymbol{P}}_{k|k-1} \tag{2.24}$$

where

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}_k\hat{\boldsymbol{x}}_{k-1|k-1} + \boldsymbol{u}_k \tag{2.25}$$

$$\hat{\boldsymbol{P}}_{k|k-1} = \boldsymbol{F}_k\hat{\boldsymbol{P}}_{k-1|k-1}\boldsymbol{F}_k^T + \boldsymbol{Q}_{k-1} \tag{2.26}$$

[1] A mixand is a individual component/term of a Gaussian mixture distribution. For instance, if we have a two term GMM, we have two mixands.

and

$$g_k(\boldsymbol{y}_k) = -\frac{1}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})}\nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \tag{2.27}$$

$$G_k(\boldsymbol{y}_k) = \nabla_{\boldsymbol{y}_k} g_k(\boldsymbol{y}_k)^T. \tag{2.28}$$

A comparison between (2.17)-(2.18) and (2.25)-(2.26) reveals that both filters have identical time update formulas. For the measurement update formulas, it can be seen from (2.23) and (2.24) that the ACM filter differs from the KF by $g_k(\boldsymbol{y}_k)$ and $G_k(\boldsymbol{y}_k)$, respectively. In general, $g_k(\boldsymbol{y}_k)$ and $G_k(\boldsymbol{y}_k)$ in (2.23) and (2.24), respectively, are determined by the characteristics of $\boldsymbol{e}_k$. However, for the special case where $\boldsymbol{e}_k$ is Gaussian distributed, it can be shown that $g_k(\boldsymbol{y}_k)$ and $G_k(\boldsymbol{y}_k)$ reduce to $\boldsymbol{S}_{k|k-1}^{-1}(\boldsymbol{y}_k - \boldsymbol{y}_{k|k-1})$ and $\boldsymbol{S}_{k|k-1}^{-1}$ in (2.13) and (2.14), respectively. Thus, the ACM filter reduces to the KF when $\boldsymbol{e}_k$ is Gaussian distributed. Therefore, the ACM filter may be interpreted as a KF that is tailored to the scenario where $\boldsymbol{e}_k$ follows a non-Gaussian distribution.

2.2.3 The Extended Kalman filter

In this section, we consider the case where (2.1)-(2.2) can be written as

$$\boldsymbol{x}_k = \boldsymbol{F}_k(\boldsymbol{x}_{k-1}) + \boldsymbol{w}_k \tag{2.29}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k(\boldsymbol{x}_k) + \boldsymbol{e}_k \tag{2.30}$$

where $\boldsymbol{F}_k(\cdot)$ is a nonlinear function, $\boldsymbol{H}_k(\cdot)$ is a nonlinear measurement function, $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k)$, $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$, and $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0)$. As mentioned before, the optimal solution for (2.29)-(2.30) is generally analytically intractable, and approximations must be made. A sensible approach would be to locally approximate the considered DSSM with a LG DSSM, so that we may apply the standard KF. Therefore, we consider the linearization of $\boldsymbol{F}_k(\boldsymbol{x}_{k-1})$ and $\boldsymbol{H}_k(\boldsymbol{x}_k)$ via a Taylor series

expansion about $\boldsymbol{x}_{k-1|k-1}$ and $\boldsymbol{x}_{k|k-1}$, respectively[2]:

$$\boldsymbol{F}_k(\boldsymbol{x}_{k-1}) \approx \boldsymbol{F}_k(\boldsymbol{x}_{k-1|k-1}) + \underbrace{\nabla_{\boldsymbol{x}_{k-1}} \boldsymbol{F}_k(\boldsymbol{x}_{k-1})\big|_{\boldsymbol{x}_{k-1|k-1}}}_{\hat{\boldsymbol{F}}_k} (\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1|k-1}) \quad (2.31)$$

$$\boldsymbol{H}_k(\boldsymbol{x}_k) \approx \boldsymbol{H}_k(\boldsymbol{x}_{k|k-1}) + \underbrace{\nabla_{\boldsymbol{x}_k} \boldsymbol{H}_k(\boldsymbol{x}_k)|_{\boldsymbol{x}_{k|k-1}}}_{\hat{\boldsymbol{h}}_k} (\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}). \quad (2.32)$$

Thus, if we substitute (2.31) and (2.32) into (2.29) and (2.30), respectively, we have

$$\boldsymbol{x}_k = \hat{\boldsymbol{F}}_k \boldsymbol{x}_{k-1} + \boldsymbol{F}_k(\boldsymbol{x}_{k-1|k-1}) - \hat{\boldsymbol{F}}_k \boldsymbol{x}_{k-1|k-1} + \boldsymbol{w}_k \quad (2.33)$$

$$\boldsymbol{y}_k = \hat{\boldsymbol{h}}_k \boldsymbol{x}_k + \boldsymbol{H}_k(\boldsymbol{x}_{k|k-1}) - \hat{\boldsymbol{h}}_k \boldsymbol{x}_{k|k-1} + \boldsymbol{e}_k. \quad (2.34)$$

The application of the KF leads to a *suboptimal* filter for the original nonlinear DSSM. Evidently, the application of the KF also implies that $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$, and $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ are each individually approximated by a Gaussian distribution. Therefore, it follows that we can write the Extended Kalman Filter (EKF) as [7]

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) \approx \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{P}}_{k|k}) \quad (2.35)$$

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \hat{\boldsymbol{P}}_{k|k-1}) \quad (2.36)$$

where

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{W}_k(\boldsymbol{y}_k - \boldsymbol{H}_k(\hat{\boldsymbol{x}}_{k|k-1})) \quad (2.37)$$

$$\hat{\boldsymbol{P}}_{k|k} = \hat{\boldsymbol{P}}_{k|k-1} - \boldsymbol{W}_k(\hat{\boldsymbol{h}}_k \hat{\boldsymbol{P}}_{k|k-1} \hat{\boldsymbol{h}}_k^T + \boldsymbol{R}_k) \boldsymbol{W}_k^T \quad (2.38)$$

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}_k(\hat{\boldsymbol{x}}_{k-1|k-1}) \quad (2.39)$$

$$\hat{\boldsymbol{P}}_{k|k-1} = \hat{\boldsymbol{F}}_k \hat{\boldsymbol{P}}_{k-1|k-1} \hat{\boldsymbol{F}}_k^T + \boldsymbol{Q}_k \quad (2.40)$$

and the Kalman gain $\boldsymbol{W}_k$ is given by

$$\boldsymbol{W}_k = \hat{\boldsymbol{P}}_{k|k-1} \boldsymbol{H}_k^T (\hat{\boldsymbol{h}}_k \hat{\boldsymbol{P}}_{k|k-1} \hat{\boldsymbol{h}}_k^T + \boldsymbol{R}_k)^{-1}. \quad (2.41)$$

[2]Higher order terms have been neglected.

For a weakly nonlinear system, the EKF provides an acceptable level of performance [7]. For a highly nonlinear system, a Gaussian distribution poorly approximates the posterior density $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$, and the inherent linearizations of $\boldsymbol{F}_k(\boldsymbol{x}_{k-1})$, and $\boldsymbol{H}_k(\boldsymbol{x}_k)$ often lead to filter divergence. Therefore, alternative solutions must be considered, and for this purpose, we consider the particle filter (PF) which has recently emerged as a powerful alternative to the EKF [14].

2.2.4 The Particle Filter

The PF is applicable to a DSSM of the form[3]

$$\boldsymbol{x}_k = \boldsymbol{F}_k(\boldsymbol{x}_{k-n:k-1}) + \boldsymbol{w}_k \tag{2.42}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k(\boldsymbol{x}_k) + \boldsymbol{e}_k \tag{2.43}$$

where $n \in \mathbb{N}$, $\boldsymbol{F}_k(\cdot)$ is a possibly nonlinear process function, $\boldsymbol{H}_k(\cdot)$ is a possibly nonlinear measurement function, $\boldsymbol{w}_k$ is a possibly non-Gaussian process noise, and $\boldsymbol{e}_k$ is a possibly non-Gaussian observation noise. The underlying idea of the PF is to use a randomly weighted set of samples or *particles* to recursively build in time a point-mass approximation of the true posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. Note that these methods do not make any Gaussianity assumption; rather, they construct an approximation of the entire posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ itself. Consequently, they are generally expected to outperform popular, traditional EKF type algorithms.

Unlike previous sections, we consider the more general objective of computing expectations of the form

$$\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \int f(\boldsymbol{x}_{1:k})p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k} \tag{2.44}$$

where $f(\cdot)$ is an arbitrary function of $\boldsymbol{x}_{1:k}$.

[3]Actually the PF is also applicable to a DSSM with multiplicative noise disturbances. However, in this work we will only consider DSSM's of the form (2.42) and (2.43).

The Sequential Importance Sampling (SIS) approach forms the basis for most particle filters [14]. Therefore, in the following, we review perfect Monte Carlo (MC) sampling and Bayesian Importance Sampling before we move onto a detailed derivation of the SIS particle filter.

2.2.4.1 Perfect Monte Carlo Sampling

Perfect MC sampling [49] supposes that we are able to draw N_p *i.i.d* samples $\{\boldsymbol{x}_{1:k}^{(i)}\}_{i=1}^{N_p}$ from the true posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. Together, these samples form a point-mass approximation of $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$:

$$\widehat{p}(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = \frac{1}{N_p}\sum_{i=1}^{N_p}\delta(\boldsymbol{x}_{1:k}-\boldsymbol{x}_{1:k}^{(i)}) \tag{2.45}$$

where $\delta(\cdot)$ is the Dirac-delta function. More importantly, by substituting (2.45) in (2.44), we can estimate (2.44) according to

$$\begin{aligned}\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] &= \int f(\boldsymbol{x}_{1:k})\hat{p}(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k} \\ &= \frac{1}{N_p}\sum_{i=1}^{N_p} f(\boldsymbol{x}_{1:k}^{(i)}).\end{aligned} \tag{2.46}$$

In [12], it is shown that for $N_p \rightarrow \infty$, (2.46) converges to the true posterior expectation (2.44) in the sense of almost sure convergence, i.e.,

$$\lim_{N_p\rightarrow+\infty}\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \frac{1}{N_p}\sum_{i=1}^{N_p} f(\boldsymbol{x}_{1:k}^{(i)}) \overset{a.s}{\rightarrow} \mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] \tag{2.47}$$

where $\overset{a.s}{\rightarrow}$ denotes almost sure convergence. Thus, perfect MC sampling offers significant advantages. Unfortunately, the posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$, being multivariate, and non-Gaussian is usually impossible to efficiently sample from. Consequently, alternate approaches must be considered.

2.2.4.2 Bayesian Importance Sampling

Bayesian Importance Sampling (IS) [49] is based on the following idea. If we cannot efficiently sample from the posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$, then a reasonable solution is to introduce a so-called *importance function* $q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$, that is *easy* to sample from. That is, suppose that we are able to draw N_p samples $\{\boldsymbol{x}_{1:k}^{(i)}\}_{i=1}^{N_p}$ from $q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$, whose support also includes that of $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. In particular, if $\boldsymbol{x}_{1:k}^{(i)} \sim q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ for $i = 1, \ldots, N_p$, and we write (2.44) as

$$\begin{aligned} \mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] &= \int f(\boldsymbol{x}_{1:k}) \frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})} q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_{1:k} \\ &= \int f(\boldsymbol{x}_{1:k}) w(\boldsymbol{x}_{1:k}) q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_{1:k} \end{aligned} \tag{2.48}$$

we can substitute

$$\widehat{q}(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = \frac{1}{N_p} \sum_{i=1}^{N_p} \delta(\boldsymbol{x}_{1:k} - \boldsymbol{x}_{1:k}^{(i)}) \tag{2.49}$$

into (2.48), and obtain for an estimate of $\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})]$:

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \frac{1}{N_p} \sum_{i=1}^{N_p} w(\boldsymbol{x}_{1:k}^{(i)}) f(\boldsymbol{x}_{1:k}^{(i)}) \tag{2.50}$$

where the so-called *importance weights* $w(\boldsymbol{x}_{1:k})$ are equal to

$$w(\boldsymbol{x}_{1:k}^{(i)}) = \frac{p(\boldsymbol{x}_{1:k}^{(i)}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}^{(i)}|\boldsymbol{y}_{1:k})}. \tag{2.51}$$

However, the estimate given by (2.50) is impractical, because the computation of the importance weight

$$w(\boldsymbol{x}_{1:k}^{(i)}) = \frac{p(\boldsymbol{y}_{1:k}|\boldsymbol{x}_{1:k}^{(i)}) p(\boldsymbol{x}_{1:k}^{(i)})}{p(\boldsymbol{y}_{1:k}) q(\boldsymbol{x}_{1:k}^{(i)}|\boldsymbol{y}_{1:k})} \tag{2.52}$$

actually requires the ability to evaluate the typically intractable normalizing density $p(\boldsymbol{y}_{1:k})$.

Therefore, we proceed to write (2.44) as

$$\begin{aligned}\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] &= \int f(\boldsymbol{x}_{1:k})p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k} \\ &= \frac{\int f(\boldsymbol{x}_{1:k})p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}}{\int p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}} \\ &= \frac{\int f(\boldsymbol{x}_{1:k})\frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}}{\int \frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}}, \end{aligned} \tag{2.53}$$

make use of (2.51)

$$\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \frac{\int f(\boldsymbol{x}_{1:k})w(\boldsymbol{x}_{1:k})q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}}{\int w(\boldsymbol{x}_{1:k})q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{1:k}}, \tag{2.54}$$

and substitute (2.49) into (2.54) to arrive at a estimate of $\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})]$ that is independent of $p(\boldsymbol{y}_{1:k})$:

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \frac{\sum_{i=1}^{N_p} w(\boldsymbol{x}_{1:k}^{(i)})f(\boldsymbol{x}_{1:k}^{(i)})}{\sum_{j=1}^{N_p} w(\boldsymbol{x}_{1:k}^{(j)})}. \tag{2.55}$$

Indeed, the normalization process eliminates the necessity of knowing $p(\boldsymbol{y}_{1:k})$. Therefore, if we introduce the so-called *normalized importance weights*

$$\tilde{w}(\boldsymbol{x}_{1:k}^{(i)}) = \frac{w(\boldsymbol{x}_{1:k}^{(i)})}{\sum_{j=1}^{N_p} w(\boldsymbol{x}_{1:k}^{(j)})}, \tag{2.56}$$

$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})]$ can be estimated by

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})f(\boldsymbol{x}_{1:k}^{(i)}). \tag{2.57}$$

Although (2.57) is a biased estimate, it is shown in [12], that for $N_p \to +\infty$, the estimate in (2.57) almost surely converges to the true posterior expectation $\mathbb{E}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})]$. The important implication is that we can interpret the Bayesian IS as a simulation method to sample from the true posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. Indeed,

if we recall (2.44) and rewrite (2.57) as

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_{1:k})] = \int f(\boldsymbol{x}_{1:k}) \underbrace{\sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\delta(\boldsymbol{x}_{1:k} - \boldsymbol{x}_{1:k}^{(i)})}_{\widehat{p}(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})} d\boldsymbol{x}_{1:k} \tag{2.58}$$

it is clear that a point-mass estimate of $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ is given by

$$\widehat{p}(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\delta(\boldsymbol{x}_{1:k} - \boldsymbol{x}_{1:k}^{(i)}). \tag{2.59}$$

The advantages of this approach are clear. Consider the scenario where we are interested in $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ and expectations of the form

$$\mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[g(\boldsymbol{x}_k)|\boldsymbol{y}_{1:k}] = \int g(\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k. \tag{2.60}$$

By simply marginalizing (2.59), we obtain for an estimate of $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$

$$\widehat{p}(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\delta(\boldsymbol{x}_k - \boldsymbol{x}_k^{(i)}) \tag{2.61}$$

and thus, an estimate of $\mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[g(\boldsymbol{x}_k)]$

$$\begin{aligned} \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[g(\boldsymbol{x}_k)] &= \int g(\boldsymbol{x}_k)\widehat{p}(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k \\ &= \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})g(\boldsymbol{x}_k^{(i)}). \end{aligned} \tag{2.62}$$

In particular, if $g(\boldsymbol{x}_k) = \boldsymbol{x}_k$, we have for an approximate MMSE estimate of $\boldsymbol{x}_k$

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[g(\boldsymbol{x}_k)] = \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\boldsymbol{x}_k^{(i)} \tag{2.63}$$

and if $g(\boldsymbol{x}_k) = (\boldsymbol{x}_k - \mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])(\boldsymbol{x}_k - \mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])^T$, we obtain for an estimate of the conditional covariance of $\boldsymbol{x}_k$

$$\begin{aligned} &\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[(\boldsymbol{x}_k - \mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])(\boldsymbol{x}_k - \mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])^T] = \\ &\quad \sum_{i=1}^{N_p} \tilde{w}(\boldsymbol{x}_{1:k}^{(i)})(\boldsymbol{x}_k^{(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])(\boldsymbol{x}_k^{(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k])^T. \end{aligned} \tag{2.64}$$

However, a limitation exists, that is, Bayesian IS in this present form is a "batch" estimator. Indeed, to sequentially in time estimate $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$, we must draw N_p samples of $\boldsymbol{x}_{1:k}$, and compute the associated normalized weights $\{\tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\}_{i=1}^{N_p}$ for every newly available observation $\boldsymbol{y}_k$. Clearly, the computational complexity increases over time, and that Bayesian IS in this conventional form is inadequate for problems that receive observations $\boldsymbol{y}_k$ in a sequential manner. Thus, it is evident that we should obtain a recursive method for estimating $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. That is, for every newly available observation $\boldsymbol{y}_k$, we wish to recursively obtain from $\{\tilde{w}(\boldsymbol{x}_{1:k-1}^{(i)})\}_{i=1}^{N_p}$ and $\{\boldsymbol{x}_{1:k-1}^{(i)}\}_{i=1}^{N_p}$, a new set of weights $\{\tilde{w}(\boldsymbol{x}_{1:k}^{(i)})\}_{i=1}^{N_p}$ and particles $\{\boldsymbol{x}_{1:k}^{(i)}\}_{i=1}^{N_p}$, respectively, which together form an estimate of $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. Therefore, in the following, we develop the aforementioned ideas, and present a Bayesian sequential importance sampling (SIS) procedure.

2.2.4.3 Bayesian Sequential Importance Sampling

We begin by using Bayes rule to write the importance function $q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ as

$$q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})q(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k}). \tag{2.65}$$

Notice that, if we adopted (2.65) as our importance function, the evaluation of (2.57) would require N_p samples of $\boldsymbol{x}_{1:k}$, for every newly available observation $\boldsymbol{y}_k$. In other words, to draw the i-th sample of $\boldsymbol{x}_{1:k}$ given a new observation $\boldsymbol{y}_k$, we would

1. Draw $\boldsymbol{x}_{1:k-1}^{(i)} \sim q(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k})$
2. Draw $\boldsymbol{x}_k^{(i)} \sim q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k})$
3. Set $\boldsymbol{x}_{1:k}^{(i)} = \{\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{x}_k^{(i)}\}$

Clearly, as k increases, the resulting computational burden renders (2.65) infeasible for real time applications. Thus, we restrict our importance function to be of the

form

$$q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})q(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k-1}). \tag{2.66}$$

For this importance function, one can draw N_p samples of $\boldsymbol{x}_{1:k}$, by simply drawing N_p samples of $\boldsymbol{x}_k$, i.e., $\boldsymbol{x}_k^{(i)} \sim q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k})$ for $i = 1, \ldots, N_p$, and appending the resulting set of realizations $\{\boldsymbol{x}_k^{(i)}\}_{i=1}^{N_p}$ to $\{\boldsymbol{x}_{1:k-1}^{(i)}\}_{i=1}^{N_p}$ for every newly available observation $\boldsymbol{y}_k$. Under this restricted form of the importance function, it can be seen that the resulting computational complexity does not increase over time. What remains is to derive a recursive formula for $w(\boldsymbol{x}_{1:k}^{(i)})$, and thus, of $\tilde{w}_k^{(i)}(\boldsymbol{x}_{1:k}^{(i)}) = [\sum_{j=1}^{N_p} w(\boldsymbol{x}_{1:k}^{(j)})]^{-1} w(\boldsymbol{x}_{1:k}^{(i)})$. For convenience, we reproduce (2.51)

$$w(\boldsymbol{x}_{1:k}) = \frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})} \tag{2.67}$$

and write $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ as

$$p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})}. \tag{2.68}$$

By substituting (2.68) and (2.66) in (2.67), we can write $w(\boldsymbol{x}_{1:k})$ as

$$\begin{aligned} w(\boldsymbol{x}_{1:k}) &= \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})} \times \underbrace{\frac{p(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k-1})}{q(\boldsymbol{x}_{1:k-1}|\boldsymbol{y}_{1:k-1})}}_{w(\boldsymbol{x}_{1:k-1})} \\ &= \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})} w(\boldsymbol{x}_{1:k-1}). \end{aligned} \tag{2.69}$$

Although the preceding relationship gives a recursive formula for $w(\boldsymbol{x}_{1:k})$, a few simplifications are in order.

Firstly, the normalization of the importance weights (see (2.56)) means that we can drop $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$ and all other irrelevant constants from (2.69), that is[4]

$$w_k \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k-1})}{q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})} w_{k-1}. \tag{2.70}$$

[4]In (2.70) and in the sequel, we write $w(\boldsymbol{x}_{1:k})$ as w_k.

Secondly, we recognize that (2.42) is a n-th order Markov process, and that, $\boldsymbol{y}_k$ is conditionally independent of $\boldsymbol{y}_{1:k-1}$ given $\boldsymbol{x}_k$. After appropriate modifications, we obtain the desired weight recursion for w_k, that is

$$w_k \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1})}{q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_{1:k})} w_{k-1}. \tag{2.71}$$

The SIS procedure is summarized as follows:

Sequential importance sampling

---

1. **Initialization**: For $i = 1, \ldots, N_p$, initialize the particles, $\boldsymbol{x}_0^{(i)} \sim p(\boldsymbol{x}_0)$, and set $w_0^{(i)} = 1$.

2. **New particles**: For $i = 1, \ldots, N_p$, draw $\boldsymbol{x}_k^{(i)} \sim q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k})$, and set $\boldsymbol{x}_{1:k}^{(i)} \triangleq (\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{x}_k^{(i)})$.

3. **Calculate importance weights**: For $i = 1, \ldots, N_p$, evaluate the importance weights up to a normalizing constant

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{(i)})p(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-n:k-1}^{(i)})}{q(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k})} w_{k-1}^{(i)}$$

   and normalize importance weights.

4. **Estimates**: Use (2.57) for any estimates of interest.

5. **Reiterate**: Set $k = k+1$, and go back to step 2.

As shown above, it is necessary to store all N_p simulated trajectories of $\boldsymbol{x}_k$ (i.e. $\{\boldsymbol{x}_{1:k}^{(i)}\}_{i=1}^{N_p}$). Thus for increasing k, the memory requirements of SIS increases over time. However, if we are only interested in the filtering posterior PDF $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$, or its associated features, then, with the condition that the chosen importance function is of the form $q(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}, \boldsymbol{y}_k)$, it is only necessary that we store $\boldsymbol{x}_{k-n+1:k}$. Typical estimates of interest including $\mathbb{E}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ and $cov_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ are given by (2.63) and (2.64) respectively. Thus, in this case, it can be seen that the memory requirement of SIS does not increase over time.

2.2.4.4 The Degeneracy Problem

In [12], it was shown that the variance of the importance weights ($w_k = p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ $/q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$) increases over time. Although it is non intuitive, this phenomenon is actually detrimental to the performance of the SIS algorithm.

To gain intuition, let us consider the ideal scenario where we are actually able to sample from the true posterior PDF $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. That is $q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$. For this case, it can be shown that the mean and the variance of importance weights w_k satisfy

$$\mathbb{E}_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[w_k] = \mathbb{E}_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}\left[\frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}\right] = 1 \tag{2.72}$$

and

$$var_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[w_k] = var_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}\left[\frac{p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}\right] = 0 \tag{2.73}$$

respectively. Clearly, we want to be as close to this case as possible. Thus, it follows that the inevitable increase in variance of the importance weights w_k leads to a reduction in the accuracy of the SIS algorithm. In practice, what one observes is that after a few iterations, all but one particle will have negligible *normalized* importance weight. Clearly, this phenomenon is undesirable, and unavoidable. From a computational standpoint, it is inefficient to devote such a large amount of effort to update particles whose contribution to the estimate of $p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$ is almost zero. In the literature, this phenomenon is known as *the degeneracy problem* [6]. In practice, it is useful to monitor the degeneracy of the SIS algorithm, and [31] showed that a suitable measure of degeneracy is given by the effective sample size N_{eff}:

$$N_{eff} = \frac{N_p}{1 + var_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[w_k]}. \tag{2.74}$$

The conditional variance of the importance weights is non-negative, and therefore, $N_{eff} \leq N_p$. Intuitively, we can think of the N_{eff} as the number of *i.i.d* particles drawn from the true posterior PDF, that would be necessary to obtain estimates with

the same quality as those given by the weighted particles [38]. In reality, we can only compute an estimate of N_{eff}:

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_p}(\tilde{w}_k^{(i)})^2} \tag{2.75}$$

where $\tilde{w}_k^{(i)}$ is the i-th *normalized* importance weight[5] given by (2.56). However, despite its simplicity (2.75) appeals to intuition. For example, consider the scenario where we have a highly degenerate realization of the SIS algorithm. For this case, as mentioned before, almost all but one particle will have negligible *normalized* importance weight $\tilde{w}_k$. Thus, it can be argued that $\hat{N}_{eff} \approx 1$. Therefore, it follows that $\hat{N}_{eff}$ can provide a reasonable measure of degeneracy, and that it is always beneficial to maximize $\hat{N}_{eff}$.

In the following, we present two strategies that tend to maximize $\hat{N}_{eff}$. The first strategy is the *resampling* of particles, and the second strategy is the selection of a good importance function $q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})$.

2.2.4.5 Resampling

Typically, the prescribed solution for reducing degeneracy is to *resample* the particles. There are numerous schemes, and some of the more popular choices are the residual resampling [34], systematic resampling [29], and the multinomial resampling scheme [20].

The basic idea is to discard particles with weak importance weights and to multiply ones with significant importance weights. Formally, resampling can be described in two steps. First, we draw with replacement N_p *i.i.d* particles $\tilde{\boldsymbol{x}}_{k-n+1:k}^{(i)}$ from [6]

$$\widehat{p}(\boldsymbol{x}_{k-n+1:k}|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \delta(\boldsymbol{x}_{k-n+1:k} - \boldsymbol{x}_{k-n+1:k}^{(i)}) \tag{2.76}$$

such that $P(\tilde{\boldsymbol{x}}_{k-n+1:k}^{(i)} = \boldsymbol{x}_{k-n+1:k}^{(j)}) = \tilde{w}_k^{(j)}$. Secondly, we set the importance weights to $w_k^{(j)} = 1/N_p$ for $j = 1, \ldots, N_p$, respectively.

[5]Notice that we have written $\tilde{w}(\boldsymbol{x}_{1:k}^{(i)})$ as $\tilde{w}_k^{(i)}$.

In [33, 34], it has been argued that when the normalized importance weights are nearly equal, introducing resampling only leads to extra variations. Consequently, we will only introduce resampling when $\hat{N}_{eff}$ is below a fixed heuristic threshold N_{Th}. In this work, we choose the residual resampling scheme, and set $N_{Th} = \frac{2N_p}{3}$.

2.2.4.6 Selection of Importance Function

The importance function can be chosen from a number of choices. However, a sensible approach is to select an importance function that minimizes the variance of the importance weights $var_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[w_k]$. This can be readily justified from (2.74). However, in a recursive scheme, at time *k-1* the importance weights are fixed. Therefore, instead of minimizing $var_{q(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k})}[w_k]$, we minimize the variance of the importance weights conditional upon the past simulated state $\boldsymbol{x}_{k-n:k-1}^{(i)}$, and the observations $\boldsymbol{y}_k$. It is shown in [12], that the resulting *optimal* importance distribution (OID) in the sense that it minimizes $var_{q(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)},\boldsymbol{y}_k)}\left[w_k|\boldsymbol{x}_{k-n:k-1}^{(i)},\boldsymbol{y}_k\right]$ is

$$\begin{aligned} q_{opt}(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)},\boldsymbol{y}_k) &= p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)},\boldsymbol{y}_k) \\ &\propto p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}). \end{aligned} \tag{2.77}$$

For this importance function, we use (2.71) to obtain the following expression for the importance weight:

$$w_k^{(i)} \propto p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})w_{k-1}^{(i)} \tag{2.78}$$

where

$$p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) = \int p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})d\boldsymbol{x}_k. \tag{2.79}$$

Unfortunately, the OID suffers from two drawbacks. Firstly, it requires the ability to sample from (2.77), and secondly, it requires the ability to solve the generally intractable integral (2.79). The aforementioned drawbacks have propelled researchers

to develop close, "easy to sample from" approximations of the OID. In [12], the authors propose a Gaussian approximation of the OID, whereas in [58], it is suggested that one approximate the OID via the application of the Unscented Transform.

Alternatively, we may choose the transition prior as the importance function:

$$q(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k) = p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}). \tag{2.80}$$

For this importance function, the importance weights take the form of

$$w_k^{(i)} \propto p(\boldsymbol{y}_k|\boldsymbol{x}_k^{(i)})w_{k-1}^{(i)}. \tag{2.81}$$

Unlike the OID, the prior $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})$ has the advantage of being relatively simple to implement; indeed, it can be readily obtained from (2.42) and the statistics of $\boldsymbol{w}_k$. However, the prior is inefficient, because it proposes samples without knowledge of the current observation $\boldsymbol{y}_k$. In certain scenarios, the proposed particles may actually reside in "wrong" regions of the state space. Generally, this may result in an inefficient PF which requires more particles to achieve a certain level of performance.

2.2.4.7 Sequential Importance Sampling Particle filter

The SIS particles filter is summarized as follows:

Sequential Importance Sampling Particle filter

---

1. **Initialization**: For $i = 1, \ldots, N_p$, initialize the particles, $\boldsymbol{x}_0^{(i)} \sim p(\boldsymbol{x}_0)$, and set $w_0^{(i)} = 1$.

2. **New particles**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)} = \boldsymbol{x}_{k-n:k-1}^{(i)}$ and draw $\tilde{\boldsymbol{x}}_k^{(i)} \sim q(\boldsymbol{x}_k|\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)$.

3. **Calculate importance weights**: For $i = 1, \ldots, N_p$, compute importance weights up to a normalizing constant

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\tilde{\boldsymbol{x}}_k^{(i)})p(\tilde{\boldsymbol{x}}_k^{(i)}|\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)})}{q(\tilde{\boldsymbol{x}}_k^{(i)}|\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)} w_{k-1}^{(i)}$$

and normalize the importance weights.

4. **Dynamic Resampling**:

   If $\hat{N}_{eff} < N_{Th}$,

   - Resample $\{\tilde{\boldsymbol{x}}_{k-n+1:k}^{(i)}\}_{i=1}^{N_p}$ w.r.t importance weights to obtain $\{\boldsymbol{x}_{k-n+1:k}^{(i)}\}_{i=1}^{N_p}$, and set $w_k^{(i)} = \frac{1}{N_p}$ for $i = 1, \ldots, N_p$.

   otherwise

   - Set $\tilde{\boldsymbol{x}}_{k-n+1:k}^{(i)} = \boldsymbol{x}_{k-n+1:k}^{(i)}$ for $i = 1, \ldots, N_p$.

5. **Estimates**: Compute $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ and $\widehat{cov}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ using (2.63), and (2.64), respectively.

6. **Reiterate**: Set $k = k+1$, and go back to step 2.

Remark 1. *The "tilde" indicates particles before resampling.*

Remark 2. *We can readily derive the Bootstrap filter [20] from the SIS particle filter. The appropriate modifications are: (i) choose the transition prior $p(\mathbf{x}_k|\mathbf{x}_{k-n:k-1})$ as the importance function $q(\mathbf{x}_k|\mathbf{x}_{k-n:k-1}, \mathbf{y}_k)$, and (ii) introduce the resampling step at every time index k.*

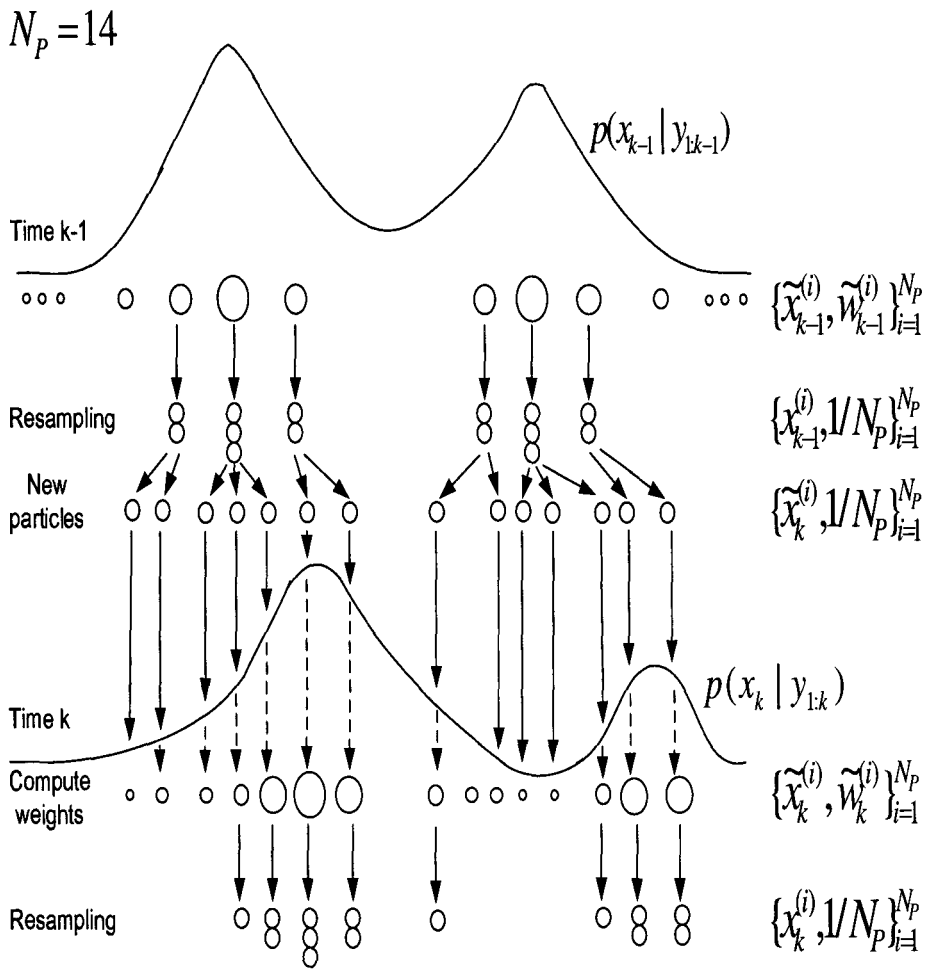


Figure 2.1: Pictorial description of the Bootstrap PF

In Fig. 2.1, we present a pictorial description of the Bootstrap PF. The solid curves represent the true posterior PDF of interest. Observe that each circle represents one particle, and that the diameter of the circle represents the weight that is assigned to that particular particle. Finally, note that, at each stage of the algorithm, the number of particles N_p remain the same.

2.3 Posterior Cramér-Rao Lower Bound

Thus far, we have seen that suboptimal filters rely on approximations. In the EKF, we approximate (linearize) the DSSM via a Taylor's series expansion. In the ACM filter, we approximate the predicted posterior PDF as a Gaussian distribution, and for the PF, we approximate the posterior PDF via a set of randomly weighted samples. Naturally, when compared to the performance of the optimal filter, these approximations degrade performance. In practice, we would like to measure this performance degradation, but more importantly, ascertain the effects of the introduced approximation. For the recursive estimation of *random* parameters, a natural strategy is to adopt a performance criterion (i.e. MSE), and compare the performances of the optimal filter, and the considered suboptimal filter. However, as mentioned before, the optimal filter is generally intractable and impossible to run. Therefore, we consider an alternate approach, and to this end, we introduce the Posterior Cramér-Rao Lower Bound (PCRLB).

The PCRLB as discussed in [55, 60], provides a lower bound on the MSE of the theoretical optimal algorithm, but more importantly, it provides a theoretical benchmark for any other practical suboptimal algorithm[6]. Thus, if we define the MSE matrix $\boldsymbol{M_k}$ as

$$\boldsymbol{M_k} = E[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))^T] \tag{2.82}$$

where $\hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k})$ need not be an unbiased state estimator [60], and use the notation $\Delta^{\boldsymbol{x}_{1:k}}_{\boldsymbol{x}_{1:k}} = \nabla_{\boldsymbol{x}_{1:k}} \nabla^T_{\boldsymbol{x}_{1:k}}$ where $\nabla_{\boldsymbol{x}_{1:k}} = \left[\frac{\partial}{\partial \boldsymbol{x}_1} \ldots, \frac{\partial}{\partial \boldsymbol{x}_k}\right]^T$ to define the Fisher information

[6]For the estimation of *nonrandom* parameters, it is appropriate to consider the Cramér-Rao Lower Bound (CRLB) not the PCRLB [7].

matrix $\boldsymbol{J}_{1:k}$, i.e.,

$$\begin{aligned}
\boldsymbol{J}_{1:k} &= -\mathbb{E}\left[\Delta_{\boldsymbol{x}_{1:k}}^{\boldsymbol{x}_{1:k}} \ln p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})\right] && (2.83)\\
&= \begin{bmatrix} -\mathbb{E}\left[\Delta_{\boldsymbol{x}_{1:k-1}}^{\boldsymbol{x}_{1:k-1}} \ln p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})\right] & -\mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_{1:k-1}} \ln p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})\right] \\ -\mathbb{E}\left[\Delta_{\boldsymbol{x}_{1:k-1}}^{\boldsymbol{x}_k} \ln p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})\right] & -\mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_k} \ln p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})\right] \end{bmatrix} && (2.84)\\
&= \begin{bmatrix} \boldsymbol{A}_k & \boldsymbol{B}_k \\ \boldsymbol{B}_k^T & \boldsymbol{C}_k \end{bmatrix} && (2.85)
\end{aligned}$$

where $p(\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})$ denotes the joint posterior PDF of $\boldsymbol{x}_{1:k}$, and $\boldsymbol{y}_{1:k}$. It follows by virtue of the PCRLB, that the right-lower block of $\boldsymbol{J}_{1:k}^{-1}$, given by $\boldsymbol{J}_k^{-1} = \left[\boldsymbol{C}_k - \boldsymbol{B}_k^T \boldsymbol{A}_k^{-1} \boldsymbol{B}_k\right]^{-1}$ lower bounds the MSE matrix $\boldsymbol{M}_k$ in the sense that

$$\boldsymbol{M}_k = E\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))^T\right] \geq \boldsymbol{J}_k^{-1} \tag{2.86}$$

where $\boldsymbol{M}_k - \boldsymbol{J}_k^{-1}$ is a positive semi-definite matrix. It can be shown that $\boldsymbol{J}_k$ satisfies the recursion [55, 60]:

$$\boldsymbol{J}_k = \boldsymbol{D}_{k-1}^{22} - \boldsymbol{D}_{k-1}^{21} \left(\boldsymbol{J}_{k-1} + \boldsymbol{D}_{k-1}^{11}\right)^{-1} \boldsymbol{D}_{k-1}^{12} \tag{2.87}$$

where

$$\begin{aligned}
\boldsymbol{D}_{k-1}^{11} &= -\mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_k} \ln p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})\right] && (2.88)\\
\boldsymbol{D}_{k-1}^{12} &= -\mathbb{E}\left[\Delta_{\boldsymbol{x}_{k-1}}^{\boldsymbol{x}_k} \ln p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})\right] && (2.89)\\
\boldsymbol{D}_{k-1}^{21} &= -\mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_{k-1}} \ln p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})\right] = \left[\boldsymbol{D}_{k-1}^{12}\right]^T && (2.90)\\
\boldsymbol{D}_{k-1}^{22} &= -\mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_k} \ln p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})\right] - \mathbb{E}\left[\Delta_{\boldsymbol{x}_k}^{\boldsymbol{x}_k} \ln p(\boldsymbol{y}_k|\boldsymbol{x}_k)\right] && (2.91)
\end{aligned}$$

and

$$\boldsymbol{J}_0 = \mathbb{E}\left[\Delta_{\boldsymbol{x}_0}^{\boldsymbol{x}_0} - \ln p(\boldsymbol{x}_0)\right]. \tag{2.92}$$

We remark that (2.87) assumes that $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ and $p(\boldsymbol{y}_k|\boldsymbol{x}_k)$ are known, and that they are twice differentiable with respect to its arguments $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k-1}$. In some

scenarios, this assumption is not satisfied. For example, in the case of $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \delta(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ where $\delta(\cdot)$ is a Dirac delta function, the prior cannot be differentiated. Thus, if we do not introduce appropriate strategies, it is not possible to derive a PCRLB. As a consequence, [55] suggests that we *regularize* the system by replacing the Dirac delta function with a differentiable Gaussian distribution $\mathcal{N}(\boldsymbol{x}_k; 0, \epsilon)$ where $0 < \epsilon \lll 1$. In this work we adopt this idea, and if necessary, approximate the Dirac delta function by a narrow Gaussian distribution $\mathcal{N}(\boldsymbol{x}_k; 0, \epsilon)$. Under this representation, we can use (2.87) to obtain a PCRLB for the regularized system, but more importantly, for $\epsilon \rightarrow 0$, the PCRLB of the regularized system converges to the PCRLB of the original system.

In light of (2.86), we also point out that

$$trace(\boldsymbol{M}_k) = \mathbb{E}\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))^T(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))\right] \geq trace(\boldsymbol{J}_k^{-1}) \tag{2.93}$$

and that

$$[\boldsymbol{M}_k]_{(i,i)} = \left[\mathbb{E}\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k}))^T\right]\right]_{(i,i)} \geq \left[\boldsymbol{J}_k^{-1}\right]_{(i,i)}, \quad i = 1, \ldots, n \tag{2.94}$$

where n denotes the dimension of $\boldsymbol{x}_k$. Equation (2.93) states that the MSE associated with $\hat{\boldsymbol{x}}_k(\boldsymbol{y}_{1:k})$ is lower bounded by the trace of $\boldsymbol{J}_k^{-1}$ where $\boldsymbol{J}_k$ is given by (2.87).

Thus far, we have the option of choosing from three bounds that are inspired by the PCRLB. These are in the form of (2.86), (2.93) and (2.94). For the ensuing experiments, it is convenient to adopt a *scalar* bound. Therefore, unless specified otherwise, we will adopt the lower bound given by (2.93). Finally, the expectations involved in (2.93), do not in general admit any closed form analytical expression. Therefore, when necessary, we proceed as in [17] and approximate the desired expectation with its appropriate Monte Carlo estimate.

Chapter 3

Efficient Particle filters

In this chapter, we introduce various strategies to improve the basic SIS particle filter. The resulting filters are known as the Auxiliary particle filter (APF), the Mixture Kalman filter (MKF), and the Extended Mixture Kalman filter (EMKF). To conclude this chapter, we conduct a series of experiments and compare a selection of the considered algorithms.

3.1 The Auxiliary Particle filter

For convenience we reproduce the considered dynamic state space model (DSSM):

$$\boldsymbol{x}_k = \boldsymbol{F}_k(\boldsymbol{x}_{k-n:k-1}) + \boldsymbol{w}_k \tag{3.1}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k(\boldsymbol{x}_k) + \boldsymbol{e}_k \tag{3.2}$$

where $n \in \mathbb{N}$, $\boldsymbol{F}_k(\cdot)$ is a possibly nonlinear function, $\boldsymbol{H}_k(\cdot)$ is a possibly nonlinear measurement function, $\boldsymbol{w}_k$ is the possibly non-Gaussian process noise, and $\boldsymbol{e}_k$ is the possibly non-Gaussian observation noise. The underlying idea of the APF [4, 43] is to attempt to improve the quality of particles at time k, by preselecting (resampling) the particles at time *k-1* with probability close to $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k})$. That is, we aim to

use all the information up to time k to improve the swarm of particles at time *k-1*. Therefore, in the following, we endeavor to obtain a close approximation of the generally intractable $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k})$. To begin, we expand $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k})$ as follows:

$$\begin{aligned} p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k}) &= \iint p(\boldsymbol{x}_{k-n:k}|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_{k-n:k-2} d\boldsymbol{x}_k \\ &= \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \iint p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k}, \boldsymbol{y}_{1:k-1}) \\ &\quad \times p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}, \boldsymbol{y}_{1:k-1}) p(\boldsymbol{x}_{k-n:k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-n:k-2} d\boldsymbol{x}_k \quad (3.3) \end{aligned}$$

Note that at time k, (3.1) and (3.2) only depend on $\boldsymbol{x}_{k-n:k-1}$ and $\boldsymbol{x}_k$, respectively. Therefore, after subsequent dropping of the normalization density $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$, (3.3) simplifies to

$$p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k}) \propto \iint p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}) p(\boldsymbol{x}_{k-n:k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-n:k-2} d\boldsymbol{x}_k \quad (3.4)$$

At time *k-1*, a particle filter estimate of $p(\boldsymbol{x}_{k-n:k-1}|\boldsymbol{y}_{1:k-1})$ is given by

$$\widehat{p}(\boldsymbol{x}_{k-n:k-1}|\boldsymbol{y}_{1:k-1}) = \sum_{i=1}^{N_p} \tilde{w}_{k-1}^{(i)} \delta(\boldsymbol{x}_{k-n:k-1} - \boldsymbol{x}_{k-n:k-1}^{(i)}). \quad (3.5)$$

Hence, it follows after substituting (3.5) into (3.4) that

$$\begin{aligned} \widehat{p}(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k}) &\propto \iint p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}) \\ &\qquad \times \sum_{i=1}^{N_p} \tilde{w}_{k-1}^{(i)} \delta(\boldsymbol{x}_{k-n:k-1} - \boldsymbol{x}_{k-n:k-1}^{(i)}) d\boldsymbol{x}_{k-n:k-2} d\boldsymbol{x}_k \\ &= \int \sum_{i=1}^{N_p} \Bigg(\int p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}) d\boldsymbol{x}_k \\ &\qquad \times \tilde{w}_{k-1}^{(i)} \delta(\boldsymbol{x}_{k-n:k-1} - \boldsymbol{x}_{k-n:k-1}^{(i)}) \Bigg) d\boldsymbol{x}_{k-n:k-2} \\ &= \sum_{i=1}^{N_p} p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) \tilde{w}_{k-1}^{(i)} \delta(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1}^{(i)}). \quad (3.6) \end{aligned}$$

Typically, it is difficult to evaluate $p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})$

$$p(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) = \int p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) d\boldsymbol{x}_k \quad (3.7)$$

since (3.7) in general requires complex multidimensional integrations. However, in [43], it is suggested that $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})$ may be adequately characterized by a point estimate $\mu_k^{(i)}$, where $\mu_k^{(i)}$ is either the mean or a sample of $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})$. Hence if we assume that $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) \approx \delta(\boldsymbol{x}_k - \mu_k^{(i)})$, and substitute into (3.7), it follows that

$$\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) \;=\; p(\boldsymbol{y}_k|\boldsymbol{x}_k = \mu_k^{(i)}). \tag{3.8}$$

Therefore, if we adopt the approximation given by (3.8), and substitute into (3.6), we obtain a proportionality for an approximation of $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k})$ that is in the form of

$$\begin{aligned}\widehat{p}(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k}) &\propto \sum_{i=1}^{N_p} \widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})\tilde{w}_{k-1}^{(i)}\delta(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1}^{(i)}) \\ &= \sum_{i=1}^{N_p} \lambda_k^{(i)}\delta(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1}^{(i)}) \end{aligned} \tag{3.9}$$

where $\lambda_k^{(i)} = \widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})\tilde{w}_{k-1}^{(i)}$. Finally, by noting that $\int \widehat{p}(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k})d\boldsymbol{x}_{k-1} = 1$, we have

$$\widehat{p}(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k}) \;=\; \sum_{i=1}^{N_p} \tilde{\lambda}_k^{(i)}\delta(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1}^{(i)}) \tag{3.10}$$

where $\tilde{\lambda}_k^{(i)} = [\sum_{j=1}^{N_p} \lambda_k^{(j)}]^{-1}\lambda_k^{(i)}$.

Equation (3.10) forms the basis of the APF. It implies that the most promising particles at time *k-1*, will have the largest associated predictive likelihoods $\lambda_k = \widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1})w_{k-1}$. Moreover, if we rewrite (2.71) as

$$\begin{aligned} w_k &\propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1})}{q(\boldsymbol{x}_k|\boldsymbol{x}_{1:k-1}, \boldsymbol{y}_k)}w_{k-1} \\ &= \underbrace{\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1})w_{k-1}}_{\lambda_k} \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1})}{\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1})q(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}, \boldsymbol{y}_k)} \end{aligned} \tag{3.11}$$

then (3.10) and (3.11) suggest that we should preselect (resample) the particles $\{\boldsymbol{x}_{k-1}^{(i)}\}_{i=1}^{N_p}$ according to the so-called first stage importance weights

$$\lambda_k^{(i)} \propto w_{k-1}^{(i)}\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}) \tag{3.12}$$

and that after preselecting, set the weights to the so-called second stage importance weights

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{(i)})p(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-n:k-1}^{(i)})}{\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})q(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)}. \tag{3.13}$$

As mentioned before, this procedure is advantageous because it uses information $\boldsymbol{y}_k$ at time k to select the most promising particles at time $k-1$. The APF is summarized as follows:

Auxiliary Particle filter

---

1. **Initialization**: For $i = 1, \ldots, N_p$, we initialize the particles, $\boldsymbol{x}_0^{(i)} \sim p(\boldsymbol{x}_0)$, and set $w_0^{(i)} = 1$.

2. **Calculate first stage weights**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)} = \boldsymbol{x}_{k-n:k-1}^{(i)}$

   (a) Draw $\mu_k^{(i)} \sim q(\boldsymbol{x}_k|\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)$.

   (b) Compute importance weights up to a normalizing constant

   $$\lambda_k^{(i)} \propto w_{k-1}^{(i)} \widehat{p}(\boldsymbol{y}_k|\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)})$$

   and normalize importance weights.

3. **Resampling**: Resample $\{\tilde{\boldsymbol{x}}_{k-n:k-1}^{(i)}\}_{i=1}^{N_p}$ w.r.t the first stage importance weights $\lambda_k^{(i)}$ to obtain $\{\boldsymbol{x}_{k-n:k-1}^{(i)}\}_{i=1}^{N_p}$.

4. **New particles**: For $i = 1, \ldots, N_p$ draw $\boldsymbol{x}_k^{(i)} \sim q(\boldsymbol{x}_k|\boldsymbol{x}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)$.

5. **Calculate second stage weights**: For $i = 1, \ldots, N$, compute importance weights up to a normalizing constant

   $$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{(i)})p(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-n:k-1}^{(i)})}{\widehat{p}(\boldsymbol{y}_k|\boldsymbol{x}_{k-n:k-1}^{(i)})q(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-n:k-1}^{(i)}, \boldsymbol{y}_k)}$$

   and normalize importance weights.

6. **Estimates**: Compute $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ and $\widehat{cov}_{p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k]$ using (2.63) and (2.64), respectively.

7. **Reiterate**: Set $k = k+1$, and go back to step 2.

3.2 The Mixture Kalman Filter

Let us suppose that $\boldsymbol{x}_k = [\boldsymbol{x}_k^{1T}, \boldsymbol{x}_k^{2T}]^T$ satisfies the following partially observed Gaussian DSSM [5]:

$$\boldsymbol{x}_k^1 = \boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k-1}^1 + \boldsymbol{w}_k^1 \tag{3.14}$$

$$\boldsymbol{x}_k^2 = \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_k^1 + \boldsymbol{w}_k^2 \tag{3.15}$$

$$\boldsymbol{y}_k = \boldsymbol{H}(\boldsymbol{x}_k^2) + \boldsymbol{e}_k \tag{3.16}$$

where $\boldsymbol{x}_k^1 \in \mathbb{R}^{n_1}$, $\boldsymbol{x}_k^2 \in \mathbb{R}^{n_2}$ are the unobserved processes and $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ is the associated noisy observation. The quantities $\boldsymbol{F}^1(\cdot)$, $\boldsymbol{A}^1(\cdot)$, $\boldsymbol{F}^2(\cdot)$, and $\boldsymbol{A}^2(\cdot)$ are known possibly nonlinear functions with proper dimensions. The process noises $\boldsymbol{w}_k^1$, $\boldsymbol{w}_k^2$ and measurement noise $\boldsymbol{e}_k$ are assumed to be mutually independent zero-mean Gaussian white noise sequences, i.e., $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$, $\boldsymbol{w}_k^2 \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k^2)$ and $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$. The initial states $\boldsymbol{x}_0^1$ and $\boldsymbol{x}_0^2$ are distributed according to $p(\boldsymbol{x}_0^1) = \mathcal{N}(\boldsymbol{x}_0^1; \hat{\boldsymbol{x}}_0^1, \hat{\boldsymbol{P}}_0^1)$ and $p(\boldsymbol{x}_0^2) = \mathcal{N}(\boldsymbol{x}_0^2; \hat{\boldsymbol{x}}_0^2, \hat{\boldsymbol{P}}_0^2)$, respectively.

The considered DSSM is a special case of (2.42)-(2.43). Thus, a "standard" PF such as an SIS particle filter uses an importance function of the form $q(\boldsymbol{x}_k^1, \boldsymbol{x}_k^2|\boldsymbol{x}_{k-n:k-1}^1, \boldsymbol{x}_{k-n:k-1}^2, \boldsymbol{y}_k)$ to obtain an estimate of $p(\boldsymbol{x}_k^1, \boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})$, and other expectations of interest $\mathbb{E}_{p(\boldsymbol{x}_k^1, \boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[f(\boldsymbol{x}_k^1, \boldsymbol{x}_k^2)]$. However, for this particular DSSM, it is possible to design a better algorithm that yields estimates with lower variances. The main idea is to exploit the linear sub-structure of our given DSSM via a Kalman filter (KF). To

elaborate, let us write the joint posterior PDF $p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$ as

$$p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}) = \underbrace{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2)}_{Kalman filter} \underbrace{p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})}_{Particle filter}. \tag{3.17}$$

Clearly, conditional on $\boldsymbol{x}_{1:k}^2$, the matrices $\boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^2)$, $\boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)$, $\boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2)$, $\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)$ become known, and (3.14)-(3.15) form a linear Gaussian (LG) system in $\boldsymbol{x}_k^1$, for which the KF is the optimal estimator. Thus, it is apparent that we can use the optimal KF to obtain the Gaussian PDF $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2)$, and that we can approximate the marginal posterior PDF $p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$ via a PF. In the literature, this hybrid approach is known as the MKF [35] or as described in [12, 15, 50, 10], the Rao-Blackwellized particle filter (RBPF). Intuitively, we may consider the MKF as a algorithm that uses particle filtering for the truly nonlinear state $\boldsymbol{x}_k^2$, and *optimal* Kalman filtering for the conditionally linear state $\boldsymbol{x}_k^1$. Note that the task of using a PF to sample from $p(\boldsymbol{x}_{1:k}^1, \boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$, has been reduced to one of sampling from the *lower dimensional* PDF $p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$. Thus, it follows from intuition, that for a given number of particles the MKF will provide better results then a standard PF. This intuition has been formally proved in [15].

Now, we will proceed with a formal derivation of the MKF. We begin with the derivation of $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2)$. As mentioned before, conditional on $\boldsymbol{x}_{1:k}^2$, (3.14) and (3.15) forms a LG system in $\boldsymbol{x}_k^1$ for which the KF is the optimal estimator. Thus, it follows from the discussion in Section 2.2.1 on Kalman filtering that $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2)$, $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$, and $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2)$ are all Gaussian distributions that satisfy

$$p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2) = \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k}^1, \boldsymbol{P}_{k|k}^1) \tag{3.18}$$

$$p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^2, \boldsymbol{S}_{k|k-1}) \tag{3.19}$$

$$p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2) = \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k-1}^1, \boldsymbol{P}_{k|k-1}^1) \tag{3.20}$$

where

$$\boldsymbol{x}^1_{k|k} = \boldsymbol{x}^1_{k|k-1} + \boldsymbol{P}^1_{k|k-1}\boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})^T\boldsymbol{S}^{-1}_{k|k-1}(\boldsymbol{x}^2_k - \boldsymbol{x}^2_{k|k-1}) \tag{3.21}$$

$$\boldsymbol{P}^1_{k|k} = \boldsymbol{P}^1_{k|k-1} - \boldsymbol{P}^1_{k|k-1}\boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})^T\boldsymbol{S}^{-1}_{k|k-1}\boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})\boldsymbol{P}^1_{k|k-1} \tag{3.22}$$

$$\boldsymbol{x}^2_{k|k-1} = \boldsymbol{F}^2(\boldsymbol{x}^2_{k-n:k-1}) + \boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})\boldsymbol{x}^1_{k|k-1} \tag{3.23}$$

$$\boldsymbol{S}_{k|k-1} = \boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})\boldsymbol{P}^1_{k|k-1}\boldsymbol{A}^2(\boldsymbol{x}^2_{k-n:k-1})^T + \boldsymbol{Q}^2_k \tag{3.24}$$

$$\boldsymbol{x}^1_{k|k-1} = \boldsymbol{F}^1(\boldsymbol{x}^2_{k-n:k-1}) + \boldsymbol{A}^1(\boldsymbol{x}^2_{k-n:k-1})\boldsymbol{x}^1_{k-1|k-1} \tag{3.25}$$

$$\boldsymbol{P}^1_{k|k-1} = \boldsymbol{A}^1(\boldsymbol{x}^2_{k-n:k-1})\boldsymbol{P}^1_{k-1|k-1}\boldsymbol{A}^1(\boldsymbol{x}^2_{k-n:k-1})^T + \boldsymbol{Q}^1_k. \tag{3.26}$$

Recall that $\boldsymbol{Q}^1_k$ and $\boldsymbol{Q}^2_k$ correspond to the covariances of $\boldsymbol{w}^1_k$ and $\boldsymbol{w}^2_k$, respectively. Now we consider the marginal posterior PDF $p(\boldsymbol{x}^2_{1:k}|\boldsymbol{y}_{1:k})$. As mentioned before this density will be approximated by the PF, in which case, $p(\boldsymbol{x}^2_{1:k}|\boldsymbol{y}_{1:k})$ can be written as

$$\widehat{p}(\boldsymbol{x}^2_{1:k}|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}^{(i)}_k \delta(\boldsymbol{x}^2_{1:k} - \boldsymbol{x}^{2(i)}_{1:k}) \tag{3.27}$$

where $\tilde{w}^{(i)}_k = [\sum_{j=1}^{N_p} w^{(j)}_k]^{-1} w^{(i)}_k$ is the *i-th* normalized weight, and $w^{(i)}_k$ satisfies the general weight recursion given by (2.70):

$$w^{(i)}_k \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}^{2,(i)}_{1:k}, \boldsymbol{y}_{1:k-1})p(\boldsymbol{x}^{2,(i)}_k|\boldsymbol{x}^{2,(i)}_{1:k-1}, \boldsymbol{y}_{1:k-1})}{q(\boldsymbol{x}^{2,(i)}_k|\boldsymbol{x}^{2,(i)}_{1:k-1}, \boldsymbol{y}_k)} w^{(i)}_{k-1}. \tag{3.28}$$

Notice from (3.16), that if $\boldsymbol{x}^2_{1:k}$ is known, we can write $p(\boldsymbol{y}_k|\boldsymbol{x}^2_{1:k}, \boldsymbol{y}_{1:k-1})$ as $p(\boldsymbol{y}_k|\boldsymbol{x}^2_k)$ and drop $\boldsymbol{y}_{1:k-1}$ from $p(\boldsymbol{x}^2_k|\boldsymbol{x}^2_{1:k-1}, \boldsymbol{y}_{1:k-1})$. Thus, (3.28) simplifies to

$$w^{(i)}_k \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}^{2,(i)}_k)p(\boldsymbol{x}^{2,(i)}_k|\boldsymbol{x}^{2,(i)}_{1:k-1})}{q(\boldsymbol{x}^{2,(i)}_k|\boldsymbol{x}^{2,(i)}_{1:k-1}, \boldsymbol{y}_k)} w^{(i)}_{k-1} \tag{3.29}$$

where the likelihood $p(\boldsymbol{y}_k|\boldsymbol{x}^2_k)$ can be determined from (3.16), and $p(\boldsymbol{x}^2_k|\boldsymbol{x}^2_{1:k-1})$ from (3.19). As usual, we can choose the prior distribution as the importance function, that is $q(\boldsymbol{x}^2_k|\boldsymbol{x}^{2,(i)}_{1:k-1}, \boldsymbol{y}_k) = p(\boldsymbol{x}^2_k|\boldsymbol{x}^{2,(i)}_{1:k-1})$.

What remains is to develop expressions for the estimates of interest. We begin by obtaining an estimate for $\mathbb{E}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$. Clearly, this is given by

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \int \boldsymbol{x}_k^2 \widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k \tag{3.30}$$

$$= \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_k^{2,(i)}. \tag{3.31}$$

Similarly, an estimate of the conditional covariance $cov_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \mathbb{E}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\bar{\boldsymbol{x}}_k^2 \bar{\boldsymbol{x}}_k^{2T}]$ where $\bar{\boldsymbol{x}}_k^2 = \boldsymbol{x}_k^2 - \mathbb{E}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$ is given by

$$\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} (\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])(\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])^T. \tag{3.32}$$

Finally by noting that $\mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$ is given by

$$\mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \int \boldsymbol{x}_k^1 p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k^1 \tag{3.33}$$

$$= \iint \boldsymbol{x}_k^1 p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k^1 d\boldsymbol{x}_{1:k}^2 \tag{3.34}$$

and $cov_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$ by

$$cov_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \int (\bar{\boldsymbol{x}}_k^1)(\bar{\boldsymbol{x}}_k^1)^T p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k^1 \tag{3.35}$$

$$= \iint (\bar{\boldsymbol{x}}_k^1)(\bar{\boldsymbol{x}}_k^1)^T p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k^1 d\boldsymbol{x}_{1:k}^2 \tag{3.36}$$

where $\bar{\boldsymbol{x}}_k^1 = \boldsymbol{x}_k^1 - \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, the substitution of (3.17) and (3.27) into (3.34) and (3.36), respectively gives

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_{k|k}^{1,(i)} \tag{3.37}$$

and

$$\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \left(\boldsymbol{P}_{k|k}^{1,(i)} + (\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1])(\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1])^T \right). \tag{3.38}$$

In (3.37) and (3.38), we can use a KF that is conditioned on $\boldsymbol{x}_{1:k}^{2,(i)}$ to compute $\boldsymbol{x}_{k|k}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$ and $\boldsymbol{P}_{k|k}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$, respectively. Therefore, a KF must be associated with each particle $\boldsymbol{x}_k^{2,(i)}$, and as the name suggests, the MKF uses a mixture of KF's to estimate $\mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, and $cov_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$. The MKF is summarized as follows:

Mixture Kalman filter

1. **Initialization**: For $i = 1, \ldots, N_p$, we initialize $\boldsymbol{x}_{0|0}^{1(i)} = \hat{x}_0^1$, $\boldsymbol{P}_{0|0}^{1,(i)} = \hat{P}_0^1$, $\boldsymbol{x}_0^{2,(i)} \sim p(\boldsymbol{x}_0^2)$, and set $w_0^{(i)} = 1$.

2. **New particles**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{k-n:k-1}^{2,(i)} = \boldsymbol{x}_{k-n:k-1}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k-1|k-1}^{1,(i)} = \boldsymbol{x}_{k-1|k-1}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k-1|k-1}^{1,(i)} = \boldsymbol{P}_{k-1|k-1}^{1,(i)}$.

   (a) Proposal: Draw $\tilde{\boldsymbol{x}}_k^{2,(i)} \sim q(\boldsymbol{x}_k^2|\tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)}, \boldsymbol{y}_k)$.

   (b) KF Prediction: Compute $\tilde{\boldsymbol{x}}_{k|k-1}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k-1}^{1,(i)}$ using (3.25) and (3.26), respectively.

   (c) KF Update: Compute $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}$ using (3.21) and (3.22), respectively.

3. **Calculate importance weights**: For $i = 1, \ldots, N_p$, compute importance weights up to a normalizing constant

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\tilde{\boldsymbol{x}}_k^{2,(i)})p(\tilde{\boldsymbol{x}}_k^{2,(i)}|\tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)})}{q(\tilde{\boldsymbol{x}}_k^{2,(i)}|\tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)}, \boldsymbol{y}_k)} w_{k-1}^{(i)}$$

   and normalize importance weights.

4. **Dynamic Resampling**:

   If $\hat{N}_{eff} < N_{Th}$,

   - Resample $\{\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ w.r.t importance weights to obtain $\{\boldsymbol{x}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{x}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{P}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ and set $w_k^{(i)} = \frac{1}{N_p}$ for $i = 1, \ldots, N_p$.

otherwise

- Set $\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)} = \boldsymbol{x}_{k-n+1:k}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)} = \boldsymbol{x}_{k|k}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)} = \boldsymbol{P}_{k|k}^{1,(i)}$ for $i = 1, \ldots, N_p$.

5. **Estimates**: Compute $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$, and $\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$ using (3.37), (3.38), (3.31), and (3.32), respectively.

6. **Reiterate**: Set $k = k+1$, and go back to step 2.

3.3 The Extended Mixture Kalman filter

In this section, we consider a class of partially observed *non-Gaussian* DSSM. Such models are useful e.g., in time–varying autoregressive models and other applications as discussed for instance in [5] and [52]. Formally, the considered DSSM is described by

$$\boldsymbol{x}_k^1 = \boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k-1}^1 + \boldsymbol{w}_k^1 \tag{3.39}$$

$$\boldsymbol{x}_k^2 = \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_k^1 + \boldsymbol{w}_k^2 \tag{3.40}$$

$$\boldsymbol{y}_k = \boldsymbol{H}(\boldsymbol{x}_k^2) + \boldsymbol{e}_k \tag{3.41}$$

where $\boldsymbol{x}_k^1 \in \mathbb{R}^{n_1}$, $\boldsymbol{x}_k^2 \in \mathbb{R}^{n_2}$ are the unobserved processes and $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ is the noisy observation. As before, $\boldsymbol{F}^1(\cdot)$, $\boldsymbol{A}^1(\cdot)$, $\boldsymbol{F}^2(\cdot)$, and $\boldsymbol{A}^2(\cdot)$ are known functions with proper dimensions. The process noise $\boldsymbol{w}_k^1$, and measurement noise $\boldsymbol{e}_k$ are assumed to be mutually independent zero-mean Gaussian white noise sequences, i.e., $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$ and $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$. The initial states are assumed to be mutually independent zero-mean Gaussian random variables, $\boldsymbol{x}_0^1 \sim \mathcal{N}(\boldsymbol{x}_0^1; \hat{\boldsymbol{x}}_0^1, \hat{\boldsymbol{P}}_0^1)$ and $\boldsymbol{x}_0^2 \sim \mathcal{N}(\boldsymbol{x}_0^2; \hat{\boldsymbol{x}}_0^2, \hat{\boldsymbol{P}}_0^2)$. However, unlike Section 3.2, the process noise driving (3.40) is assumed to be a white noise sequence that is distributed according to a N component

Gaussian Mixture Model (GMM):

$$p(\boldsymbol{w}_k^2) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(j)}, \boldsymbol{Q}_k^{2,(j)}) \tag{3.42}$$

where $\sum_{j=1}^{N} p_j = 1$. Thus, in light of (3.42), it is clear that the considered DSSM is a partially observed *non-Gaussian* system. Notice that, the MKF is still applicable to the considered DSSM [35]. At first sight, this seems implausible, since (3.39)-(3.40) conditional on $\boldsymbol{x}_{1:k}^2$ is no longer LG in $\boldsymbol{x}_k^1$. However, if we introduce an indicator random variable $I_k \in \mathcal{I}^N = \{n | n = 1, \ldots, N\}$ that satisfies

$$I_k = \begin{cases} 1 & \text{if } \boldsymbol{w}_k^2 \sim \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(1)}, \boldsymbol{Q}_k^{2,(1)}) \\ \vdots & \\ N & \text{if } \boldsymbol{w}_k^2 \sim \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(N)}, \boldsymbol{Q}_k^{2,(N)}) \end{cases}$$

where $p(I_k = 1) = p_1, \ldots, p(I_k = N) = p_N$, we can note that (3.39)-(3.40) conditional on $\boldsymbol{x}_{1:k}^2$ and $I_{1:k}$ reduces to a LG system for which the KF is the optimal estimator. Intuitively, the random variable I_k indicates the *effective* distribution of $\boldsymbol{w}_k^2$ at time index k. Thus, as before, we can use the optimal KF to obtain the Gaussian PDF $p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})$, and use the PF to approximate $p(\boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k})$:

$$p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k}) = \underbrace{p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})}_{Kalman filter} \underbrace{p(\boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k})}_{Particle filter}. \tag{3.43}$$

We proceed as Section 3.2, and begin with the derivation of $p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})$. As mentioned before, conditional on $\boldsymbol{x}_{1:k}^2$, $I_{1:k-1}$, and $I_k = n$, (3.39)-(3.40) form a LG system in $\boldsymbol{x}_k^1$ for which the KF is the optimal estimator. Note that, n is the specific realization of I_k at time index k. Thus, as discussed in Section 2.2.1 on Kalman filtering, $p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})$, $p(\boldsymbol{x}_k^2 | \boldsymbol{x}_{1:k-1}^2, I_{1:k})$, and $p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k-1}^2, I_{1:k})$ are all Gaussian distributions

that satisfy

$$p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2, I_{1:k-1}, I_k = n) = \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k}^1, \boldsymbol{P}_{k|k}^1) \tag{3.44}$$

$$p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, I_{1:k-1}, I_k = n) = \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^2, \boldsymbol{S}_{k|k-1}) \tag{3.45}$$

$$p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2, I_{1:k-1}, I_k = n) = \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k-1}^1, \boldsymbol{P}_{k|k-1}^1) \tag{3.46}$$

where

$$\boldsymbol{x}_{k|k}^1 = \boldsymbol{x}_{k|k-1}^1 + \boldsymbol{P}_{k|k-1}^1 \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T \boldsymbol{S}_{k|k-1}^{-1}(\boldsymbol{x}_k^2 - \boldsymbol{x}_{k|k-1}^2) \tag{3.47}$$

$$\boldsymbol{P}_{k|k}^1 = \boldsymbol{P}_{k|k-1}^1 - \boldsymbol{P}_{k|k-1}^1 \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T \boldsymbol{S}_{k|k-1}^{-1} \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2) \boldsymbol{P}_{k|k-1}^1 \tag{3.48}$$

$$\boldsymbol{x}_{k|k-1}^2 = \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k|k-1}^1 + \overline{\boldsymbol{w}}_k^{2,(I_k=n)} \tag{3.49}$$

$$\boldsymbol{S}_{k|k-1} = \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k|k-1}^1 \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T + \boldsymbol{Q}_k^{2,(I_k=n)} \tag{3.50}$$

$$\boldsymbol{x}_{k|k-1}^1 = \boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k-1|k-1}^1 \tag{3.51}$$

$$\boldsymbol{P}_{k|k-1}^1 = \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k-1|k-1}^1 \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)^T + \boldsymbol{Q}_k^1. \tag{3.52}$$

We emphasize that (3.47)-(3.52) are all dependent on $I_{1:k-1}$, and $I_k = n$. Indeed, the entire past is summarized by the *sufficient statistic* $\boldsymbol{x}_{k|k}^1$, and its associated covariance $\boldsymbol{P}_{k|k}^1$.

Now let us consider the marginal posterior PDF $p(\boldsymbol{x}_{1:k}^2, I_{1:k}|\boldsymbol{y}_{1:k})$. As mentioned before we use the PF to approximate this density, therefore we can write

$$\widehat{p}(\boldsymbol{x}_{1:k}^2, I_{1:k}|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \delta((\boldsymbol{x}_{1:k}^2, I_{1:k}) - (\boldsymbol{x}_{1:k}^2, I_{1:k})^{(i)}) \tag{3.53}$$

where $\tilde{w}_k^{(i)} = [\sum_{j=1}^{N_p} w_k^{(j)}]^{-1} w_k^{(i)}$ is the *i-th* normalized weight, and $w_k^{(i)}$ satisfies

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)}) p(\boldsymbol{x}_k^{2,(i)}, I_k^{(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)})}{q(\boldsymbol{x}_k^{2,(i)}, I_k^{(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k)} w_{k-1}^{(i)}. \tag{3.54}$$

In (3.54), the likelihood $p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)})$ can be determined from (3.41) and the statistics of $\boldsymbol{e}_k$, i.e.,

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)}) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}(\boldsymbol{x}_k^{2,(i)}), \boldsymbol{R}_k). \tag{3.55}$$

The prior $p(\boldsymbol{x}_k^2, I_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)})$ for each element of $\mathcal{I}^N = \{n|n = 1, \ldots, N\}$ is given by

$$p(\boldsymbol{x}_k^2, I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}) = p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n)p(I_k = n) \quad (3.56)$$

where $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n)$ is given by (3.45), and $p(I_k = n)$ from the *apriori* probabilities of I_k, i.e., $p(I_k = n) = p_n$.

There are numerous choices for the importance function $q(\boldsymbol{x}_k^2, I_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k})$. As usual, we may either select the prior (3.56):

$$q(\boldsymbol{x}_k^2, I_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k}) = p(\boldsymbol{x}_k^2, I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}), \quad (3.57)$$

the OID, or a approximation of the OID for the importance function. Indeed, for the considered DSSM, the OID $p(\boldsymbol{x}_k^2, I_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k)$ is generally analytically intractable. However, if $\boldsymbol{H}(\boldsymbol{x}_k^2)$ is a linear function of $\boldsymbol{x}_k^2$, i.e.,

$$\boldsymbol{H}(\boldsymbol{x}_k^2) = \boldsymbol{H}\boldsymbol{x}_k^2 \quad (3.58)$$

in (3.41), it can be shown that the OID $p(\boldsymbol{x}_k^2, I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k)$ for $\boldsymbol{x}_k^2$ and each element in $\mathcal{I}^N = \{n|n = 1, \ldots, N\}$ satisfies

$$\begin{aligned} q_{opt}(\boldsymbol{x}_k^2, I_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_{1:k}) &= p(\boldsymbol{x}_k^2, I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k) \quad (3.59) \\ &= p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n, \boldsymbol{y}_k) \\ &\quad \times p(I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k) \quad (3.60) \end{aligned}$$

where

$$p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n, \boldsymbol{y}_k) = \mathcal{N}(\boldsymbol{x}_k^2; \hat{\boldsymbol{x}}_k^{2,(i)}, \hat{\boldsymbol{P}}_k^{2,(i)}) \quad (3.61)$$

with

$$\begin{aligned} \hat{\boldsymbol{x}}_k^{2,(i)} &= \boldsymbol{x}_{k|k-1}^{2,(i)} + \boldsymbol{S}_{k|k-1}^{(i)}\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{(i)}\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(i)}) \quad (3.62) \\ \hat{\boldsymbol{P}}_k^{2,(i)} &= \boldsymbol{S}_{k|k-1}^{(i)} - \boldsymbol{S}_{k|k-1}^{(i)}\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{(i)}\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{(i)} \quad (3.63) \end{aligned}$$

and

$$p(I_k = n|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k) \propto p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n)p(I_k = n) \tag{3.64}$$

where $p(\boldsymbol{y}_k|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, I_k = n) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(i)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{(i)}\boldsymbol{H}^T + \boldsymbol{R}_k)$.

Thus in drawing $\{\boldsymbol{x}_k^{2(i)}, I_k^{(i)}\}$ from (3.60), we first sample $I_k^{(i)}$ with probability commensurate with (3.64), and then sample $\boldsymbol{x}_k^{2,(i)}$ in accordance with (3.61), $\boldsymbol{x}_k^{2,(i)} \sim \mathcal{N}(\boldsymbol{x}_k^2; \hat{\boldsymbol{x}}_k^{2,(i)}, \hat{\boldsymbol{P}}_k^{2,(i)})$. Finally, if we proceed as in Section 3.2, it can be shown that

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_{k|k}^{1,(i)} \tag{3.65}$$

$$\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \Big(\boldsymbol{P}_{k|k}^{1,(i)} + (\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]) \times (\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1])^T\Big) \tag{3.66}$$

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_k^{2,(i)} \tag{3.67}$$

$$\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} (\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])(\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])^T \tag{3.68}$$

where $\boldsymbol{x}_{k|k}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)}, I_{1:k}^{(i)})}[\boldsymbol{x}_k^1]$ and $\boldsymbol{P}_{k|k}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)}, I_{1:k}^{(i)})}[\boldsymbol{x}_k^1]$. Recall that $\boldsymbol{x}_{k|k}^{1,(i)}$ and $\boldsymbol{P}_{k|k}^{1,(i)}$ and are computed by (3.47) and (3.48), respectively. Thus, as before, a KF is associated with each particle $(\boldsymbol{x}_k^{2,(i)}, I_k^{(i)})$. The algorithm for the EMKF is summarized below:

Extended Mixture Kalman filter

1. **Initialization**: For $i = 1, \ldots, N_p$, we initialize $\boldsymbol{x}_{0|0}^{1(i)} = \hat{x}_0^1$, $\boldsymbol{P}_{0|0}^{1,(i)} = \hat{P}_0^1$, $\boldsymbol{x}_0^{2,(i)} \sim p(\boldsymbol{x}_0^2)$, and set $w_0^{(i)} = 1$.

2. **New particles**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{k-n:k-1}^{2,(i)} = \boldsymbol{x}_{k-n:k-1}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k-1|k-1}^{1,(i)} = \boldsymbol{x}_{k-1|k-1}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k-1|k-1}^{1,(i)} = \boldsymbol{P}_{k-1|k-1}^{1,(i)}$.

   - Proposal: Draw $(\tilde{\boldsymbol{x}}_k^{2,(i)}, \tilde{I}_k^{(i)}) \sim q(\boldsymbol{x}_k^2, I_k|\tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)}, \tilde{I}_{1:k-1}^{(i)}, \boldsymbol{y}_k)$.

- KF Prediction: Compute $\tilde{\boldsymbol{x}}_{k|k-1}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k-1}^{1,(i)}$ using (3.51) and (3.52), respectively.
- KF Update: Compute $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}$ using (3.47) and (3.48), respectively.

3. **Calculate importance weights**: For $i = 1, \ldots, N_p$, compute importance weights up to a normalizing constant

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)})p(\boldsymbol{x}_k^{2,(i)}, I_k^{(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)})}{q(\boldsymbol{x}_k^{2,(i)}, I_k^{(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}, I_{1:k-1}^{(i)}, \boldsymbol{y}_k)} w_{k-1}^{(i)} \tag{3.69}$$

and normalize importance weights.

4. **Dynamic Resampling**:

If $\hat{N}_{eff} < N_{Th}$,

- Resample $\{\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ w.r.t importance weights to obtain $\{\boldsymbol{x}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{x}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{P}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ and set $w_k^{(i)} = \frac{1}{N_p}$ for $i = 1, \ldots, N_p$.

otherwise

- Set $\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)} = \boldsymbol{x}_{k-n+1:k}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)} = \boldsymbol{x}_{k|k}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)} = \boldsymbol{P}_{k|k}^{1,(i)}$ for $i = 1, \ldots, N_p$.

5. **Estimates**: Compute $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$, and $\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$ using (3.65), (3.66), (3.67), and (3.68), respectively.

6. **Reiterate**: Set $k = k+1$, and go back to step 2.

3.4 Experiments

The considered algorithms are applied to two nonlinear models. The first is the highly nonlinear Univariate Nonstationary Growth model (UVGM) [32, 28, 20, 12],

and the second is the well documented time-varying auto-regressive (TVAR) model. The TVAR model has been investigated by many researchers, and some examples include [18, 22] in which the TVAR model is used for speech processing, and in [16] where the TVAR model is used for modelling of high resolution high range resolution radar signatures.

3.4.1 Univariate Nonstationary Growth Model

The highly nonlinear UVGM is given by

$$x_k = 0.5x_{k-1} + 25\frac{x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2(k-1)) + w_k \tag{3.70}$$

$$y_k = \frac{x_k^2}{20} + e_k, \quad k = 1, \ldots, N \tag{3.71}$$

where $w_k \sim \mathcal{N}(w_k; 0, Q_k)$, $e_k \sim \mathcal{N}(e_k; 0, R_k)$ and $p(x_0) = \mathcal{N}(x_0; \hat{x}_0, \hat{P}_0)$. Here, we choose $Q_k = 1$, $R_k = 1$, $\hat{x}_0 = 0$, $\hat{P}_0 = 1$, and $N = 70$. The considered algorithms are the EKF, Bootstrap PF, and the APF as described in Section 2.2.3, 2.2.4.7, and 3.1, respectively. Each PF uses $N_p = 50$ particles and employs the prior distribution

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k; 0.5x_{k-1} + 25\frac{x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2(k-1)), Q_k) \tag{3.72}$$

for the importance function. Figures 3.1, 3.2, and 3.3 show a typical trajectory of x_k, the corresponding estimated trajectory of $\mathbb{E}_{p(x_k|y_{1:k})}[x_k]$, and an estimate of the 2σ confidence intervals (CI) for the EKF, Bootstrap PF, and APF, respectively.

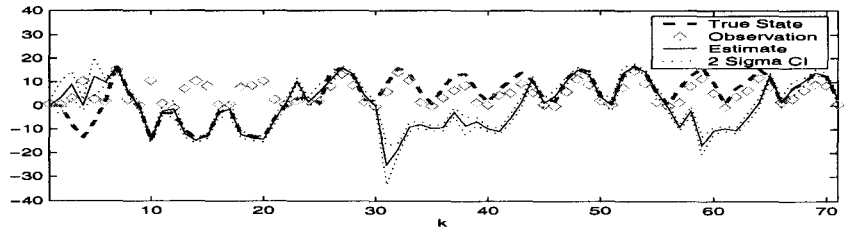


Figure 3.1: True and Estimated state of EKF

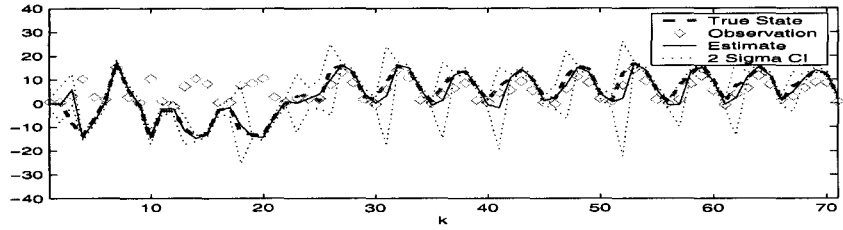


Figure 3.2: True and Estimated state of Bootstrap PF

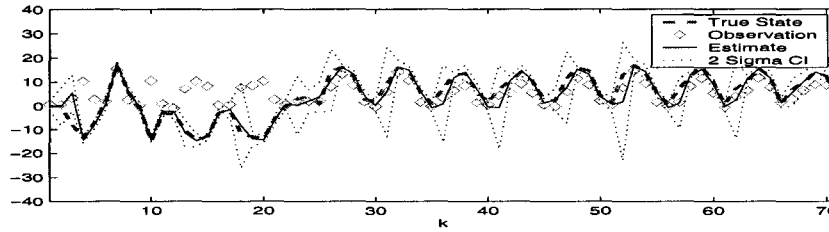


Figure 3.3: True and Estimated state of APF

All the considered PF's outperform the EKF. Evidently, for this highly nonlinear model, the inherent linearizations of the EKF tend to lead to filter divergence. Moreover, the EKF is based upon the assumption that the posterior distribution can be adequately approximated by a Gaussian distribution. In this example and many other scenarios, this is often untrue, and $p(x_k|y_{1:k})$ can actually be multi-modal, or

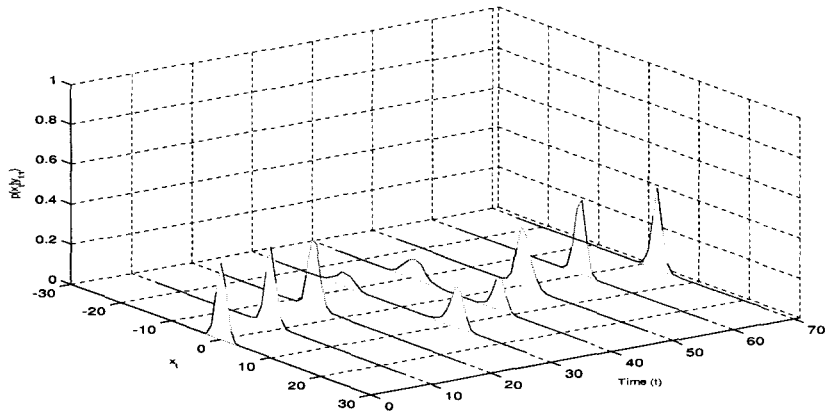


Figure 3.4: APF estimate of $p(x_k|y_{1:k})$

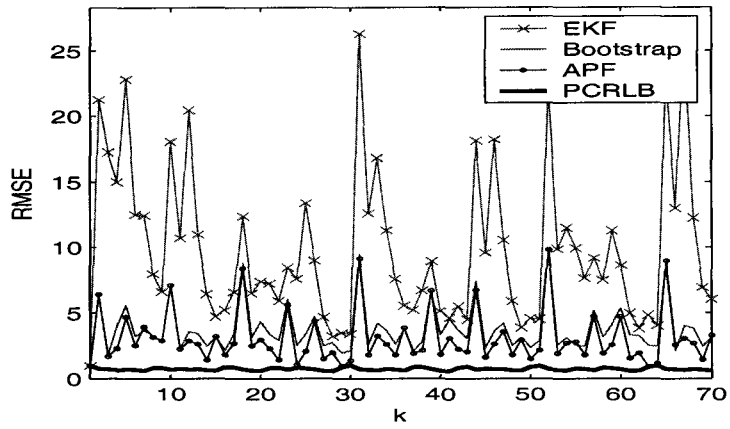


Figure 3.5: Root Mean Square Error curves for UVGM

very skewed. As seen from Figure 3.4, the posterior distribution is actually bimodal for $k = 30$. Unlike the EKF, PF's neither employ a linearization of the DSSM nor a Gaussianity assumption of $p(x_k|y_{1:k})$. Indeed, as illustrated in Figures 3.2, and 3.3, these methods are well suited for state estimation of highly nonlinear, non-Gaussian dynamical systems.

Finally, to compare the error performances of each algorithm, we computed RMSE (square root of MSE) curves for each filter by running all the considered algorithms on the same realizations of data for $M = 200$ experiments. Figure 3.5 illustrates the resulting RMSE curves. For this example, the APF outperforms all the considered algorithms.

3.4.2 Time-Varying Autoregressive Model

The P-th order TVAR model can be written as [18]

$$\boldsymbol{a}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{w}_k^1 \tag{3.73}$$

$$z_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + w_k^2 \tag{3.74}$$

$$y_k = z_k + e_k \tag{3.75}$$

where $\boldsymbol{F} = \beta \boldsymbol{I}_{P\times P}$, $\boldsymbol{a}_k = [a_k^1, \ldots, a_k^P]^T$ are the AR coefficients, z_k is the AR process, $\boldsymbol{G}_k(z_{k-P:k-1}) = [z_{k-1} \ldots z_{k-P}]$, $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$ is the process noise, and $e_k \sim \mathcal{N}(e_k; 0, R_k)$ is the measurement noise. The driving noise w_k^2 for the TVAR model is distributed according to

$$p(w_k^2) = \mathcal{N}(w_k^2; 0, Q_k^2). \tag{3.76}$$

Here, we choose $\boldsymbol{Q}_k^1 = (0.01)^2 \boldsymbol{I}_{P\times P}$, $Q_k^2 = 0.5$, and $R_k = 0.05$. The elements of $\boldsymbol{a}_0$ and $z_{0:P-1}$ are each distributed in accordance to a Gaussian distribution with mean 0, and variance 0.25. In this experiment, we consider a fourth order TVAR model ($P = 4$) with known coefficient $\beta = 0.995$. The considered algorithms are the Bootstrap PF and the MKF as described in Sections 2.2.4.7 and 3.2, respectively. For the Bootstrap PF, we consider two different implementations. The first design uses the prior $p(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1})$ as the importance function $q(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, \boldsymbol{y}_k)$, that is

$$\begin{aligned} q(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k) &= p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k) p(\boldsymbol{a}_k | \boldsymbol{a}_{k-1}) && (3.77) \\ &= \mathcal{N}(z_k; \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k, Q_k^2) \mathcal{N}(\boldsymbol{a}_k; \boldsymbol{F}\boldsymbol{a}_{k-1}, \boldsymbol{Q}_k^1) && (3.78) \end{aligned}$$

The other uses the OID $p(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k)$ for its importance function (See Appendix C for details):

$$q_{opt}(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k) = p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, y_k) p(\boldsymbol{a}_k | \boldsymbol{a}_{k-1}, y_k). \tag{3.79}$$

In (3.79)

$$p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, y_k) = \mathcal{N}(z_k; \hat{z}_k, \hat{P}_k) \tag{3.80}$$

where

$$\begin{aligned} \hat{z}_k &= \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + \frac{Q_k^2}{R_k + Q_k^2}(y_k - \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k) \\ \hat{P}_k &= \left(\frac{1}{R_k} + \frac{1}{Q_k^2}\right)^{-1} \end{aligned} \tag{3.81}$$

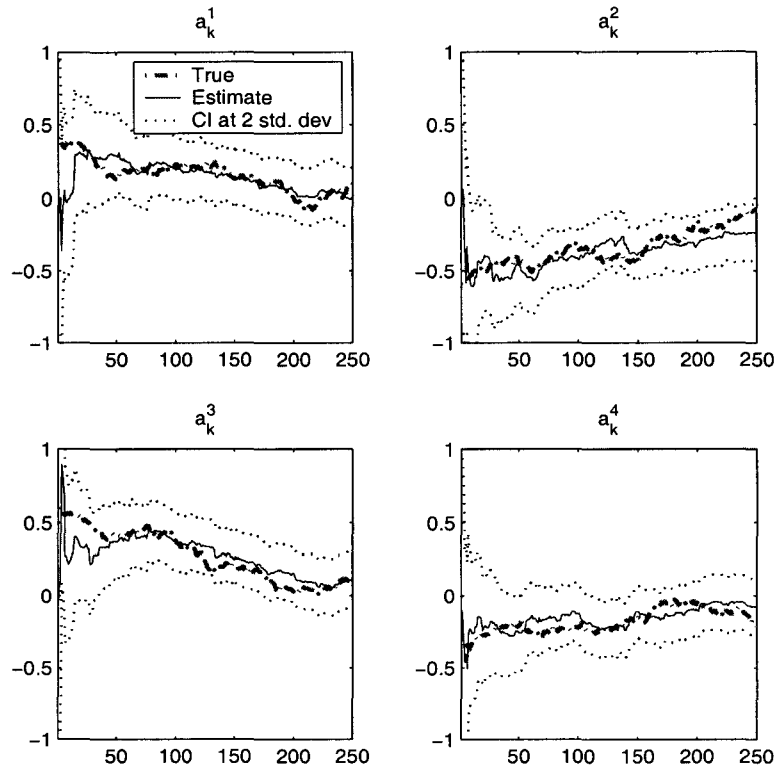


Figure 3.6: True and Estimated a_k using MKF

and

$$p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}, y_k) = \mathcal{N}(\boldsymbol{a}_k; \hat{\boldsymbol{a}}_k, \boldsymbol{\Sigma}_k) \tag{3.82}$$

where

$$\hat{\boldsymbol{a}}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{Q}_k^1 \boldsymbol{G}_k(z_{k-P:k-1})^T [\widehat{\boldsymbol{\Sigma}}_k]^{-1} (y_k - \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{F}\boldsymbol{a}_{k-1}) \tag{3.83}$$

$$\boldsymbol{\Sigma}_k = \boldsymbol{Q}_k^1 - \boldsymbol{Q}_k^1 \boldsymbol{G}_k(z_{k-P:k-1})^T [\widehat{\boldsymbol{\Sigma}}_k]^{-1} \boldsymbol{G}_k(z_{k-P:k-1}) \boldsymbol{Q}_k^1 \tag{3.84}$$

and $\widehat{\boldsymbol{\Sigma}}_k = \boldsymbol{G}_k(z_{k-P:k-1}) \boldsymbol{Q}_k^1 \boldsymbol{G}_k(z_{k-P:k-1})^T + Q_k^2 + R_k$. The OID exploits the current measurement y_k in the proposal of new particles $(z_k, \boldsymbol{a}_k)$. Thus, it is expected to boost the efficiency of the Bootstrap PF.

For the case of the MKF, $\boldsymbol{a}_k$ plays the role of $\boldsymbol{x}_k^1$, and z_k the role of $\boldsymbol{x}_k^2$ as described in Section 3.2. In fact, conditional on $z_{1:k}$, (3.73) and (3.74) form a LG system in

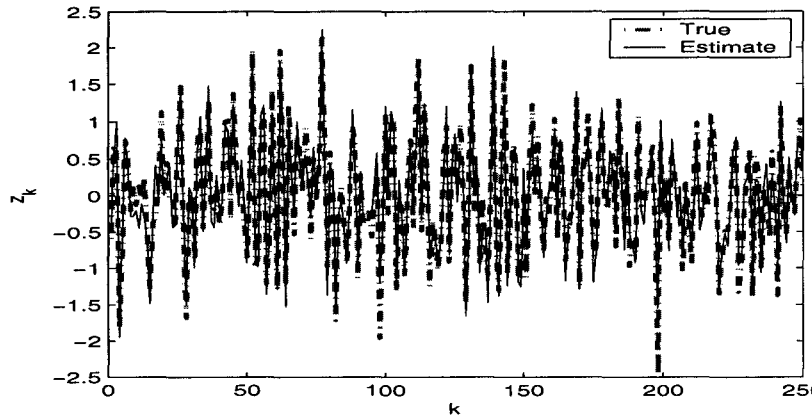


Figure 3.7: True and Estimated z_k using MKF

$\boldsymbol{a}_k$, for which the KF is the optimal estimator. Thus, as discussed in Section 3.2, we estimate $\boldsymbol{a}_k$ via the KF, and track z_k with the PF. For the PF, we choose the marginalized prior $p(z_k|z_{k-1})$ as the importance function.

In this experiment, each PF is implemented with $N_p = 50$ particles. Figures 3.6 and 3.7 show results that are typical of the MKF. The MKF provides good performance. Although there is a fair amount of uncertainty (large CI) about the precise value of $\boldsymbol{a}_k$, it can be seen that the MKF tracks $\boldsymbol{a}_k$ reasonably well. Similarly, for the AR process z_k.

For the MSE calculations, we ran each filter on the same realizations of data, and repeated the experiment $M = 200$ times[1]. Notice that we have also computed the PCRLB (see Appendix D). As shown in Figure 3.8, the MKF outperforms all the considered algorithms. With the exception of the Bootstrap PF using the OID $p(z_k, \boldsymbol{a}_k|z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k)$ for its importance function, no other filter offers comparable performance. Indeed, for the chosen number of particles, the "basic" Bootstrap filter performs poorly. Although it is possible improve performance by increasing the number of particles N_p, the cost of increased computational complexity may result in

[1]The MSE calculations are based only on stable realizations of the TVAR model.

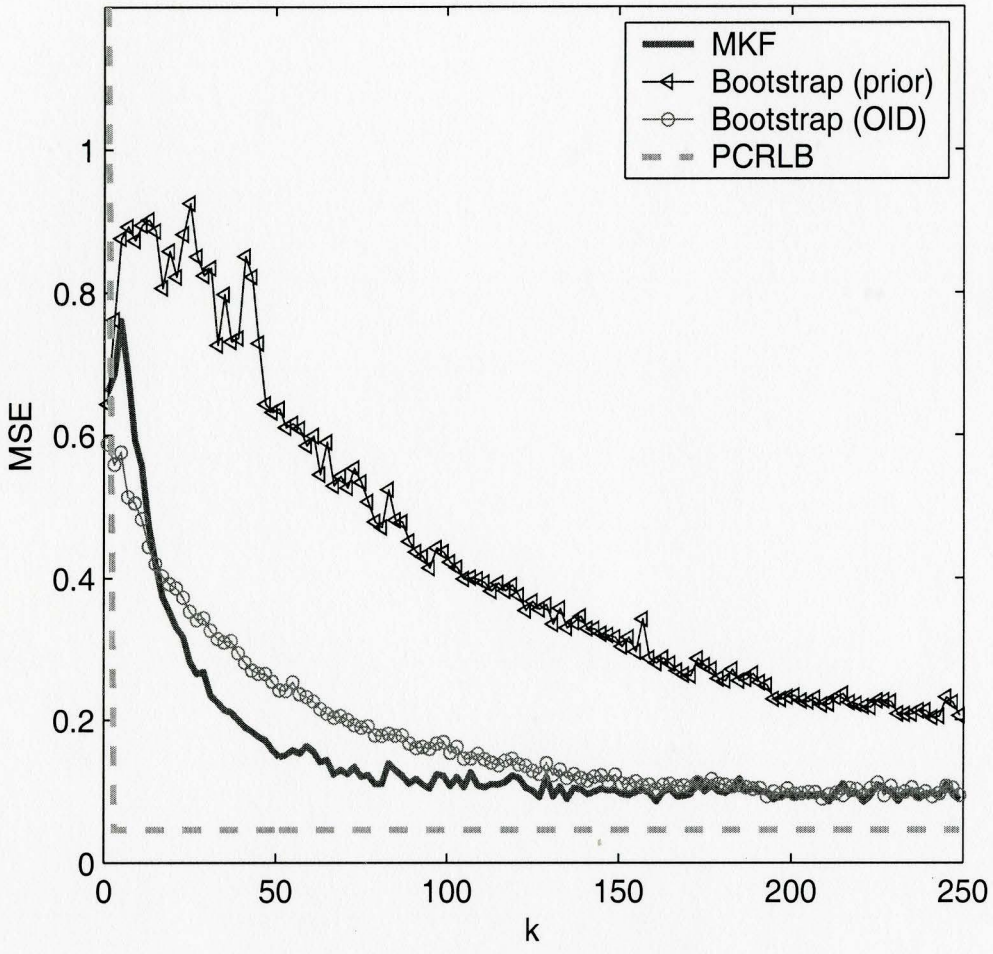


Figure 3.8: Mean Square Error curves for TVAR model

a prohibitively expensive filter.

Thus, in designing a efficient PF, it would seem critical to exploit the most recent measurement y_k, or if possible, exploit the linear sub-structure of the given DSSM. Arguably, in most scenario's, the computational cost of using the OID or the MKF will be more then offset by the reduction in the number of particles required to achieve a certain level of performance [6].

Chapter 4

The Approximate Conditional Mean Particle filter

In this chapter, we introduce a novel particle filter known as the approximate conditional mean particle filter (ACM-PF). As the name suggests, the ACM-PF is a merger between the ACM filter and the PF. To begin, we motivate the proposed algorithm. Subsequently, we provide a derivation of the proposed algorithm. Finally, to validate the algorithm, we apply the ACM-PF to a time-varying autoregressive (TVAR) model that is driven by impulsive noise, and mixture Gaussian noise.

4.1 ACM-PF

The considered DSSM is identical to that which was considered in Section 3.3. For convenience, we reproduce the considered dynamic state space model (DSSM). That is, let $\boldsymbol{x}_k = [\boldsymbol{x}_k^{1T}, \boldsymbol{x}_k^{2T}]^T$, and assume that (2.42)-(2.43) may be written in the form of

a partially observed *non-Gaussian* system:

$$\boldsymbol{x}_k^1 = \boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k-1}^1 + \boldsymbol{w}_k^1 \tag{4.1}$$

$$\boldsymbol{x}_k^2 = \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_k^1 + \boldsymbol{w}_k^2 \tag{4.2}$$

$$\boldsymbol{y}_k = \boldsymbol{H}(\boldsymbol{x}_k^2) + \boldsymbol{e}_k \tag{4.3}$$

where $\boldsymbol{x}_k^1 \in \mathbb{R}^{n_1}$, $\boldsymbol{x}_k^2 \in \mathbb{R}^{n_2}$ are the unobserved processes and $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ is the noisy observation. Again, $\boldsymbol{F}^1(\cdot)$, $\boldsymbol{A}^1(\cdot)$, $\boldsymbol{F}^2(\cdot)$, $\boldsymbol{A}^2(\cdot)$, and $\boldsymbol{H}(\cdot)$ are known functions with proper dimensions. The process noise $\boldsymbol{w}_k^1$, and measurement noise $\boldsymbol{e}_k$ are assumed to be mutually independent zero-mean Gaussian white noise sequences, $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$ and $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$ [1]. The process noise driving (4.2) is assumed to be a white noise sequence that is distributed according to a N component Gaussian Mixture Model (GMM):

$$p(\boldsymbol{w}_k^2) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(j)}, \boldsymbol{Q}_k^{2,(j)}) \tag{4.4}$$

where $\sum_{j=1}^{N} p_j = 1$. As before, the initial states are assumed to be mutually independent zero-mean Gaussian random variables, $\boldsymbol{x}_0^1 \sim \mathcal{N}(\boldsymbol{x}_0^1; \hat{\boldsymbol{x}}_0^1, \hat{\boldsymbol{P}}_0^1)$ and $\boldsymbol{x}_0^2 \sim \mathcal{N}(\boldsymbol{x}_0^2; \hat{\boldsymbol{x}}_0^2, \hat{\boldsymbol{P}}_0^2)$. The considered DSSM (4.1)-(4.3) is a special case of (2.42)-(2.43). Thus, it is amenable to an application of a "standard" PF such as the Bootstrap PF. Typically, to maintain a certain level of performance, it is necessary to increase the number of particles N_p as the dimensionality of the state vector increases. Thus, for complex problems with a large number of state components (i.e. large dimension), it is necessary to boost the efficiency of the standard PF to achieve an acceptable level of performance. Indeed, as shown in Section 3.4.2, it is beneficial to exploit the observation (i.e. OID), the structure of the DSSM (i.e. EMKF), or both to improve the

[1] Actually, the measurement noise $\boldsymbol{e}_k$ can be arbitrarily distributed so long as the pdf of $\boldsymbol{e}_k$ is known. However, for clarity of presentation, we restrict our analysis to the aforementioned Gaussian distributed measurement noise.

efficiency of the PF. For this particular DSSM, it is very difficult to derive the exact OID, if not impossible. Recall, to exploit the structure of the DSSM (i.e EMKF), we introduce a indicator random variable $I_k \in \mathcal{I}^N = \{n | n = 1, \ldots, N\}$ that satisfies

$$I_k = \begin{cases} 1 & \text{if } \boldsymbol{w}_k^2 \sim \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(1)}, \boldsymbol{Q}_k^{2,(1)}) \\ \vdots & \\ N & \text{if } \boldsymbol{w}_k^2 \sim \mathcal{N}(\boldsymbol{w}_k^2; \overline{\boldsymbol{w}}_k^{2,(N)}, \boldsymbol{Q}_k^{2,(N)}) \end{cases}$$

where $p(I_k = 1) = p_1, \ldots, p(I_k = N) = p_N$. Indeed, we can note that (4.1)-(4.2) conditional on $\boldsymbol{x}_{1:k}^2$ and $I_{1:k}$ form a LG system in $\boldsymbol{x}_k^1$ for which the Kalman filter (KF) is the optimal estimator. As such, we can use the optimal KF to obtain the Gaussian PDF $p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})$, and apply the PF to approximate $p(\boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k})$, i.e.,

$$p(\boldsymbol{x}_k^1, \boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k}) = \underbrace{p(\boldsymbol{x}_k^1 | \boldsymbol{x}_{1:k}^2, I_{1:k})}_{Kalman filter} \underbrace{p(\boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k})}_{Particle filter}. \tag{4.5}$$

Note that we are applying the optimal KF to the conditionally linear states $\boldsymbol{x}_k^1$, and using the PF for the truly nonlinear states $\{\boldsymbol{x}_{1:k}^2, I_{1:k}\}$. Thus, we require a reduced number of particles to achieve a certain level of performance. Yet, it may be possible to further increase efficiency. Indeed, for particle filtering, it is advantageous to reduce the dimensionality of the space in which we draw samples from. Thus, for the considered DSSM, we endeavor to design a novel PF that exploits the structure of the considered DSSM while dispensing with the need to introduce an Indicator random variable I_k. The advantages are clear. By eliminating the need to introduce the Indicator random variable, the task of using a PF to approximate $p(\boldsymbol{x}_{1:k}^2, I_{1:k} | \boldsymbol{y}_{1:k})$ is reduced to one of approximating a lower dimensional PDF $p(\boldsymbol{x}_{1:k}^2 | \boldsymbol{y}_{1:k})$. Intuitively, we require a reduced number of particles to achieve a certain level of performance. Therefore, it is of interest to develop these aforementioned ideas, and in the sequel we proceed with the derivation of the ACM-PF.

To begin, we write $p(\boldsymbol{x}_{1:k}^2, \boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})$ as

$$\begin{aligned} p(\boldsymbol{x}_{1:k}^2, \boldsymbol{x}_k^1|\boldsymbol{y}_{1:k}) &= p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2, \boldsymbol{y}_{1:k})p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}) \quad (4.6) \\ &= p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^2)p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}). \quad (4.7) \end{aligned}$$

As mentioned before, we aim to exploit the linear substructure of the considered DSSM. Thus, we only use the PF to approximate $p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$. Hence, if we draw N_p samples of $\boldsymbol{x}_k^2$ from $q(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k)$, that is $\boldsymbol{x}_k^{2,(i)} \sim q(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^{2,(i)}, \boldsymbol{y}_k)$ for $i = 1, \ldots, N_p$ and update the importance weights $w_k^{(i)}$ as

$$w_k^{(i)} \propto \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)})p(\boldsymbol{x}_k^{2,(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)})}{q(\boldsymbol{x}_k^{2,(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}, \boldsymbol{y}_k)} w_{k-1}^{(i)}, \quad (4.8)$$

we obtain for a PF approximation of $p(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k})$:

$$\hat{p}(\boldsymbol{x}_{1:k}^2|\boldsymbol{y}_{1:k}) = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \delta(\boldsymbol{x}_{1:k}^2 - \boldsymbol{x}_{1:k}^{2,(i)}) \quad (4.9)$$

where $\tilde{w}_k^{(i)} = [\sum_{j=1}^{N_p} w_k^{(j)}]^{-1} w_k^{(i)}$. Moreover, if we proceed as in Section 3.2, it follows that the estimates of interest satisfy

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_{k|k}^{1,(i)} \quad (4.10)$$

$$\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \left(\boldsymbol{P}_{k|k}^{1,(i)} + (\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1])(\boldsymbol{x}_{k|k}^{1,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1])^T \right) \quad (4.11)$$

$$\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \boldsymbol{x}_k^{2,(i)} \quad (4.12)$$

$$\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2] = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} (\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])(\boldsymbol{x}_k^{2,(i)} - \widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2])^T. \quad (4.13)$$

where $\boldsymbol{x}_{k|k}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$ and $\boldsymbol{P}_{k|k}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$ is the mean and covariance of $\boldsymbol{x}_k^1$ conditioned on $\boldsymbol{x}_{1:k}^{2,(i)}$, respectively. As in the case of the EMKF, we aim

to compute $\boldsymbol{x}_{k|k}^{1,(i)}$ and $\boldsymbol{P}_{k|k}^{1,(i)}$ analytically. Recall for the EMKF, the former are computed via the KF. However, in [53] it is shown that $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})$ is a GMM such that the number of mixands increases exponentially with k. Hence, $\boldsymbol{x}_{k|k}^{1,(i)}$ and $\boldsymbol{P}_{k|k}^{1,(i)}$ are obtained through a growing symphony of KF's, each corresponding to one mixand of $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})$. For practical applications, it is infeasible to sequentially in time compute $\boldsymbol{x}_{k|k}^{1,(i)}$ and $\boldsymbol{P}_{k|k}^{1,(i)}$ for all time indexes k. Thus, with the aim of designing a computationally attractive algorithm, we follow [37] and adopt the *Masreliez approximation* as discussed in Section 2.2.2, i.e.,

$$p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2) \approx \mathcal{N}(\boldsymbol{x}_k^1; \hat{\boldsymbol{x}}_{k|k-1}^1, \hat{\boldsymbol{P}}_{k|k-1}^1). \tag{4.14}$$

Under this assumption, we can derive an ACM filter [62, 45] for $\boldsymbol{x}_k^1$. In [37], it is shown that for a linear DSSM with non-Gaussian observation noise distributed in accordance to a GMM, the ACM filter yields near optimal performance. In particular, conditional on $\boldsymbol{x}_{1:k}^2$, (4.1)-(4.2) forms such a model, that is, a linear substructure with non-Gaussian observation noise distributed in accordance to a GMM. Thus, we propose to merge the ACM filter, and the standard PF into a hybrid algorithm called the ACM-PF. Notice, for the *conditionally* linear substructure given by (4.1)-(4.2), $\boldsymbol{x}_k^2$ plays the role of the observation. Therefore, as discussed in Section 2.2.2 on ACM filtering, we can compute an estimate of $\boldsymbol{x}_{k|k}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$ and $\boldsymbol{P}_{k|k}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1]$ according to

$$\hat{\boldsymbol{x}}_{k|k}^{1,(i)} = \hat{\boldsymbol{x}}_{k|k-1}^{1,(i)} + \hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^{2,(i)})^T g_k(\boldsymbol{x}_k^{2,(i)}) \tag{4.15}$$

$$\hat{\boldsymbol{P}}_{k|k}^{1,(i)} = \hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} - \hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^{2,(i)})^T G_k(\boldsymbol{x}_k^{2,(i)}) \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^{2,(i)}) \hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} \tag{4.16}$$

where

$$\hat{\boldsymbol{x}}_{k|k-1}^{1,(i)} = \boldsymbol{F}^1(\boldsymbol{x}_{k-n:k-1}^{2,(i)}) + \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^{2,(i)}) \hat{\boldsymbol{x}}_{k-1|k-1}^{1,(i)} \tag{4.17}$$

$$\hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} = \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^{2,(i)}) \hat{\boldsymbol{P}}_{k-1|k-1}^{1,(i)} \boldsymbol{A}^1(\boldsymbol{x}_{k-n:k-1}^{2,(i)})^T + \boldsymbol{Q}_{k-1}^1 \tag{4.18}$$

and

$$g_k(\boldsymbol{x}_k^{2,(i)}) = -\frac{1}{p(\boldsymbol{x}_k^{2,(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)})}\nabla_{\boldsymbol{x}_k^{2,(i)}} p(\boldsymbol{x}_k^{2,(i)}|\boldsymbol{x}_{1:k-1}^{2,(i)}) \tag{4.19}$$

$$G_k(\boldsymbol{x}_k^{2,(i)}) = \nabla_{\boldsymbol{x}_k^{2,(i)}} g_k(\boldsymbol{x}_k^{2,(i)})^T. \tag{4.20}$$

At this point, it should be understood that $\hat{\boldsymbol{x}}_{k|k}^{1,(i)}$, $\hat{\boldsymbol{P}}_{k|k}^{1,(i)}$ replaces $\boldsymbol{x}_{k|k}^{1,(i)}$, $\boldsymbol{P}_{k|k}^{1,(i)}$ in (4.10), (4.11), respectively. Moreover, although

$$\hat{\boldsymbol{x}}_{k|k}^{1,(i)} \approx \boldsymbol{x}_{k|k}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1] \tag{4.21}$$

$$\hat{\boldsymbol{P}}_{k|k}^{1,(i)} \approx \boldsymbol{P}_{k|k}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k}^{2,(i)})}[\boldsymbol{x}_k^1] \tag{4.22}$$

$$\hat{\boldsymbol{x}}_{k|k-1}^{1,(i)} \approx \boldsymbol{x}_{k|k-1}^{1,(i)} = \mathbb{E}_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^{2,(i)})}[\boldsymbol{x}_k^1] \tag{4.23}$$

$$\hat{\boldsymbol{P}}_{k|k-1}^{1,(i)} \approx \boldsymbol{P}_{k|k-1}^{1,(i)} = cov_{p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^{2,(i)})}[\boldsymbol{x}_k^1], \tag{4.24}$$

in the sequel, we will abuse notation and write $\hat{\boldsymbol{x}}_{k|k}^{1,(i)}$, $\hat{\boldsymbol{P}}_{k|k}^{1,(i)}$, $\hat{\boldsymbol{x}}_{k|k-1}^{1,(i)}$, $\hat{\boldsymbol{P}}_{k|k-1}^{1,(i)}$ as $\boldsymbol{x}_{k|k}^{1,(i)}$, $\boldsymbol{P}_{k|k}^{1,(i)}$, $\boldsymbol{x}_{k|k-1}^{1,(i)}$, $\boldsymbol{P}_{k|k-1}^{1,(i)}$, respectively.

Now, to complete the derivation of the ACM-PF, we will obtain each of the component PDF's found in the weight recursion given by (4.8). Clearly, by virtue of (4.3) and the statistics of $\boldsymbol{e}_k$, the likelihood $p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)})$ is given by

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)}) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}(\boldsymbol{x}_k^{2,(i)}), \boldsymbol{R}_k). \tag{4.25}$$

However, unlike the likelihood $p(\boldsymbol{y}_k|\boldsymbol{x}_k^{2,(i)})$, the *marginalized* prior $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$ is a GMM such that the number of mixands increases exponentially with k [53]. Therefore, to limit computational complexity, we again make use of the *Masreliez approximation*, and thus, derive a finite dimensional approximation of $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$, that is (see Appendix E for derivation)

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{S}_{k|k-1}^{2,(j)}) \tag{4.26}$$

where

$$\boldsymbol{x}_{k|k-1}^{2,(j)} = \overline{\boldsymbol{w}}_k^{2,(j)} + \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k|k-1}^1 \tag{4.27}$$

$$\boldsymbol{S}_{k|k-1}^{2,(j)} = \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k|k-1}^1\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T + \boldsymbol{Q}_k^{2,(j)}. \tag{4.28}$$

For the importance function $q(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k)$, we have numerous candidates, but usual choices are to either select the marginalized prior density $\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$, or the OID $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k)$. The marginalized prior density is readily given by (4.26), and the OID for (4.1)-(4.3) in its most general form is analytically intractable. However, if $\boldsymbol{H}_k(\boldsymbol{x}_k^2)$ is a linear function of $\boldsymbol{x}_k^2$, that is

$$\boldsymbol{H}_k(\boldsymbol{x}_k^2) = \boldsymbol{H}\boldsymbol{x}_k^2, \tag{4.29}$$

it is possible to derive a finite dimensional approximation of the OID $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k)$. In particular, again making use of the *Masreliez approximation*, it can be shown that (see Appendix F for derivation)

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k) = \sum_{j=1}^{N} \overline{p}_j \mathcal{N}(\boldsymbol{x}_k^2; \hat{\boldsymbol{x}}_k^{2,(j)}, \hat{\boldsymbol{P}}_k^{2,(j)}) \tag{4.30}$$

where

$$\overline{p}_j = \frac{p_j \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T + \boldsymbol{R}_k)}{\sum_{n=1}^{N} p_n \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(n)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(n)}\boldsymbol{H}^T + \boldsymbol{R}_k)} \tag{4.31}$$

$$\hat{\boldsymbol{x}}_k^{2,(j)} = \boldsymbol{x}_{k|k-1}^{2,(j)} + \boldsymbol{W}_k^{(j)}(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(j)}) \tag{4.32}$$

$$\hat{\boldsymbol{P}}_k^{2,(j)} = \boldsymbol{S}_{k|k-1}^{2,(j)} - \boldsymbol{W}_k^{(j)}\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)} \tag{4.33}$$

and $\boldsymbol{W}_k^{(j)} = \boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}$. Recall that $\boldsymbol{x}_{k|k-1}^{2,(j)}$, $\boldsymbol{S}_{k|k-1}^{2,(j)}$ are computed by (4.27), (4.28), respectively. The suboptimal importance function (4.30) exploits the information in the most recent observation $\boldsymbol{y}_k$. Thus it incorporates additional information into the proposal of new particles $\boldsymbol{x}_k^{2,(i)}$, and thereby, improves the efficiency of the ACM-PF. The algorithm for the ACM-PF is summarized as follows:

ACM-PF

---

1. **Initialization**: For $i = 1, \ldots, N_p$, we initialize the particles, $\boldsymbol{x}_0^{2,(i)} \sim p(\boldsymbol{x}_0^2)$, $\boldsymbol{x}_{0|0}^{1,(i)} = \hat{\boldsymbol{x}}_0^1$, $\boldsymbol{P}_{0|0}^{1,(i)} = \hat{\boldsymbol{P}}_0^1$ and set $w_0^{(i)} = \frac{1}{N_p}$.

2. **New particles**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{k-n:k-1}^{2,(i)} = \boldsymbol{x}_{k-n:k-1}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k-1|k-1}^{1,(i)} = \boldsymbol{x}_{k-1|k-1}^{1,(i)}$, $\tilde{\boldsymbol{P}}_{k-1|k-1}^{1,(i)} = \boldsymbol{P}_{k-1|k-1}^{1,(i)}$.

   (a) Proposals: Draw $\tilde{\boldsymbol{x}}_k^{2,(i)} \sim q(\boldsymbol{x}_k^2 | \tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)}, \boldsymbol{y}_{1:k})$.

   (b) ACM prediction: Compute $\tilde{\boldsymbol{x}}_{k|k-1}^{1,(i)}$, $\tilde{\boldsymbol{P}}_{k|k-1}^{1,(i)}$ using (4.17), and (4.18), respectively.

   (c) ACM update: Compute $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}$, $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}$ using (4.15), and (4.16), respectively.

3. **Calculate Importance Weights**: For $i = 1, \ldots, N_p$, evaluate the importance weights up to a normalizing constant

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\boldsymbol{y}_k | \tilde{\boldsymbol{x}}_k^{2,(i)}) \hat{p}(\tilde{\boldsymbol{x}}_k^{2,(i)} | \tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)})}{q(\tilde{\boldsymbol{x}}_k^{2,(i)} | \tilde{\boldsymbol{x}}_{1:k-1}^{2,(i)}, \boldsymbol{y}_{1:k})}$$

   and normalize importance weights.

4. **Dynamic Resampling**:

   If $\hat{N}_{eff} < N_{Th}$,

   - Resample $\{\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{x}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\tilde{\boldsymbol{P}}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ w.r.t importance weights to obtain $\{\boldsymbol{x}_{k-n+1:k}^{2,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{x}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$, $\{\boldsymbol{P}_{k|k}^{1,(i)}\}_{i=1}^{N_p}$ and set $w_k^{(i)} = \frac{1}{N_p}$ for $i = 1, \ldots, N_p$.

   otherwise

   - Set $\tilde{\boldsymbol{x}}_{k-n+1:k}^{2,(i)} = \boldsymbol{x}_{k-n+1:k}^{2,(i)}$, $\tilde{\boldsymbol{x}}_{k|k}^{1,(i)} = \boldsymbol{x}_{k|k}^{1,(i)}$, and $\tilde{\boldsymbol{P}}_{k|k}^{1,(i)} = \boldsymbol{P}_{k|k}^{1,(i)}$ for $i = 1, \ldots, N_p$.

5. **Estimates**: Compute $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{cov}_{p(\boldsymbol{x}_k^1|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^1]$, $\widehat{\mathbb{E}}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$, and $\widehat{cov}_{p(\boldsymbol{x}_k^2|\boldsymbol{y}_{1:k})}[\boldsymbol{x}_k^2]$ using (4.10), (4.11), (4.12), and (4.13), respectively.

6. **Reiterate**: Set $k = k+1$, and go back to step 2.

Remark 3. *Recall that the ACM filter (i.e. (4.15) to (4.18)) reduces to the KF when* $\mathbf{w}_k^2$ *is Gaussian distributed. Hence, it can be seen that the ACM-PF is equivalent to the EMKF if* $\mathbf{w}_k^2 \sim \mathcal{N}(\mathbf{w}_k^2; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ *where* $\boldsymbol{\mu}$ *and* $\boldsymbol{\Sigma}$ *are the mean and covariance, respectively.*

4.2 Simulations: Non-Gaussian TVAR model

It is a well known fact that many random processes of interest can be successfully modelled by AR models that are driven by Gaussian distributed noise (see [27] for a detailed treatment). Unfortunately, for some phenomena, this approach may not be entirely appropriate. For instance, in seismology and underwater acoustics, we often find large impulses/spikes in the observed signal (see [51] and the references within). In such scenarios, a Gaussian PDF with its sharp roll off cannot adequately model these impulses. Therefore, it is of interest to consider driving noises that are distributed according to non-Gaussian distributions. Moreover, we may aim to model non-stationarities. Therefore, we consider a TVAR model that is driven by non-Gaussian noise [11], in particular, driving noise that is distributed according to a GMM.

Now, we present the considered DSSM. In fact, the following DSSM is almost identical to that which was considered in Section 3.4.2. That is,

$$\boldsymbol{a}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{w}_k^1 \tag{4.34}$$

$$z_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + w_k^2 \tag{4.35}$$

$$y_k = z_k + e_k, \quad k = 1, \ldots, n \tag{4.36}$$

where $\boldsymbol{F} = \beta \boldsymbol{I}_{P\times P}$, $\boldsymbol{a}_k = [a_k^1, \ldots, a_k^P]^T$ are the AR coefficients, z_k is the AR process, $\boldsymbol{G}_k(z_{k-P:k-1}) = [z_{k-1} \ldots z_{k-P}]$, $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$ is the process noise, and $e_k \sim \mathcal{N}(e_k; 0, R_k)$ is the measurement noise. However, unlike the TVAR model that was considered in Section 3.4.2, we consider driving noise w_k^2 that is distributed according to

$$p(w_k^2) = \sum_{j=1}^{N} p_j \mathcal{N}(w_k^2; \overline{w}_k^{2,(j)}, Q_k^{2,(j)}) \tag{4.37}$$

where $\sum_{j=1}^{N} p_j = 1$. Here, we choose $\boldsymbol{Q}_k^1 = (0.01)^2 \boldsymbol{I}_{P\times P}$, and $R_k = 1$. The elements of $\boldsymbol{a}_0$ and $z_{0:P-1}$ are each distributed in accordance to a Gaussian distribution with mean 0, and variance 0.5. As before, we consider a fourth order TVAR model ($P = 4$) with known coefficient $\beta = 0.995$. It should be noted that the considered DSSM is a special case of (4.1)-(4.3). That is, $\boldsymbol{a}_k$ plays the role of $\boldsymbol{x}_k^1$, and z_k the role of $\boldsymbol{x}_k^2$ in (4.1)-(4.3).

The implemented algorithms are the EMKF, the ACM-PF, the Bootstrap PF as described in Sections 3.3, 4.1, and 2.2.4.7, respectively. For the Bootstrap PF, we implemented two schemes that are discussed as follows. The first scheme employs the prior importance function $p(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1})$, and the second scheme uses the OID for its importance function (derivation is similar to that which was presented in Appendix C):

$$q_{opt}(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k) = p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, y_k) p(\boldsymbol{a}_k | \boldsymbol{a}_{k-1}, y_k). \tag{4.38}$$

In the above

$$p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, y_k) \propto \sum_{j=1}^{N} p_j \mathcal{N}(y_k; \theta_k^{(j)}, Q_k^{2,(j)} + R_k) \mathcal{N}(z_k; \hat{z}_k^{(j)}, \hat{P}_k^{(j)}) \tag{4.39}$$

where

$$\hat{z}_k^{(j)} = \theta_k^{(j)} + \frac{Q_k^{2,(j)}}{(Q_k^{2,(j)} + R_k)}(y_k - \theta_k^{(j)}) \tag{4.40}$$

$$\hat{P}_k^{(j)} = \left(\frac{1}{R_k} + \frac{1}{Q_k^{2,(j)}}\right)^{-1} \tag{4.41}$$

$$\theta_k^{(j)} = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + \overline{w}_k^{2,(j)} \tag{4.42}$$

and

$$p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}, y_k) \propto \sum_{j=1}^{N} p_j \mathcal{N}(y_k; \phi_k^{(j)}, \widehat{\Sigma}_k^{(j)}) \mathcal{N}(\boldsymbol{a}_k; \hat{\boldsymbol{a}}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)}) \tag{4.43}$$

where

$$\phi_k^{(j)} = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{F}\boldsymbol{a}_{k-1} + \overline{w}_k^{2,(j)} \tag{4.44}$$

$$\widehat{\Sigma}_k^{(j)} = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{Q}_k^1\boldsymbol{G}_k(z_{k-P:k-1})^T + Q_k^{2,(j)} + R_k \tag{4.45}$$

$$\hat{\boldsymbol{a}}_k^{(j)} = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{W}^{(j)}(y_k - \phi_k^{(j)}) \tag{4.46}$$

$$\boldsymbol{\Sigma}_k^{(j)} = \boldsymbol{Q}_k^1 - \boldsymbol{W}^{(j)}\boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{Q}_k^1 \tag{4.47}$$

and $\boldsymbol{W}^{(j)} = \boldsymbol{Q}_k^1\boldsymbol{G}_k(z_{k-P:k-1})^T[\widehat{\boldsymbol{\Sigma}}_k^{(j)}]^{-1}$. The importance function in (4.38) exploits the current measurement $\boldsymbol{y}_k$ in the proposal of new particles. Thus, it increases the efficiency of the Bootstrap PF.

Both the EMKF and the ACM-PF used the prior distribution for the importance function. However, in designing an efficient PF, it may be prudent to exploit both the observation y_k and the structure of the DSSM; therefore, we also implemented a ACM-PF and an EMKF that uses (4.30) and the OID (3.60) for the importance function, respectively.

We conducted two experiments, each with $n = 250$ observations. As usual, we adopt the MSE criterion to gauge the performance of each algorithm. In particular, we consider two such metrics. The first is the MSE at the k-th time step defined as

$$MSE_k = \frac{1}{M}\sum_{i=1}^{M}\left(\|\boldsymbol{a}_k^i - \hat{\boldsymbol{a}}_{k|k}^i\|_2^2 + \|z_k^i - \hat{z}_{k|k}^i\|_2^2\right) \tag{4.48}$$

where $\|\cdot\|_2$ is the Euclidean norm, $\hat{\boldsymbol{a}}_{k|k}^i$ and $\hat{z}_{k|k}^i$ is an estimate of $\boldsymbol{a}_k^i$ and z_k^i for the i-th Monte Carlo simulation. The second is the average MSE that is computed by

$$\overline{MSE} = \frac{1}{n}\sum_{k=1}^{n} MSE_k. \tag{4.49}$$

Note, in calculating (4.48) and hence (4.49), we ran each filter on the same realizations of data and repeated the experiment $M = 200$ times.

4.2.1 TVAR model driven by GMM noise

For the first experiment, the driving noise w_k^2 is distributed according to

$$p(w_k^2) = \epsilon\mathcal{N}(w_k^2; -3, 1) + (1-\epsilon)\mathcal{N}(w_k^2; 8, 1)$$

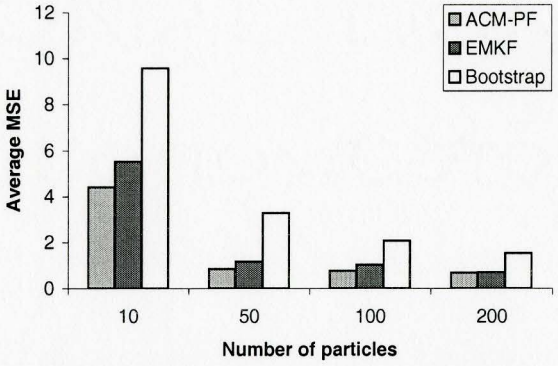
where $\epsilon = 0.8$. Figure 4.1 shows the average MSE for $N_p = 10, 50, 100$ and 200 particles.

Bootstrap PF

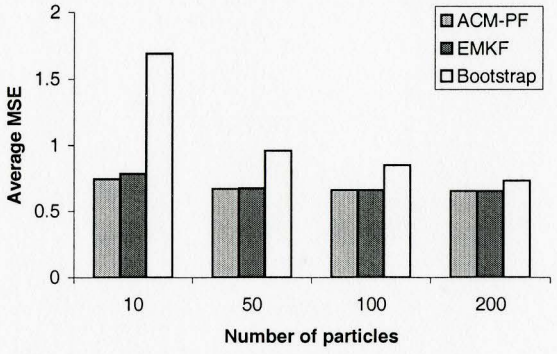
For $N_p = 10$, the Bootstrap PF performs poorly. As expected, we improve performance as we increase the number of particles N_p. Alternatively, we may increase the efficiency of the Bootstrap PF via the OID given by (4.38). As shown in Figure 4.1(b), the Bootstrap filter using the OID, yields much improved performance over the Bootstrap filter using the prior as the importance function. Indeed, by virtue of a more sophisticated importance function, we take advantage of the recent observation, and hence improve the efficiency of the Bootstrap PF.

EMKF

The EMKF yields much improved performance over the Bootstrap PF. As expected, the use of the OID (3.60) also improves performance.



(a) PF's using the prior for the importance function.



(b) PF's using more sophisticated importance functions (i.e. exploits observation).

Figure 4.1: Average MSE curves

ACM-PF

The ACM-PF performs better than the EMKF. As compared to the Bootstrap PF, the ACM-PF yields much improved performance. For example, as shown in Figure 4.1(b), the ACM-PF merely uses 10 particles to attain a MSE of 0.7, while in the case of the Bootstrap PF, we require 200 particle to achieve a comparable level of performance, a twenty-fold increase in N_p. Furthermore, notice that for $N_p \geq 50$, a more sophisticated importance function (4.30) does not significantly improve the

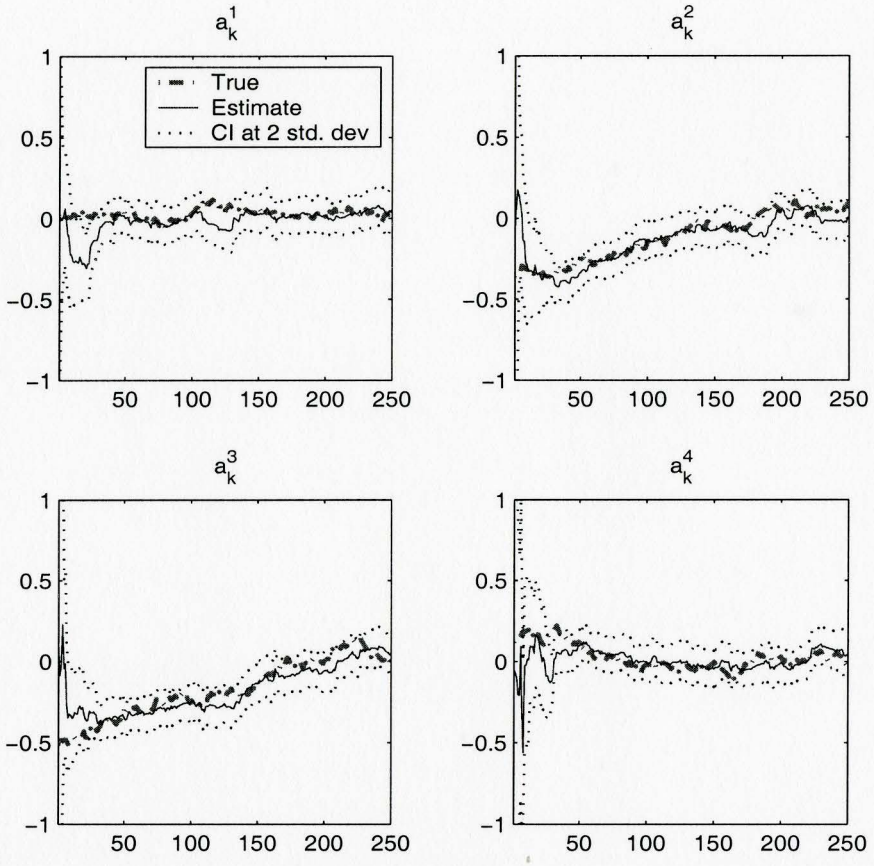


Figure 4.2: True and estimated trajectory of $\boldsymbol{a}_k$ via ACM-PF with $N_p = 50$ particles and the prior for the importance function.

performance of the ACM-PF. We observe similar trends for the case of the EMKF.

Thus, it seems sufficient to simply choose the computationally attractive prior (4.26) for the importance distribution. Indeed, the computational requirements for a more sophisticated importance function (4.30) are considerable, so any subsequent improvement should be significant enough to justify its use. For this particular DSSM, it would seem unnecessary to exploit both the observation, and the structure of the DSSM. Of course, the appropriate decision will depend on the desired tradeoff between

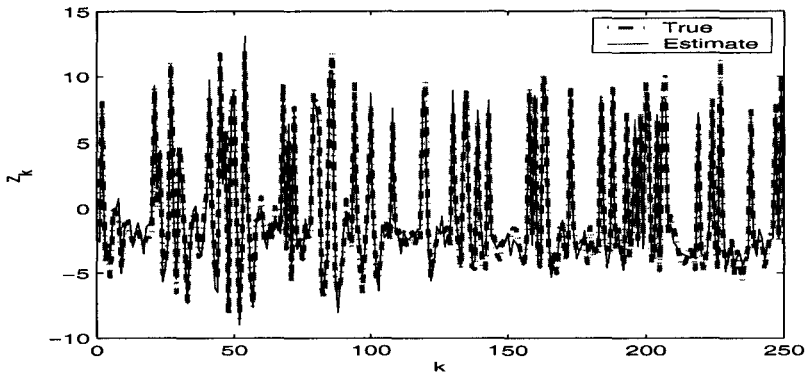
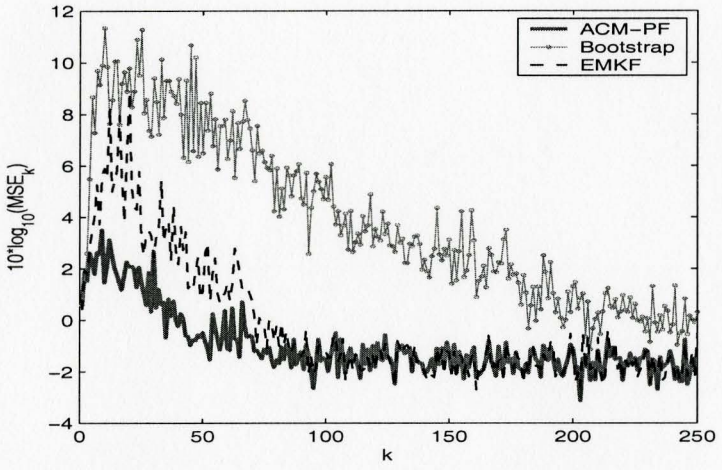


Figure 4.3: True and estimated trajectory of z_k via ACM-PF with $N_p = 50$ particles and the prior for the importance function.

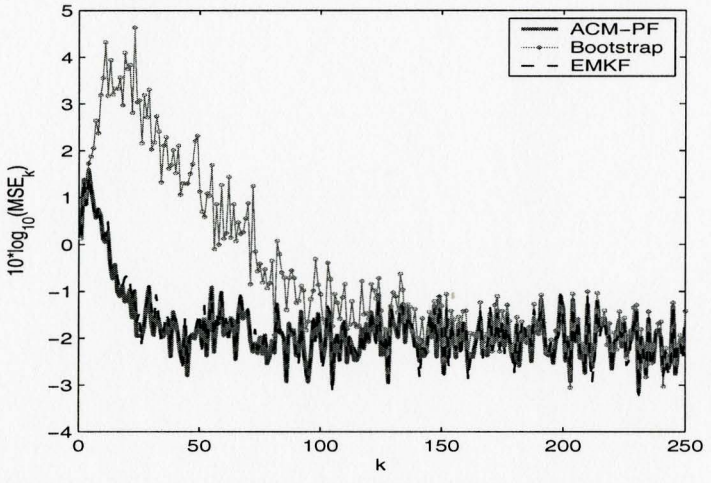
performance and complexity.

However, there are scenarios where it is necessary to jointly exploit both the observation, and the structure of the DSSM. This is the case for a highly nonlinear non-Gaussian dynamical system. In Section 4.2.2, we consider such a scenario. Now, consider Figure 4.2 which shows a true and estimated trajectory of $\boldsymbol{a}_k$. Notice that after 50 observations, the time variations of $\boldsymbol{a}_k$ are closely tracked over time. In fact, the ACM-PF is fairly accurate about the precise value of $\boldsymbol{a}_k$ (i.e. small confidence interval). As shown in Figure 4.3, the AR process z_k is also accurately tracked. In general, the ACM-PF (using the prior importance function) demonstrates good performance. For the case of the ACM-PF using (4.30) for the importance function, we observed a modest improvement in performance. As such, we omit these results.

Now, we investigate the acquisition time for each filter. To this end, we use (4.48) to compute the MSE at each time step. Note, as shown in Figure 4.4, the resulting MSE curves are a function of time. Of the considered algorithms, the ACM-PF yielded the shortest acquisition time, and the smallest steady state MSE. Thus, for this experiment, the ACM-PF offers the best performance. However, one must be careful in making any general conclusions. Indeed, for the majority of the simulations,



(a) PF's using the prior for the importance function.



(b) PF's using more sophisticated importance functions (i.e. exploits observation).

Figure 4.4: MSE curves

the EMKF and the ACM-PF yield comparable steady state MSE, as shown in Figure 4.4. Thus, the performance differences between the ACM-PF and the EMKF can be largely attributed to the apparently large acquisition time of the EMKF.

4.2.2 TVAR model driven by impulsive noise

For the second experiment, the TVAR model is driven by impulsive noise. To model impulsive noise, we follow [1, 61, 66], and assume that w_k^2 has a two-term GMM distribution that is in the form of

$$p(w_k^2) = (1-\epsilon)\mathcal{N}(w_k^2; 0, \sigma_1^2) + \epsilon\mathcal{N}(w_k^2; 0, \sigma_2^2) \tag{4.50}$$

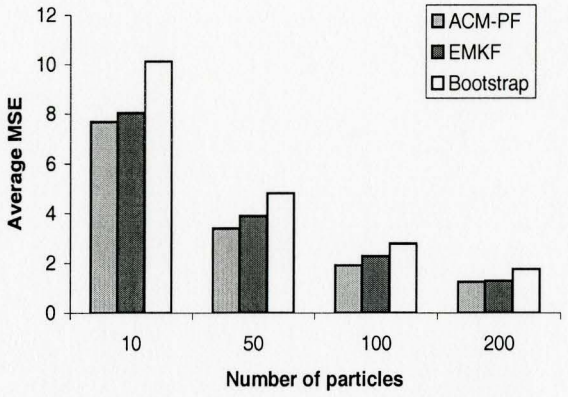
where $0 \leq \epsilon < 1$ and $\sigma_2^2 \gg \sigma_1^2$. In the preceding equation, we choose $\epsilon = 0.1$, $\sigma_1^2 = 1$, and $\sigma_2^2 = 100$. Note that $\mathcal{N}(w_k^2; 0, \sigma_1^2)$ corresponds to the PDF of the nominal background noise and that $\mathcal{N}(w_k^2; 0, \sigma_2^2)$ corresponds to the PDF of the impulsive noise component with ϵ representing the probability that an impulse will occur. Again, we compute the average MSE for the considered experiment. Figure 4.5 shows the average MSE for $N_p = 10, 50, 100$ and 200 particles.

Bootstrap PF

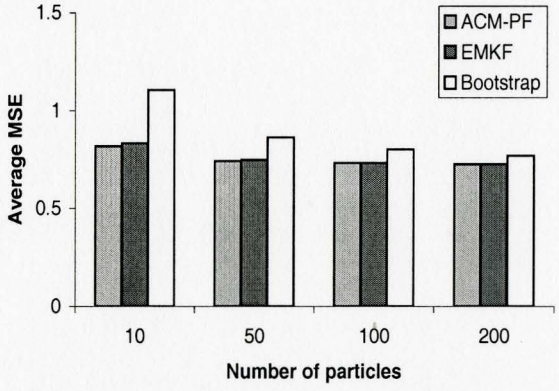
Like the previous experiment, for a small number of particles, the Bootstrap PF performs poorly. As expected, the performance of the filter increases as the number of particles N_p increases. However, if we are to increase the efficiency of the Bootstrap PF, it is clear that we should use the OID (4.38) for the importance function.

EMKF

As shown in Figure 4.5(a), the EMKF significantly outperforms the Bootstrap PF. Of particular interest, is that the use of the OID (3.60) drastically improves the performance of the EMKF.



(a) PF's using the prior for the importance function.

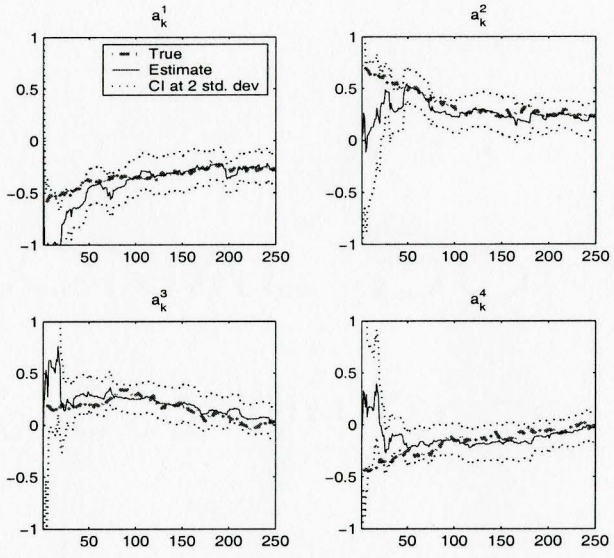


(b) PF's using more sophisticated importance functions (i.e. exploits observation).

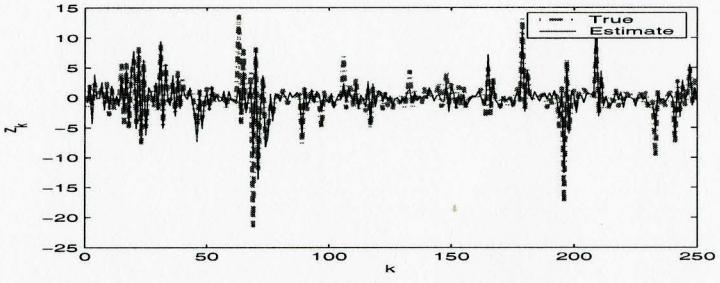
Figure 4.5: Average MSE curves

ACM-PF

In Figure 4.5(b), the ACM-PF merely uses 10 particles to yield a MSE of 0.8, while in the case of the Bootstrap PF, we use at least 100 particles to achieve a similar level of performance, a ten-fold increase in N_p. As compared to the EMKF, the ACM-PF shows marginally improved performance. Of particular interest, is that the



(a) True and tracked trajectory of $\boldsymbol{a}_k$

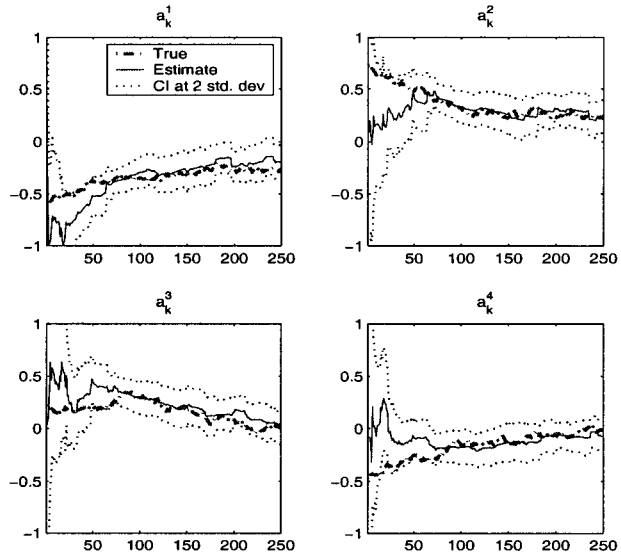


(b) True and tracked trajectory of z_k

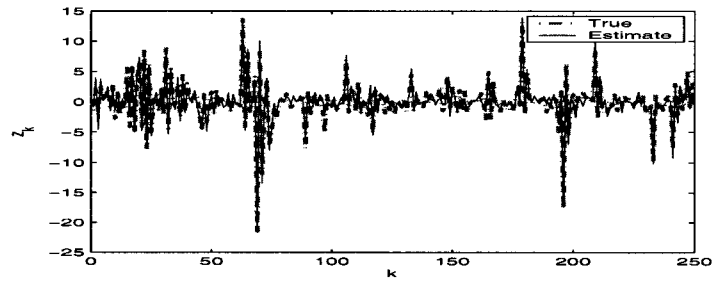
Figure 4.6: True and estimated trajectory of $\boldsymbol{x}_k$ via ACM-PF with $N_p = 50$ particles and the prior for the importance function.

use of a more sophisticated importance function (4.30) also drastically improves the performance of the ACM-PF.

Now, we present some simulations that are typical of the ACM-PF. As shown in Figure 4.6, the ACM-PF (using the prior importance function) tracks $\boldsymbol{a}_k$ reasonably well. Unfortunately, the same cannot be said for the estimated trajectory of z_k. Indeed, on more than one occasion the ACM-PF actually loses track of z_k. For



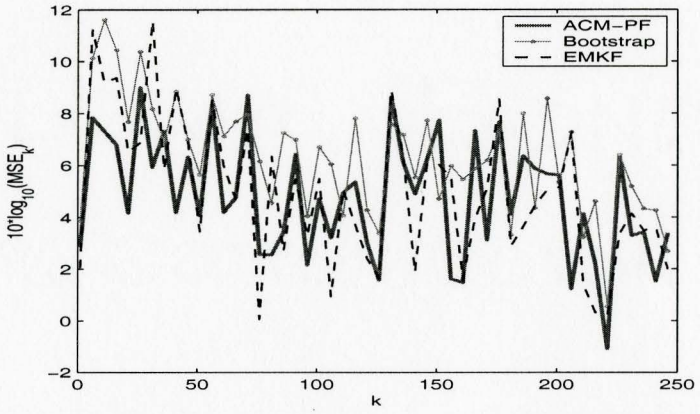
(a) True and tracked trajectory of $\boldsymbol{a}_k$



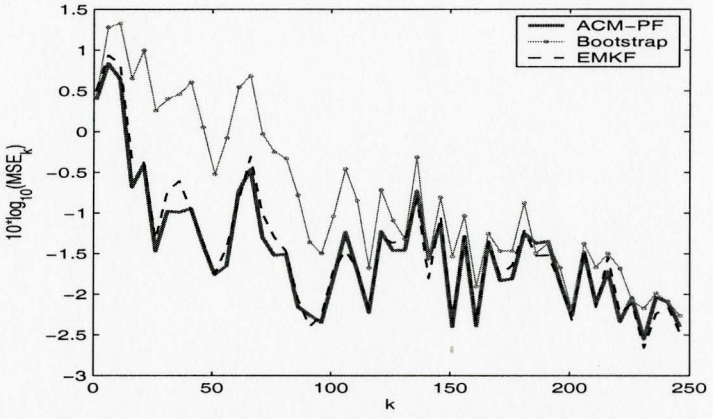
(b) True and tracked trajectory of z_k

Figure 4.7: True and estimated trajectory of $\boldsymbol{x}_k$ via ACM-PF using $N_p = 50$ particles and a more sophisticated importance function.

instance, around $k = 60$ and $k = 70$, the estimated impulse does not even remotely coincide with the true impulse. Evidently, for this particular DSSM, there is considerable motivation to employ a more sophisticated importance function. Indeed, as shown in Figure 4.7, if the ACM-PF uses (4.30) as its importance function, we obtain much improved performance. In fact, for this choice of the importance function, the ACM-PF accurately tracks both the AR coefficients $\boldsymbol{a}_k$ and the AR process z_k. Of



(a) PF's using the prior for the importance function.



(b) PF's using more sophisticated importance functions.

Figure 4.8: MSE curves

particular significance is that the estimated trajectory of z_k closely matches the true trajectory of z_k. Indeed, a more sophisticated importance function (4.30) uses the most recent observation to propose new particles. Thus, it improves the efficiency of the ACM-PF. For the EMKF, we observed similar trends in performance.

Now, we investigate the acquisition time of each filter. As before, we use (4.48) to compute the MSE at each time step. The results are shown in Figure 4.8. As

shown in Figure 4.8(a), the obtained performance is rather poor if each filter uses the simple prior distribution for its importance function. However, if each filter uses a more sophisticated importance function, it can be seen that each filter shows much improved performance. The latter is shown in Figure 4.8(b). Notice that, the ACM-PF performs the best amongst the considered algorithms. Thus, by incorporating the measurement y_k, or in the other words, the observation of an impulse into the proposal of new particles, the PF generates particles in the space of the true z_k, and thereby improves performance. Apparently, for the considered TVAR model, it is necessary to exploit both the measurement and the structure of DSSM in the designing of an efficient PF. This is contrary to the previous experiment in which we only needed to exploit one of the aforementioned techniques.

4.3 Conclusion

In this chapter, we have proposed a novel filter for a class of partially observed *non-Gaussian* DSSM's. The proposed method is a efficient combination of the ACM filter and the particle filter. The considered DSSM consists of a combination of linear and nonlinear states, and a non-Gaussian state evolution noise. Results show that the ACM-PF outperforms the Bootstrap PF, and the EMKF.

Chapter 5

Channel Equalization and Phase Noise Suppression in OFDM Systems Using Particle Filtering

In this chapter, we apply the PF to the problem of channel equalization and phase noise suppression in orthogonal frequency division multiplexing (OFDM) systems. First, we review OFDM, and introduce the problem of channel equalization and phase noise suppression for OFDM systems. Then, we introduce the baseband OFDM system, and develop the required dynamic state space model (DSSM). Subsequently, we present a derivation of the proposed particle filtering algorithm. Finally, we present some simulation results that illustrates the effectiveness of the proposed algorithm.

5.1 Introduction

A frequency-selective channel introduces intersymbol interference (ISI). That is, a symbol experiences interference from other symbols that have been delayed by multipath. Generally, there is severe ISI if the channel delay spread[1] T_m is greater than the symbol time T_s, i.e. $T_m > T_s$. Conversely, when $T_m \ll T_s$ there is insignificant ISI. Note that in most wideband applications, the latter condition cannot be satisfied, that is, there is severe ISI. Thus, it is of interest to eliminate or minimize ISI. Traditionally, complex time domain equalization has been used to counteract ISI. Alternatively, we may consider multicarrier modulation as a effective means for dealing with ISI [46]. Indeed, in the last few years, there has been increasing interest in multicarrier modulation schemes that are robust to ISI; in particular, OFDM.

The basic idea of OFDM is to divide the high rate data stream into N parallel lower rate substreams, each modulating a orthogonal subcarrier that is transmitted in parallel to maintain the total desired data rate. In particular, to minimize ISI, we must ensure that the symbol time of each lower rate substream T is much larger than the delay spread of the channel T_m (i.e. $T \gg T_m$). For OFDM, $T = NT_s$ where T_s is the symbol time of the high rate data stream. Therefore, in practise, we set $N \gg 1$ so that $T \gg T_m$. Consequently, each lower rate substream will not experience significant ISI, or in other words, each lower rate substream undergoes flat fading which in practice is relatively simple to equalize. In fact, if we introduce a cyclic prefix (CP) with proper duration, ISI can be completely removed. In practice, we use efficient FFT/IFFT hardware to implement OFDM. Thus, it can be seen that OFDM offers considerable robustness against multipath fading, and is low in complexity. Indeed, many applications have adopted OFDM, and some include the digital audio broadcasting (DAB), digital video broadcasting (DVB) standards, the

[1]The channel delay spread is the time difference between the first and last component of the impulse response of the channel.

wireless LAN standards, such as IEEE 802.11a, and HiperLAN2.

However, OFDM systems suffer from some drawbacks as well, and one is the increased sensitivity to random phase noise (PN) that is introduced by the local oscillator [44]. PN in OFDM systems causes two effects. The first is a random phase rotation that is common to all subcarriers, that is appropriately referred to as the common phase error (CPE). The second is the introduction of intercarrier interference (ICI), resulting from the loss of orthogonality between each subcarrier. Indeed, many researchers [9, 44, 42, 56] have studied the effects of PN in OFDM systems.

Moreover, schemes for PN compensation in OFDM systems, have been proposed by several authors. In [41, 64], the chosen approach was to counter rotate the received signal constellation, via an estimate of the CPE term. In this chapter, we present a pilot tone aided algorithm that jointly equalizes the channel and compensates for the CPE in a time–varying frequency selective channel. The algorithm is based on the time domain tracking/estimation of the *effective* dynamic channel, i.e., the combined effect of the CPE, and the time-varying frequency selective channel. However, for our estimates of interest, the optimal Bayesian estimators (i.e. MMSE, MAP estimates) are analytically intractable. Hence, for online estimation of the *effective* dynamic channel, we resort to particle filtering. With the aim of designing an efficient PF, we introduce the Auxiliary Mixture Kalman filter (Aux-MKF), and as the name suggests, it is a merger between the MKF, and strategies stemming from the APF. A derivation of the Aux-MKF is presented in Section 5.4. For now, we will begin with a brief review of the baseband OFDM system.

5.2 OFDM System Model

In practice, an OFDM system is implemented as shown in Figure 5.1. At the front end of the transmitter, an information source sends a stream of high rate serial M-QAM

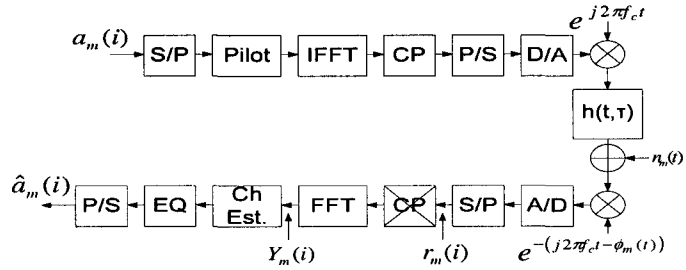


Figure 5.1: Baseband OFDM System

symbols, each of sample period T_s, to the serial/parallel (S/P) converter. Then, a block of N M-QAM symbols is converted to a block of N parallel M-QAM symbols where $a_m(i)$ denotes the M-QAM symbol at the i-th subcarrier of the m-th OFDM symbol. We assume that the power of $a_m(i)$ has been normalized to unity, that is $\mathbb{E}\left[|a_m(i)|^2\right] = 1$.

After S/P conversion, P pilot tones are inserted into $a_m(i)$ such that

$$a_m(i) = \begin{cases} c_{pilot}(i) & i \in \Omega \\ \text{information data} & i \notin \Omega \end{cases}$$

where Ω denotes the set of pilot tone locations. In particular, we choose Ω to satisfy [39]

$$\Omega = \{i | i = kS, \quad \text{with } k = 0, \ldots, P-1\}$$

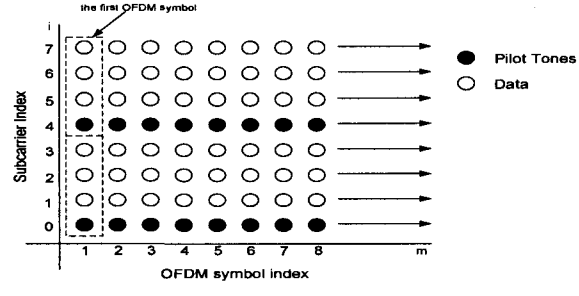


Figure 5.2: Example Pilot Tone Scheme where $S = 4$, $P = 2$ and $N = 8$.

where $S = \frac{N}{P}$ is the spacing between each pilot tone. Now, $\{a_m(i)\}_{i=0}^{N-1}$ are sent to the IDFT to produce

$$s_m(k) = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} a_m(i) e^{\frac{j2\pi ik}{N}}, \qquad k = 0, \ldots, N-1 \tag{5.1}$$

Subsequently, a CP of length N_{cp} is introduced to remove inter-symbol interference (ISI). That is we precede $\{s_m(k)\}_{k=0}^{N-1}$ by $\{s_m(k)\}_{k=-N_{cp}}^{-1}$ where

$$s_m(-k) = s_m(N-k), \qquad k = 1, \ldots, N_{cp} \tag{5.2}$$

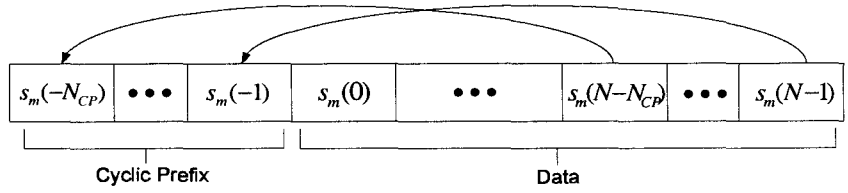


Figure 5.3: Introduction of Cyclic Prefix to enable removal of ISI

Then, $\{s_m(k)\}_{k=-N_{cp}}^{N-1}$ is sent into the D/A converter at a sample rate of $1/T_s$. The result is the m-th transmitted OFDM symbol $s_m(t)$ which can be written as

$$s_m(t) = \sum_{k=-N_{cp}}^{N-1} s_m(k) g(t - kT_s) \tag{5.3}$$

where $g(t)$ is the impulse response of the transmitter D/A converter. We remark that the resulting symbol time of each OFDM symbol is $T_t = T_{CP} + T$ where $T_{CP} = N_{cp}T_s$ and $T = NT_s$ is the duration of the CP and the useful (i.e. data) portion of each OFDM symbol, respectively. Note, as mentioned before $T \gg T_m$.

In this work, the time varying frequency selective channel is assumed to be *quasi-static* during one OFDM symbol T_t. Thus, if we use $h_m(\tau)$ to denote the channel for the m-th OFDM symbol, and assume that $h_m(\tau)$ is represented by a tapped delay model of length L [46], it follows that

$$h_m(\tau) = \sum_{l=0}^{L-1} h_{m,l} \delta(\tau - lT_s) \tag{5.4}$$

where $\delta(\cdot)$ denotes the Dirac delta function, and $h_{m,l}$ denotes the l-th channel tap for the m-th OFDM symbol. The additive white noise is denoted as $n(t)$.

At the receiver, the *free-running* local oscillator introduces phase noise $\phi_m(t)$. Therefore, at the output of the A/D converter, the received m-th OFDM symbol $r_m(t)$ is described by

$$r_m(t) = e^{j\phi_m(t)} \sum_{l=0}^{L-1} h_{m,l} \sum_{k=-N_{cp}}^{N-1} s_m(k) p(t - lT_s - kT_s) + n(t) \tag{5.5}$$

where $p(t) = g(t) * f(t)$ and $f(t)$ is the impulse response of the receiver A/D converter. We assume perfect frequency and timing synchronization and that $p(t)$ satisfies the Nyquist criterion [46]. Therefore, we can write the i-th received sample of the m-th OFDM symbol as

$$r_m(i) = \sum_{l=0}^{L-1} h_{m,l} s_m(i-l) e^{j\phi_m(i)} + n_m(i), \qquad i = -N_{cp}, \ldots, N-1 \tag{5.6}$$

where $\phi_m(i)$ is the i-th sample of the phase noise for the m-th OFDM symbol and $n_m(i)$ is a zero mean complex Gaussian random variable with variance σ_n^2. We make the assumption that the entire channel impulse response lies within the CP, i.e., $N_{cp} \geq L-1$, then discarding of the CP followed by the DFT of $\{r_m(i)\}_{i=0}^{N-1}$ yields

$$\begin{aligned} Y_m(i) &= a_m(i) H_m(i) \underbrace{I_m(0)}_{CPE} \\ &\quad + \underbrace{\sum_{\substack{n=0 \\ n \neq i}}^{N-1} a_m(n) H_m(n) I_m(i-n)}_{ICI} + W_m(i), \qquad i = 0, \ldots, N-1 \end{aligned} \tag{5.7}$$

where

$$H_m(n) = \sum_{l=0}^{L-1} h_{m,l} e^{-j\frac{2\pi nl}{N}} \tag{5.8}$$

$$I_m(n) = \frac{1}{N} \sum_{k=0}^{N-1} e^{j\phi_m(k)} e^{-j\frac{2\pi nk}{N}} \tag{5.9}$$

and $W_m(i) = DFT\{n_m(i)\}$. The multiplicative distortion $I_m(0)$ is given by

$$I_m(0) = \frac{1}{N}\sum_{i=0}^{N-1} e^{j\phi_m(n)}. \tag{5.10}$$

As shown in Section 5.3, it is convenient to approximate $I_m(0)$ with [41]

$$I_m(0) \approx \hat{I}_m(0) = e^{j\theta_m} \tag{5.11}$$

where

$$\theta_m = \frac{1}{N}\sum_{i=0}^{N-1} \phi_m(n). \tag{5.12}$$

Compared to the scenario of an ideal local oscillator (i.e. no PN) where the subcarriers retain their orthogonality, and hence $a_m(i)$ undergoes flat fading, i.e.,

$$Y_m(i) \;=\; a_m(i)H_m(i) + W_m(i), \qquad i = 0, \ldots, N-1 \tag{5.13}$$

it can be seen from (5.7) that PN introduces two problems. The first problem is the additional phase variation of the desired sample $a_m(i)H_m(i)$ by the CPE $arg\{I_m(0)\}$, and the second is the ICI, which results from the loss of orthogonality between each sub-carrier. Moreover, if we define $H_m^{eff}(i) = H_m(i)I_m(0)$ as the *effective* channel response, then it is clear that we must obtain accurate estimates of $H_m^{eff}(i)$ so that we may reliably recover $a_m(i)$. Zero-forcing (ZF) equalization follows after channel estimation, and the transmitted symbol $a_m(i)$ can be estimated by

$$\hat{a}_m(i) \;=\; \frac{Y_m(i)}{\hat{H}_m^{eff}(i)}, \qquad i = 0, \ldots, N-1 \tag{5.14}$$

where $\hat{H}_m^{eff}(i)$ is the estimate of the effective channel response, $H_m^{eff}(i)$. Finally, in this work, we make no attempt in correcting for ICI.

5.3 State Space Model

Particle filters require a process equation and a observation equation. The aim of this section is to develop the required DSSM by exploiting the known statistics of the

channel and PN.

5.3.1 Channel Model

As mentioned before, the frequency selective Rayleigh fading channel is characterized by a tapped-delay model of length L independent taps. However, to fully characterize the channel, we need to introduce an appropriate statistical model for each tap. To this end, we assume that the channel taps $\{h_{m,l}\}_{l=0}^{L-1}$ are mutually uncorrelated, zero mean complex Gaussian random variables. Furthermore, as shown in Figure 5.4, we consider the scenario where the transmitter is fixed, the mobile receiver is moving at a velocity v, and the transmitted signal is scattered by stationary objects surrounding the mobile.

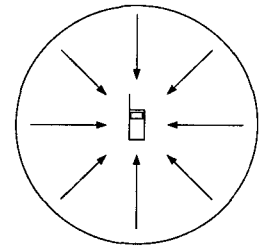


Figure 5.4: Uniform Scattering Environment

With these assumptions, the autocorrelation function (ACF) of the l-th channel tap $r_l(k)$ satisfies Jakes ACF [24]. That is

$$r_l(k) = E[h_{m,l}h^*_{m+k,l}] = r_l(0)J_0(2\pi f_d k T_t), \qquad l = 0, \ldots, L-1 \tag{5.15}$$

where $r_l(0)$ denotes the power of the l-th channel tap, $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind, T_t as mentioned before is the total OFDM symbol time, and f_d denotes the maximum Doppler frequency that is related to the mobile velocity v by

$$f_d = \frac{v f_c}{c} \tag{5.16}$$

where f_c and c denote the carrier frequency and speed of light, respectively. In addition, all L taps of the channel have been normalized such that

$$\sum_{l=0}^{L-1} \mathbb{E}\left[|h_{m,l}|^2\right] = 1. \tag{5.17}$$

However, because the ACF is nonrational, the exact modelling of (5.15) via an autoregressive moving-average (ARMA) model is impossible. Thus, to obtain a useful DSSM, we proceed as in [19] and use a autoregressive (AR) model to approximately capture (5.15). To this end, we adopt an AR(2) model that is given by

$$h_{m,l} = -\gamma_1 h_{m-1,l} - \gamma_2 h_{m-2,l} + v_{m,l}, \qquad l = 0\ ,\ \ldots\ ,\ L\text{-}1 \tag{5.18}$$

where $v_{m,l}$ denotes a zero mean complex Gaussian random variable with variance σ_l^2. To complete the model, we choose coefficients γ_1 and γ_2 such that the ACF of (5.18) closely matches Jakes ACF (5.15). In [63], it is motivated that

$$\begin{aligned} \gamma_1 &= -2r\cos(2\pi f_d T_t/\sqrt{2}) \\ \gamma_2 &= r^2 \end{aligned}$$

where $r \in [0.9, 0.999]$ is the pole radius of the AR(2) model. Finally, if we appeal to the Yule-Walker equations [40] for (5.18), it can be shown that the variance of $v_{m,l}$ satisfies

$$\sigma_l^2 = r_l(0)\frac{(1-\gamma_2)\left((1+\gamma_2)^2 - \gamma_1^2\right)}{(1+\gamma_2)}, \qquad l = 0\ ,\ \ldots\ ,\ L\text{-}1.$$

On the following page, we show a plot of (5.15), and the empirical ACF of (5.18) for 50 lags, $r = 0.965$ and $f_d T_t = 0.035$. The latter ACF has been generated with 20000 samples of the AR(2) model. As shown in Figure 5.5, both ACF's have been normalized to one, and that for high lags (i.e. $k > 15$), the empirical ACF of the AR(2) model does not match the Jakes ACF. However, for effective tracking, it is usually sufficient to match the ACF's at lower lags [30]. Indeed, the considered AR(2) model closely matches Jakes model at lower lags. Therefore, we adopt this model, and thus, proceed to design a filter that will successfully track the target.

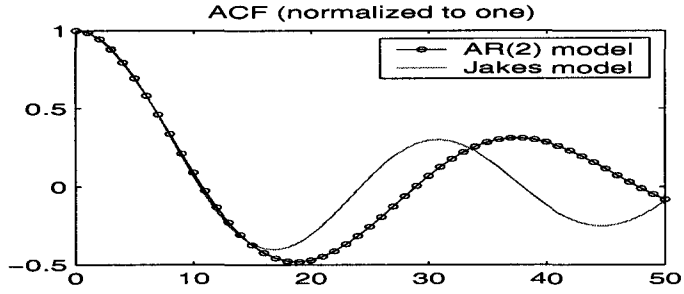


Figure 5.5: ACF of Jakes Model and AR(2) model

5.3.2 Phase Noise Model

The power spectrum of the noisy carrier $c(t) = e^{j\phi_m(t)}$ in (5.5) is shown to be Lorenzian [44, 57, 9]:

$$S_c(f) = \frac{2}{\pi B} \frac{1}{[1 + (\frac{2f}{B})^2]}$$

where B denotes the two-sided 3 dB bandwidth of $S_c(f)$.

As discussed in Section 5.2, the discrete time Wiener PN $\phi_m(n)$ denotes the n-th sample of the PN for the m-th OFDM symbol. In particular, it can be shown that [65]

$$\phi_m(n) = \phi_{m-1}(N-1) + \sum_{i=-N_{CP}}^{n} u[m(N+N_{CP})+i] \tag{5.19}$$

where the $u(i)'s$ denote mutually independent zero mean Gaussian random variables with variance $\sigma_w^2 = 2\pi BT_s = 2\pi BT/N$. We remark that for a fixed T, system performance decreases as BT increases.

Equation (5.12) together with (5.19) result in the desired CPE process equation, that is [41]

$$\theta_m = \theta_{m-1} + \tilde{w}_m \tag{5.20}$$

where $\tilde{w}_n$ is a zero mean Gaussian random variable with variance $\sigma_{cpe}^2 = \left(\frac{2N^2+1}{3N} + N_{cp}\right)\sigma_w^2$.

5.3.3 Observation Model

Our starting point is (5.7). At known pilot tone locations, the least squares (LS) estimate of the effective channel response $H_m^{eff}(n)$ is

$$\hat{H}_m^{eff}(n) = \frac{Y_m(n)}{c_{pilot}(n)}, \quad n \in \Omega \tag{5.21}$$

Stacking $\{\hat{H}_m^{eff}(n)\}_{n\in\Omega}$ into a $P \times 1$ vector $\boldsymbol{H}_{LS}^{eff}$ results in

$$\boldsymbol{H}_{LS}^{eff} = \boldsymbol{V} I_m(0)\boldsymbol{h}_m + \boldsymbol{I}_m + \boldsymbol{Z}_m \tag{5.22}$$

where $\boldsymbol{h}_m = [h_{m,0}, \ldots, h_{m,L-1}]^T$ is a vector of channel taps, $\boldsymbol{I}_m$ is a vector of ICI quantities, $\boldsymbol{Z}_m$ is an AWGN vector, and $\boldsymbol{V}$ is the following Vandermonde DFT Matrix:

$$\boldsymbol{V} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_N^S & \ldots & W_N^{S(L-1)} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & W_N^{(P-1)S} & \ldots & W_N^{(P-1)S(L-1)} \end{bmatrix}$$

where $W_N = e^{-j2\pi/N}$. Assuming $P = L$, the Vandermonde matrix is non-singular and is hence invertible. Therefore, a noisy estimate of $I_m(0)\boldsymbol{h}_m$ is given by

$$\boldsymbol{Y}_m = \boldsymbol{V}^{-1}\boldsymbol{H}_{LS}^{eff} = I_m(0)\boldsymbol{h}_m + \tilde{\boldsymbol{I}}_m + \tilde{\boldsymbol{Z}}_m. \tag{5.23}$$

Equation (5.23) provides a coarse estimate of $I_m(0)\boldsymbol{h}_m$, and thus, can be seen as an observation equation in our DSSM.

5.3.4 Dynamic State Space Model

Use of (5.18), (5.20), and (5.23), leads to the considered DSSM:

$$\boldsymbol{x}_m = \boldsymbol{F}\boldsymbol{x}_{m-1} + \boldsymbol{c}\boldsymbol{v}_m \tag{5.24}$$

$$\theta_m = \theta_{m-1} + \tilde{w}_m \tag{5.25}$$

$$\boldsymbol{Y}_m = \boldsymbol{G}(\theta_m)\boldsymbol{x}_m + \tilde{\boldsymbol{I}}_m + \tilde{\boldsymbol{Z}}_m \tag{5.26}$$

where $\boldsymbol{x}_m = [\boldsymbol{h}_m^T \boldsymbol{h}_{m-1}^T]^T \in \mathbb{C}^{2L\times 1}$. The matrices $\boldsymbol{F} \in \mathbb{R}^{2L\times 2L}$, $\boldsymbol{c} \in \mathbb{R}^{2L\times 1}$, and $\boldsymbol{G}(\theta_m) \in \mathbb{C}^{L\times 2L}$ have the form of

$$\boldsymbol{F} = \begin{bmatrix} -\gamma_1 \boldsymbol{I}_{L\times L} & -\gamma_2 \boldsymbol{I}_{L\times L} \\ \boldsymbol{I}_{L\times L} & \boldsymbol{0}_{L\times L} \end{bmatrix}, \qquad \boldsymbol{c} = \begin{bmatrix} \boldsymbol{I}_{L\times 1} \\ \boldsymbol{0}_{L\times 1} \end{bmatrix}, \qquad \boldsymbol{G}(\theta_m) = \begin{bmatrix} \hat{I}_m(0) \cdot \boldsymbol{I}_{L\times L} & \boldsymbol{0}_{L\times L} \end{bmatrix}$$

where $\boldsymbol{I}_{L\times L}$ and $\boldsymbol{0}_{L\times L}$ denote the $L\times L$ identity and the $L\times L$ zero matrix respectively. We remark that in place of $I_m(0)$, we have substituted the convenient approximation $I_m(0) \approx \hat{I}_m(0) = e^{j\theta_m}$. In addition, $\boldsymbol{v}_m$ denotes a $L \times 1$ vector of white zero mean Gaussian noise with covariance matrix $\boldsymbol{Q}_m = E[\boldsymbol{v}_m \boldsymbol{v}_m^H] = diag(\sigma_0^2, \ldots, \sigma_{L-1}^2)$, whereas $\tilde{\boldsymbol{I}}_m$ and $\tilde{\boldsymbol{Z}}_m$ denote vectors of transformed ICI and noise components respectively. We make the assumptions in [64], so that for $N \gg 1$ the elements of $\tilde{\boldsymbol{I}}_m$ approximately follow a zero mean complex Gaussian distribution with variance $\sigma_I^2 = 2\pi BT/3P$. Furthermore, it can be shown that the elements of $\tilde{\boldsymbol{Z}}_m$ follow a zero mean complex Gaussian distribution with variance $\sigma_Z^2 = \sigma_n^2/P$. Thus, if we conveniently denote the combined effect of $\tilde{\boldsymbol{I}}_m$, and $\tilde{\boldsymbol{Z}}_m$ by $\boldsymbol{e}_m$, that is $\boldsymbol{e}_m = \tilde{\boldsymbol{I}}_m + \tilde{\boldsymbol{Z}}_m$. The PDF of the *effective* measurement noise $\boldsymbol{e}_m$ is given by

$$p(\boldsymbol{e}_m) = \mathcal{N}_c(\boldsymbol{e}_m; \boldsymbol{0}, \boldsymbol{R}_m) \tag{5.27}$$

where $\boldsymbol{R}_m = (\sigma_I^2 + \sigma_Z^2)\boldsymbol{I}_{p\times p}$.

Our main objective is to obtain an approximate minimum mean square error (MMSE) estimate of $\{I_m(0)h_{m,l}\}_{l=0}^{L-1}$, that is

$$\widehat{\mathbb{E}}_{p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m] = \iint \hat{I}_m(0)\boldsymbol{h}_m p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m}) d\theta_m d\boldsymbol{h}_m \tag{5.28}$$

Unfortunately, the posterior PDF $p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m})$ is analytically intractable. Thus, we propose to numerically approximate $p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m})$ via particle filtering, so that we may ultimately compute $\widehat{\mathbb{E}}_{p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m]$, and thus the effective channel response

$$\hat{H}_m^{eff}(i) = DFT\left[\widehat{\mathbb{E}}_{p(\theta_m, \boldsymbol{h}_m | \boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m]\right] \tag{5.29}$$

in (5.14).

Finally, we point out that the observation $\boldsymbol{Y}_m$ is linear in $\boldsymbol{x}_m$, whereas $\boldsymbol{G}(\theta_m)$ is nonlinear in θ_m by virtue of $\hat{I}_m(0)$. This form motivates the development of a MKF type PF which is further discussed below.

5.4 Auxiliary Mixture Kalman Filter

For this particular DSSM, it is possible to design a MKF that yields estimates with lower variances. The idea as described in Section 3.2 is to exploit the inherent linear sub-structure of our given DSSM. Indeed, if we write the joint posterior distribution $p(\boldsymbol{x}_m, \theta_{1:m}|\, \boldsymbol{Y}_{1:m})$ as

$$p(\boldsymbol{x}_m, \theta_{1:m}|\, \boldsymbol{Y}_{1:m}) = p(\boldsymbol{x}_m|\theta_{1:m},\, \boldsymbol{Y}_{1:m})p(\theta_{1:m}|\, \boldsymbol{Y}_{1:m}) \tag{5.30}$$

it is clear that we can obtain the Gaussian PDF $p(\boldsymbol{x}_m|\theta_{1:m},\, \boldsymbol{Y}_{1:m})$, via a Kalman filter (KF), and that we can approximate the marginal posterior distribution $p(\theta_{1:m}|\, \boldsymbol{Y}_{1:m})$ with a PF. Now, at time m, assume for an estimate of $p(\theta_{1:m}|\, \boldsymbol{Y}_{1:m})$ we have

$$\hat{p}(\theta_{1:m}|\, \boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} \delta(\theta_{1:m} - \theta_{1:m}^{(i)}). \tag{5.31}$$

By substituting (5.31) into (5.30), we obtain

$$\hat{p}(\boldsymbol{x}_m, \theta_{1:m}|\, \boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} p(\boldsymbol{x}_m|\theta_{1:m}^{(i)},\, \boldsymbol{Y}_{1:m}) \delta(\theta_{1:m} - \theta_{1:m}^{(i)}). \tag{5.32}$$

Therefore, after marginalizing out $\theta_{1:m-1}$, we obtain for an estimate of $p(\boldsymbol{x}_m, \theta_m|\, \boldsymbol{Y}_{1:m})$:

$$\hat{p}(\boldsymbol{x}_m, \theta_m|\, \boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} p(\boldsymbol{x}_m|\theta_{1:m}^{(i)},\, \boldsymbol{Y}_{1:m}) \delta(\theta_m - \theta_m^{(i)}) \tag{5.33}$$

where the i-th importance weight $w_m^{(i)}$ satisfies

$$w_m^{(i)} \propto w_{m-1}^{(i)} \frac{p(\, \boldsymbol{Y}_m|\theta_{1:m}^{(i)},\, \boldsymbol{Y}_{1:m-1}) p(\theta_m^{(i)}|\theta_{m-1}^{(i)})}{q(\theta_m^{(i)}|\theta_{1:m-1}^{(i)},\, \boldsymbol{Y}_{1:m})}. \tag{5.34}$$

Now, we will evaluate $p(\boldsymbol{x}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m})$ and $p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})$ in (5.33) and (5.34), respectively. The PDF

$$p(\boldsymbol{x}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m}) = \mathcal{N}_c(\boldsymbol{x}_m; \boldsymbol{x}_{m|m}^{(i)}, \boldsymbol{P}_{m|m}^{(i)}) \tag{5.35}$$

in (5.33) is a complex Gaussian PDF with mean $\boldsymbol{x}_{m|m}^{(i)} = E[\boldsymbol{x}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m}]$, and covariance $\boldsymbol{P}_{m|m}^{(i)} = cov[\boldsymbol{x}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m}]$. Similarly, the PDF

$$p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1}) = \mathcal{N}_c(\boldsymbol{Y}_m; \boldsymbol{Y}_{m|m-1}^{(i)}, \boldsymbol{S}_m^{(i)}) \tag{5.36}$$

in (5.34) is a complex Gaussian PDF with mean $\boldsymbol{Y}_{m|m-1}^{(i)} = E[\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1}]$, and covariance $\boldsymbol{S}_m^{(i)} = cov[\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1}]$. In fact, the KF efficiently computes the PDF's $p(\boldsymbol{x}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m})$, and $p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})$ in (5.33) and (5.34) respectively. Thus, for each sample $\theta_m^{(i)}$ we compute (5.35) and (5.36) via a KF:

$$\boldsymbol{x}_{m|m}^{(i)} = \boldsymbol{x}_{m|m-1}^{(i)} + \boldsymbol{W}_m^{(i)}(\boldsymbol{Y}_m - \boldsymbol{Y}_{m|m-1}^{(i)}) \tag{5.37}$$

$$\boldsymbol{P}_{m|m}^{(i)} = \boldsymbol{P}_{m|m-1}^{(i)} - \boldsymbol{W}_m^{(i)}\boldsymbol{S}_m^{(i)}\boldsymbol{W}_m^{(i)H} \tag{5.38}$$

$$\boldsymbol{Y}_{m|m-1}^{(i)} = \boldsymbol{G}(\theta_m^{(i)})\boldsymbol{x}_{m|m-1}^{(i)} \tag{5.39}$$

$$\boldsymbol{S}_m^{(i)} = \boldsymbol{G}(\theta_m^{(i)})\boldsymbol{P}_{m|m-1}^{(i)}\boldsymbol{G}(\theta_m^{(i)})^H + \boldsymbol{R}_m \tag{5.40}$$

$$\boldsymbol{W}_m^{(i)} = \boldsymbol{P}_{m|m-1}^{(i)}\boldsymbol{G}(\theta_m^{(i)})\left(\boldsymbol{S}_m^{(i)}\right)^{-1} \tag{5.41}$$

where

$$\boldsymbol{x}_{m|m-1}^{(i)} = \boldsymbol{F}\boldsymbol{x}_{m-1|m-1}^{(i)} \tag{5.42}$$

$$\boldsymbol{P}_{m|m-1}^{(i)} = \boldsymbol{F}\boldsymbol{P}_{m-1|m-1}^{(i)}\boldsymbol{F}^H + \boldsymbol{c}\boldsymbol{Q}_m\boldsymbol{c}^T. \tag{5.43}$$

Therefore, it is apparent that a KF is associated with each particle $\theta_m^{(i)}$, and that the MKF utilizes a bank of KFs to approximate the true joint posterior distribution (see (5.33)).

Moreover, in order to minimize computational complexity, we choose the prior $p(\theta_m|\theta_{m-1})$ as the importance function, that is $q(\theta_m|\theta_{1:m-1}, \boldsymbol{Y}_{1:m}) = p(\theta_m|\theta_{m-1})$. In

this case, the importance weights given by (5.34) simplify to

$$w_m^{(i)} \propto w_{m-1}^{(i)} p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1}) \tag{5.44}$$

where $p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})$ is given by (5.36).

We observe, however, that the prior $p(\theta_m|\theta_{m-1})$ is inefficient; it proposes samples $\{\theta_m^{(i)}\}_{i=1}^{N_p}$ without any knowledge of the current observation $\boldsymbol{Y}_m$. Thus, with the aim of minimizing computational complexity and to incorporate the recent observation into the proposal of new particles, we follow a strategy that is inspired by the APF [13]. As discussed in Section 3.1, the APF attempts to improve the quality of particles at time m, by preselecting (resampling) the particles at time *m-1* with probability close to $p(\theta_{m-1}|\boldsymbol{Y}_{1:m})$. Thus, we proceed to derive a close approximation of the generally intractable $p(\theta_{m-1}|\boldsymbol{Y}_{1:m})$. The following derivation is conceptually similar to that which was presented in Section 3.1. Therefore, we only provide a brief derivation of $p(\theta_{m-1}|\boldsymbol{Y}_{1:m})$. To begin, we expand $p(\theta_{1:m-1}|\boldsymbol{Y}_{1:m})$ as

$$\begin{aligned} p(\theta_{1:m-1}|\boldsymbol{Y}_{1:m}) &= \int p(\theta_{1:m}|\boldsymbol{Y}_{1:m}) d\theta_m \\ &\propto \int p(\boldsymbol{Y}_m|\theta_{1:m}, \boldsymbol{Y}_{1:m-1}) p(\theta_m|\theta_{m-1}) p(\theta_{1:m-1}|\boldsymbol{Y}_{1:m-1}) d\theta_m \end{aligned} \tag{5.45}$$

At time *m-1*, a PF estimate of $p(\theta_{1:m-1}|\boldsymbol{Y}_{1:m-1})$ is given by

$$\widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m-1}) = \sum_{i=1}^{N_p} \tilde{w}_{m-1}^{(i)} \delta(\theta_{1:m-1} - \theta_{1:m-1}^{(i)}). \tag{5.46}$$

Hence, by substituting (5.46) into (5.45) it follows that

$$\begin{aligned} \widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m}) &\propto \sum_{i=1}^{N_p} \left(\int p(\boldsymbol{Y}_m|\theta_m, \theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1}) p(\theta_m|\theta_{m-1}^{(i)}) d\theta_m \right) \\ &\qquad \times \tilde{w}_{m-1}^{(i)} \delta(\theta_{1:m-1} - \theta_{1:m-1}^{(i)}) \\ &= \sum_{i=1}^{N_p} p(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1}) \tilde{w}_{m-1}^{(i)} \delta(\theta_{1:m-1} - \theta_{1:m-1}^{(i)}) \end{aligned} \tag{5.47}$$

The evaluation of

$$p(\boldsymbol{Y}_m|\theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) = \int p(\boldsymbol{Y}_m|\theta_m, \theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) p(\theta_m|\theta^{(i)}_{m-1}) d\theta_m \tag{5.48}$$

in (5.47) is difficult, since $p(\boldsymbol{Y}_m|\theta_{1:m}, \boldsymbol{Y}_{1:m-1})$ as given by (5.36) depends on θ_m via a nonlinear measurement function $\boldsymbol{G}(\theta_m)$.

To circumvent this difficulty, we again follow the lead of [43], which under the assumption that $\sigma^2_{cpe} \ll 1$, suggested that $p(\theta_m|\theta_{m-1})$ may be adequately characterized by μ_m, where μ_m is either a sample or mode of $p(\theta_m|\theta_{m-1})$. For PN, $\sigma^2_{cpe} \ll 1$, i.e., the variance of PN is much less than one [64]. Thus, by assuming that $p(\theta_m|\theta_{m-1}) \approx \delta(\theta_m - \mu_m)$, the predictive likelihood can be approximated by

$$\begin{aligned} \widehat{p}(\boldsymbol{Y}_m|\theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) &= \int p(\boldsymbol{Y}_m|\theta_m, \theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) \delta(\theta_m - \mu^{(i)}_m) d\theta_m \\ &= p(\boldsymbol{Y}_m|\theta_m = \mu^{(i)}_m, \theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}). \end{aligned} \tag{5.49}$$

If we adopt the approximation given by (5.49), and substitute into (5.47), we obtain a proportionality for an approximation of $\widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m})$:

$$\begin{aligned} \widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m}) &\propto \sum_{i=1}^{N_p} \widehat{p}(\boldsymbol{Y}_m|\theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) \tilde{w}^{(i)}_{m-1} \delta(\theta_{1:m-1} - \theta^{(i)}_{1:m-1}) \\ &= \sum_{i=1}^{N_p} \lambda^{(i)}_m \delta(\theta_{1:m-1} - \theta^{(i)}_{1:m-1}) \end{aligned} \tag{5.50}$$

where $\lambda^{(i)}_m = \widehat{p}(\boldsymbol{Y}_m|\theta^{(i)}_{1:m-1}, \boldsymbol{Y}_{1:m-1}) \tilde{w}^{(i)}_{m-1}$. We note that $\int \widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m}) d\theta_{1:m-1} = 1$, and therefore

$$\widehat{p}(\theta_{1:m-1}|\boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{\lambda}^{(i)}_m \delta(\theta_{1:m-1} - \theta^{(i)}_{1:m-1}) \tag{5.51}$$

where $\tilde{\lambda}^{(i)}_m = [\sum_{j=1}^{N_p} \lambda^{(j)}_m]^{-1} \lambda^{(i)}_m$. Finally by marginalizing out $\theta_{1:m-2}$, we obtain an approximation of $p(\theta_{m-1}|\boldsymbol{Y}_{1:m})$ which is in the form of

$$\widehat{p}(\theta_{m-1}|\boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{\lambda}^{(i)}_m \delta(\theta_{m-1} - \theta^{(i)}_{m-1}). \tag{5.52}$$

Equation (5.52) forms the basis of the Aux-MKF, and it implies that the most promising particles at time *m-1*, will have the largest associated predictive likelihoods $\lambda_m^{(i)} = \widehat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})\tilde{w}_{m-1}^{(i)}$. Moreover, if we rewrite (5.44) as

$$w_m^{(i)} \propto \underbrace{w_{m-1}^{(i)}\widehat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})}_{\lambda_m^{(i)}} \frac{p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})}{\widehat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})} \tag{5.53}$$

then (5.52) and (5.53) suggest that we should preselect (resample) the particles $\{\theta_{m-1}^{(i)}\}_{i=1}^{N_p}$ according to the so-called first stage importance weights, i.e.,

$$\lambda_m^{(i)} \propto \tilde{w}_{m-1}^{(i)}\widehat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1}) \tag{5.54}$$

and that after preselecting, set the weights to the so-called second stage importance weights:

$$w_m^{(i)} \propto \frac{p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})}{\widehat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})}. \tag{5.55}$$

What remains is to derive an expression for an estimate of effective dynamic channel $\mathbb{E}_{p(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m})}[\hat{I}_m(0)\boldsymbol{h}_m]$. Recall that $\boldsymbol{x}_m = [\boldsymbol{h}_m^T \boldsymbol{h}_{m-1}^T]^T$; thus, after marginalizing out $\boldsymbol{h}_{m-1}$ from (5.33), we obtain for an estimate of $p(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m})$:

$$\hat{p}(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m}) = \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} p(\boldsymbol{h}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m})\delta(\theta_m - \theta_m^{(i)}). \tag{5.56}$$

If we substitute (5.56) into (5.28), we have

$$\begin{aligned}\widehat{\mathbb{E}}_{p(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m] &= \iint e^{j\theta_m}\boldsymbol{h}_m \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} p(\boldsymbol{h}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m})\delta(\theta_m - \theta_m^{(i)})d\theta_m d\boldsymbol{h}_m \\ &= \sum_{i=1}^{N_p} \int e^{j\theta_m}\tilde{w}_m^{(i)}\boldsymbol{h}_{m|m}^{(i)}\delta(\theta_m - \theta_m^{(i)})d\theta_m \end{aligned} \tag{5.57}$$

where

$$\boldsymbol{h}_{m|m}^{(i)} = \int \boldsymbol{h}_m p(\boldsymbol{h}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m})d\boldsymbol{h}_m. \tag{5.58}$$

We recognize that the KF efficiently evaluates (5.58). That is

$$\boldsymbol{h}_{m|m}^{(i)} = \begin{bmatrix} I_{L\times L} & 0_{L\times L} \end{bmatrix} \boldsymbol{x}_{m|m}^{(i)} \tag{5.59}$$

where $\boldsymbol{x}_{m|m}^{(i)}$ is given by (5.37). Therefore, by substituting (5.59) into (5.57), we obtain for an estimate of the effective dynamic channel:

$$\begin{aligned} \widehat{\mathbb{E}}_{p(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m] &= \sum_{i=1}^{N_p} \int e^{j\theta_m} \tilde{w}_m^{(i)} \boldsymbol{h}_{m|m}^{(i)} \delta(\theta_m - \theta_m^{(i)}) d\theta_m \\ &= \sum_{i=1}^{N_p} \tilde{w}_m^{(i)} e^{j\theta_m^{(i)}} \boldsymbol{h}_{m|m}^{(i)}. \end{aligned} \tag{5.60}$$

The substitution of (5.60) into (5.29) provides the desired estimate of the *effective* channel response $H_m^{eff}(i)$. The Aux-MKF is summarized as follows:

Auxiliary Mixture Kalman filter

1. **Initialization**: For $i = 1, \ldots, N_p$, we initialize the particles, $\theta_0^{(i)} \sim p(\theta_0)$, $\tilde{\boldsymbol{x}}_{0|0}^{(i)} = \boldsymbol{0}$, $\tilde{\boldsymbol{P}}_{0|0}^{(i)} = \boldsymbol{P}_0$ and set $w_0^{(i)} = 1$.

2. **Calculate first stage weights**: For $i = 1, \ldots, N_p$, set $\tilde{\boldsymbol{x}}_{m-1|m-1}^{(i)} = \boldsymbol{x}_{m-1|m-1}^{(i)}$, $\tilde{\boldsymbol{P}}_{m-1|m-1}^{(i)} = \boldsymbol{P}_{m-1|m-1}^{(i)}$, and $\tilde{\theta}_{m-1}^{(i)} = \theta_{m-1}^{(i)}$

   (a) Draw $\mu_m^{(i)} \sim p(\theta_m|\tilde{\theta}_{m-1}^{(i)})$.

   (b) Compute predicted state $\tilde{\boldsymbol{x}}_{m|m-1}^{(i)}$ and predicted covariance $\tilde{\boldsymbol{P}}_{m|m-1}^{(i)}$ using (5.42) and (5.43), respectively.

   (c) Use (5.49) to compute importance weights up to a normalizing constant

   $$\lambda_m^{(i)} \propto w_{m-1}^{(i)} \hat{p}(\boldsymbol{Y}_m|\tilde{\theta}_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})$$

   and normalize importance weights.

3. **Resampling**: Resample $\{\tilde{\boldsymbol{x}}_{m|m-1}^{(i)}, \tilde{\boldsymbol{P}}_{m|m-1}^{(i)}, \tilde{\theta}_{m-1}^{(i)}\}_{i=1}^{N_p}$ w.r.t importance weights $\lambda_m^{(i)}$ to obtain $\{\boldsymbol{x}_{m|m-1}^{(i)}, \boldsymbol{P}_{m|m-1}^{(i)}, \theta_{m-1}^{(i)}\}_{i=1}^{N_p}$.

4. **New particles**: For $i = 1, \ldots, N_p$

   (a) Draw $\theta_m^{(i)} \sim p(\theta_m|\theta_{m-1}^{(i)})$.

   (b) Compute $\boldsymbol{x}_{m|m}^{(i)}$, and $\boldsymbol{P}_{m|m}^{(i)}$ using (5.37) and (5.38), respectively.

5. **Calculate second stage importance weights**: For $i = 1, \ldots, N_p$, compute importance weights up to a normalizing constant

$$w_m^{(i)} \propto \frac{p(\boldsymbol{Y}_m|\theta_{1:m}^{(i)}, \boldsymbol{Y}_{1:m-1})}{\hat{p}(\boldsymbol{Y}_m|\theta_{1:m-1}^{(i)}, \boldsymbol{Y}_{1:m-1})}$$

   and normalize importance weights.

6. **Estimate**: Use (5.60) to compute an estimate of $\mathbb{E}_{p(\theta_m, \boldsymbol{h}_m|\boldsymbol{Y}_{1:m})}[I_m(0)\boldsymbol{h}_m|\boldsymbol{Y}_{1:m}]$. Substitute the former into (5.29) and thus obtain an estimate of the *effective* channel response $H_m^{eff}(i)$.

7. **Reiterate**: Set $m \rightarrow m + 1$, and go back to step 2.

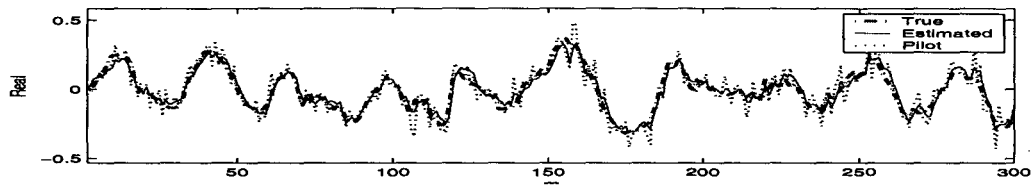
5.5 Simulations

We considered a 16-QAM OFDM system with system parameters $N = 128$, $P = 4$, $N_{cp} = 8$ and total channel bandwidth $B_w = 1MHz$. A four path i.e., $L = 4$ frequency selective channel was generated from Jakes fading model with power delay profile

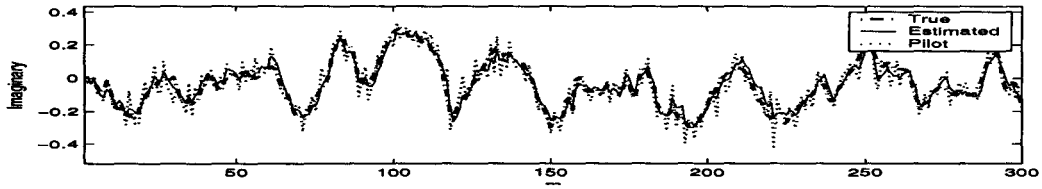
$$\mathbb{E}\left[|h_{m,l}|^2\right] = \mathbb{E}\left[|h_{m,0}|^2\right](1 + 2l/(L-1))^{-c} \tag{5.61}$$

where $c = 3$. This corresponds to $10\log_{10}(\mathbb{E}\left[|h_{m,0}|^2\right]/\mathbb{E}\left[|h_{m,0}|^2\right]) = 0$, $10\log_{10}(\mathbb{E}\left[|h_{m,1}|^2\right]/\mathbb{E}\left[|h_{m,0}|^2\right]) = -13.3$, $10\log_{10}(\mathbb{E}\left[|h_{m,2}|^2\right]/\mathbb{E}\left[|h_{m,0}|^2\right]) = -22.1$, and $10\log_{10}(\mathbb{E}\left[|h_{m,3}|^2\right]/\mathbb{E}\left[|h_{m,0}|^2\right]) = -28.6$.

The adopted delay profile was chosen to be $[0, 1, 2, 3]$ μs and the phase noise rate BT was set to 0.01, which corresponds to conditions of severe phase noise. We

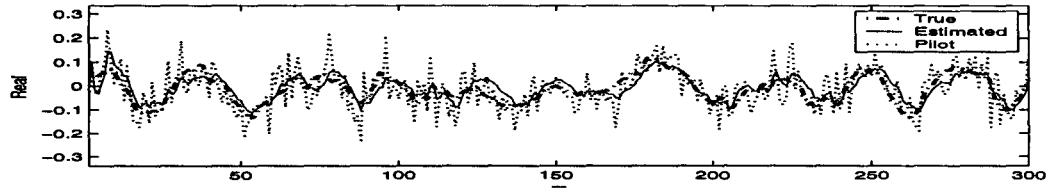


(a) Real

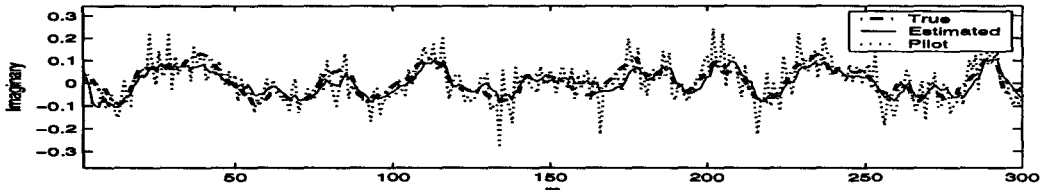


(b) Imaginary

Figure 5.6: True and Tracked Trajectory of $I_m(0)h_{m,2}$



(a) Real



(b) Imaginary

Figure 5.7: True and Tracked Trajectory of $I_m(0)h_{m,3}$

considered a fast fading scenario for which the time-Doppler fading rate f_dT_t was set to 0.04. The proposed algorithm was implemented with $N_p = 50$ particles. Figures 5.6 and 5.7 show typical true and tracked trajectories of $I_m(0)h_{m,2}$ and $I_m(0)h_{m,3}$,

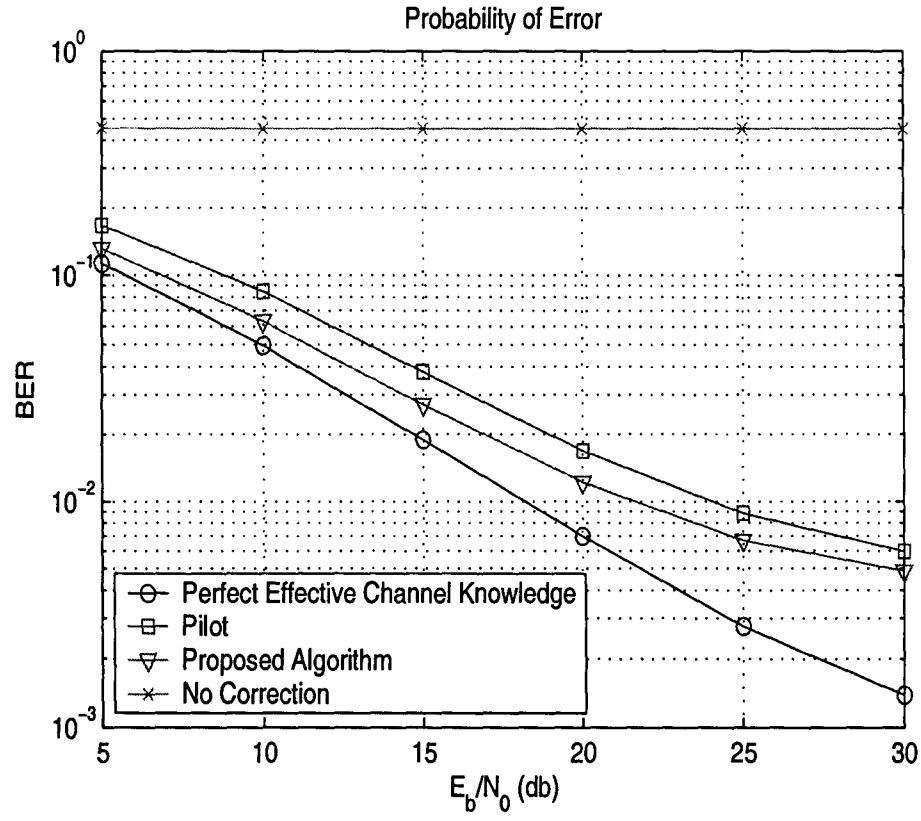


Figure 5.8: BER for 16-QAM

respectively, at a SNR of 10 dB. Evidently, the observed effective channel is very noisy, and that the Aux-MKF tracks the true trajectories reasonably well. We observed similar results for the remaining effective channel taps $I_m(0)h_{m,1}$ and $I_m(0)h_{m,4}$. As such, we do not show these results. The bit error rate (BER) was evaluated at each SNR for 8000 OFDM symbols, and the resulting BER curves are shown in Figure 5.8. It is clear that we must obtain accurate estimates of the *effective* channel response $H_m^{eff}(i)$. With "no correction", we simply cannot recover any useful information. The considered "pilot" tone scheme, that is, we use $\boldsymbol{Y}_m$ in (5.23) for equalization offers intermediate performance. Clearly, the Aux-MKF improves performance, albeit, at a

higher computational complexity. For example, at a SNR of 12 dB, the Aux-MKF is approximately 2 dB away from the ideal curve, while the considered pilot tone scheme is almost 4 dB away. Finally, for the considered application, an increased number of particles N_p, did not result in any noticeable gain in performance.

5.6 Conclusion

In conclusion, we have proposed a new Mixture Kalman filter and Auxiliary particle filtering technique for channel equalization and phase noise suppression in OFDM systems. Results show about $2-3$ dB improvement over a naive scheme based solely on an LS estimate of the effective dynamic channel using the available pilot tones.

Chapter 6

Conclusions

6.1 Conclusion

The aim of this thesis is to motivate the use of particle filtering, and to extend their applications to various problems in wireless communications, and general nonlinear optimal filtering. To this end, we reviewed the general theory of particle filtering, and showed how these methods can be applied to the problem of channel equalization and phase noise suppression in orthogonal frequency division multiplexing (OFDM) systems.

We also addressed the optimal filtering problem for a general class of partially observed *non–Gaussian* dynamic state space models. By doing so, we introduced a novel particle filter, called the approximate conditional mean particle filter, which as the simulations results show, outperform other state–of–the–art particle filtering algorithms.

Traditionally, the EKF and its variants, have been the prescribed solution for nonlinear sequential signal processing. As shown in this thesis, the PF not only improves performance in these applications, but also allows to deal with a larger class of complex nonlinear, non–Gaussian systems that were in most cases out of reach a

few years earlier.

6.2 Contributions to the Scientific Literature

The contributions of this thesis have been published in various conferences. Paper 3 details research not included in this thesis.

1. D. Yee, J.P. Reilly, T. Kirubarajan, "Approximate conditional mean particle filter", to appear in *Proc. IEEE Workshop on Statistical Signal Processing, SSP–2005*, Bordeaux, France, July 17-20, 2005

2. D. Yee, J.P. Reilly, T. Kirubarajan, "Channel equalization and phase noise suppression for OFDM Systems in a time–varying frequency selective channel using particle filtering", *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, 2005, pp. 777–780

3. D. Yee, J.P. Reilly, T. Kirubarajan, "Blind particle filtering for detection in a time–varying frequency selective channel with non-Gaussian noise", to appear in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC–2005*, New York, July 5–8, 2005. *Invited paper.*

6.3 Future Research

In this thesis, we applied particle filtering to solve various problems in nonlinear optimal filtering, and to solve the problem of channel equalization and phase noise suppression in OFDM systems. Now, we discuss a few suggestions for future research:

- Channel equalization and phase noise suppression for OFDM Systems via fully blind particle filtering.

- Verification of the algorithms with real–life data

6.3.1 Channel equalization and phase noise suppression for OFDM Systems via fully blind particle filtering

The proposed solution equalizes in the frequency domain, and uses a pilot tone aided *particle filter* to track/estimate the effective dynamic channel in the time domain. To increase efficiency, we should consider a blind implementation of the proposed algorithm, that is, an algorithm that does not rely on the presence of pilot tones.

6.3.2 Verification of the algorithms with real–life data

Throughout this research, we only used simulated data for the presented experiments. However, real-life data should be collected and used to verify the practical performances of these algorithms.

Appendix A

Derivation of the Kalman filter

Before we begin the derivation of the Kalman filter, we will prove the following identity.

Lemma 1. *If* $\mathbf{x} \in \mathbb{R}^{n_x}$, $\mu \in \mathbb{R}^{n_x}$, $\mathbf{y} \in \mathbb{R}^{n_y}$, *and* $\mathbf{H} \in \mathbb{R}^{n_y \times n_x}$

$$\mathcal{N}(\mathbf{x}; \mu, \Sigma_{xx})\mathcal{N}(\mathbf{y}; \mathbf{H}\mathbf{x}, \Sigma_{yy}) = \mathcal{N}(\mathbf{x}; \mu_{x|y}, \Sigma_{x|y})\mathcal{N}(\mathbf{y}; \mu_y, \Sigma_y) \tag{A.1}$$

where

$$\begin{aligned}
\mu_{x|y} &= \mu + \Sigma_{xx}\mathbf{H}^T(\mathbf{H}\Sigma_{xx}\mathbf{H}^T + \Sigma_{yy})^{-1}(\mathbf{y} - \mathbf{H}\mu) \\
\Sigma_{x|y} &= \Sigma_{xx} - \Sigma_{xx}\mathbf{H}^T(\mathbf{H}\Sigma_{xx}\mathbf{H}^T + \Sigma_{yy})^{-1}\mathbf{H}\Sigma_{xx} \\
\mu_y &= \mathbf{H}\mu \\
\Sigma_y &= \mathbf{H}\Sigma_{xx}\mathbf{H}^T + \Sigma_{yy}
\end{aligned}$$

Proof. The following is based on the derivation given in [54]. To begin, we expand the left hand side of (A.1) as follows:

$$\begin{aligned}
&\mathcal{N}(\boldsymbol{x}; \mu, \Sigma_{xx})\mathcal{N}(\boldsymbol{y}; \boldsymbol{H}\boldsymbol{x}, \Sigma_{yy}) \\
&= (2\pi)^{-n_x/2}|\Sigma_{xx}|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \mu)^T\Sigma_{xx}^{-1}(\boldsymbol{x} - \mu)\right) \\
&\qquad \times (2\pi)^{-n_y/2}|\Sigma_{yy}|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x})^T\Sigma_{yy}^{-1}(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x})\right) .
\end{aligned} \tag{A.2}$$

If we define the following change of variables:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{\mu} \tag{A.3}$$

$$\tilde{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{H}\boldsymbol{\mu}. \tag{A.4}$$

and substitute (A.3)–(A.4) into (A.2), we have

$$\begin{aligned}
&\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}_{xx})\mathcal{N}(\boldsymbol{y};\boldsymbol{H}\boldsymbol{x},\boldsymbol{\Sigma}_{yy}) \\
&= (2\pi)^{(-n_y-n_x)/2}|\boldsymbol{\Sigma}_{xx}|^{-1/2}|\boldsymbol{\Sigma}_{yy}|^{-1/2} \\
&\quad \times \exp\left(-\frac{1}{2}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}-\boldsymbol{H}\tilde{\boldsymbol{x}}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{\Sigma}_{xx}^{-1} & 0\\ 0 & \boldsymbol{\Sigma}_{yy}^{-1}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}-\boldsymbol{H}\tilde{\boldsymbol{x}}\end{bmatrix}\right) \\
&= (2\pi)^{(-n_y-n_x)/2}|\boldsymbol{\Sigma}_{xx}|^{-1/2}|\boldsymbol{\Sigma}_{yy}|^{-1/2} \\
&\quad \times \exp\left(-\frac{1}{2}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ -\boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{\Sigma}_{xx}^{-1} & 0\\ 0 & \boldsymbol{\Sigma}_{yy}^{-1}\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ -\boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}\right).
\end{aligned} \tag{A.5}$$

However, note that

$$\begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ -\boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix} = \begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ \boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix}^{-1}, \tag{A.6}$$

therefore (A.5) can be written as

$$\begin{aligned}
&\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}_{xx})\mathcal{N}(\boldsymbol{y};\boldsymbol{H}\boldsymbol{x},\boldsymbol{\Sigma}_{yy}) \\
&= (2\pi)^{(-n_y-n_x)/2}|\boldsymbol{\Sigma}_{xx}|^{-1/2}|\boldsymbol{\Sigma}_{yy}|^{-1/2} \\
&\quad \times \exp\left(-\frac{1}{2}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}^T\left(\begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ \boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_{xx} & 0\\ 0 & \boldsymbol{\Sigma}_{yy}\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{k\times k} & 0\\ \boldsymbol{H} & \boldsymbol{I}_{g\times g}\end{bmatrix}^T\right)^{-1}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}\right) \\
&= (2\pi)^{(-n_y-n_x)/2}|\boldsymbol{\Sigma}_{xx}|^{-1/2}|\boldsymbol{\Sigma}_{yy}|^{-1/2} \\
&\quad \times \exp\left(-\frac{1}{2}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}^T\begin{bmatrix}\boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xx}\boldsymbol{H}^T\\ \boldsymbol{H}\boldsymbol{\Sigma}_{xx} & \boldsymbol{H}\boldsymbol{\Sigma}_{xx}\boldsymbol{H}^T+\boldsymbol{\Sigma}_{yy}\end{bmatrix}^{-1}\begin{bmatrix}\tilde{\boldsymbol{x}}\\ \tilde{\boldsymbol{y}}\end{bmatrix}\right).
\end{aligned} \tag{A.7}$$

It can be shown that

$$\begin{vmatrix} \Sigma_{xx} & \Sigma_{xx}H^T \\ H\Sigma_{xx} & H\Sigma_{xx}H^T + \Sigma_{yy} \end{vmatrix} = |\Sigma_{xx}||\Sigma_{yy}|. \tag{A.8}$$

Therefore, by substituting (A.3) and (A.4) into (A.7), the joint distribution of $\boldsymbol{x}$ and $\boldsymbol{y}$ can be written as

$$\begin{aligned} p(\boldsymbol{x},\boldsymbol{y}) &= (2\pi)^{(-n_y-n_x)/2}|\Sigma_{xx}|^{-1/2}|\Sigma_{yy}|^{-1/2} \\ &\quad \times \exp\left(-\frac{1}{2}\begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu} \\ \boldsymbol{y}-H\boldsymbol{\mu} \end{bmatrix}^T \begin{bmatrix} \Sigma_{xx} & \Sigma_{xx}H^T \\ H\Sigma_{xx} & H\Sigma_{xx}H^T+\Sigma_{yy} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu} \\ \boldsymbol{y}-H\boldsymbol{\mu} \end{bmatrix}\right). \end{aligned} \tag{A.9}$$

Finally, by using the standard theory of joint and conditional Gaussian random variables, the desired identity (A.1) follows. □

Now, we will proceed with the derivation of the KF. For convenience, we reproduce (2.8)-(2.9):

$$\boldsymbol{x}_k = \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k + \boldsymbol{w}_k \tag{A.10}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{e}_k \tag{A.11}$$

where $\boldsymbol{F}_k$ and $\boldsymbol{H}_k$ are known matrices, $\boldsymbol{u}_k$ is a known input vector, $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k)$, $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{e}_k; \boldsymbol{0}, \boldsymbol{R}_k)$ and $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0)$. To begin, we suppose that $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1})$ is given by $\mathcal{N}(\boldsymbol{x}_{k-1}; \boldsymbol{x}_{k-1|k-1}, \boldsymbol{P}_{k-1|k-1})$. However, from (A.10), the prior is given by

$$p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k, \boldsymbol{Q}_k). \tag{A.12}$$

Therefore, we obtain using (2.3)

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_{k-1} \tag{A.13}$$

$$= \int \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k, \boldsymbol{Q}_k)\mathcal{N}(\boldsymbol{x}_{k-1}; \boldsymbol{x}_{k-1|k-1}, \boldsymbol{P}_{k-1|k-1})d\boldsymbol{x}_{k-1} \tag{A.14}$$

If we use (A.1) with appropriate integrations, it follows that $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ satisfies

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{x}_{k|k-1}, \boldsymbol{P}_{k|k-1}) \tag{A.15}$$

where

$$\boldsymbol{x}_{k|k-1} = \boldsymbol{F}_k \boldsymbol{x}_{k-1|k-1} + \boldsymbol{u}_k \tag{A.16}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_k^T + \boldsymbol{Q}_k. \tag{A.17}$$

Now, we will derive the normalization constant $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$. At time k, $p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$ is given by (2.5). Thus, if we use (A.11) to determine $p(\boldsymbol{y}_k|\boldsymbol{x}_k)$, i.e.,

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}_k \boldsymbol{x}_k, \boldsymbol{R}_k) \tag{A.18}$$

and substitute $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$ given by (A.15) into (2.5), we obtain

$$p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) = \int p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k \tag{A.19}$$

$$= \int \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}_k \boldsymbol{x}_k, \boldsymbol{R}_k) \mathcal{N}(\boldsymbol{x}_k, \boldsymbol{x}_{k|k-1}, \boldsymbol{P}_{k|k-1}) d\boldsymbol{x}_k. \tag{A.20}$$

Again, we make use of (A.1). Therefore, it follows after appropriate integrations that

$$p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{y}_{k|k-1}, \boldsymbol{S}_{k|k-1}) \tag{A.21}$$

where

$$\boldsymbol{y}_{k|k-1} = \boldsymbol{H}_k \boldsymbol{x}_{k|k-1} \tag{A.22}$$

$$\boldsymbol{S}_{k|k-1} = \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^T + \boldsymbol{R}_k. \tag{A.23}$$

What remains is to derive an expression for the posterior PDF at time k. This is given by (2.4), which is recognized as Bayes rule:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) = \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k})}. \tag{A.24}$$

We have determined all the constituent PDF's found in (A.24). Therefore, if we substitute (A.18), (A.15), (A.21) into (A.24) we obtain

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{x}_{k|k}, \boldsymbol{P}_{k|k}) \tag{A.25}$$

where

$$\boldsymbol{x}_{k|k} = \boldsymbol{x}_{k|k-1} + \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T \boldsymbol{S}_{k|k-1}^{-1}(\boldsymbol{y}_k - \boldsymbol{y}_{k|k-1}) \tag{A.26}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{P}_{k|k-1}\boldsymbol{H}_k^T \boldsymbol{S}_{k|k-1}^{-1}\boldsymbol{H}_k \boldsymbol{P}_{k|k-1}. \tag{A.27}$$

This completes the derivation of the KF.

Appendix B

Derivation of ACM filter

For convenience, we reproduce the considered DSSM. From (2.19) and (2.20) we have

$$\boldsymbol{x}_k = \boldsymbol{F}_k \boldsymbol{x}_{k-1} + \boldsymbol{u}_k + \boldsymbol{w}_k \tag{B.1}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{e}_k \tag{B.2}$$

where $\boldsymbol{F}_k$ and $\boldsymbol{H}_k$ are known matrices, $\boldsymbol{u}_k$ is a known input vector, $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{w}_k; \boldsymbol{0}, \boldsymbol{Q}_k)$,

$$\boldsymbol{e}_k \sim \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{e}_k; \mu_k^j, \boldsymbol{R}_k^j) \tag{B.3}$$

and $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = \mathcal{N}(\boldsymbol{x}_0; \hat{\boldsymbol{x}}_0, \hat{\boldsymbol{P}}_0)$.

B.1 Derivation of (2.23)

By definition, $\boldsymbol{x}_{k|k}$ is given by

$$\begin{aligned}
\boldsymbol{x}_{k|k} &= \int \boldsymbol{x}_k p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k}) d\boldsymbol{x}_k \\
&= \int \boldsymbol{x}_k \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})} d\boldsymbol{x}_k \\
&= \hat{\boldsymbol{x}}_{k|k-1} + \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \hat{\boldsymbol{P}}_{k|k-1} \\
&\qquad \times \int p(\boldsymbol{y}_k|\boldsymbol{x}_k) [\hat{\boldsymbol{P}}_{k|k-1}]^{-1} (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1}) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k.
\end{aligned} \tag{B.4}$$

Thus, by adopting the *Masreliez approximation*, i.e., $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \hat{\boldsymbol{P}}_{k|k-1})$, we can note that

$$\nabla_{\boldsymbol{x}_k} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx -[\hat{\boldsymbol{P}}_{k|k-1}]^{-1}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_{k|k-1})p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \tag{B.5}$$

and that (B.4) may be approximated by

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} - \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \hat{\boldsymbol{P}}_{k|k-1} \int p(\boldsymbol{y}_k|\boldsymbol{x}_k) \nabla_{\boldsymbol{x}_k} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k. \tag{B.6}$$

For reasons shown below, it is necessary to apply integration by parts. By doing so, we can write $\int p(\boldsymbol{y}_k|\boldsymbol{x}_k)\nabla_{\boldsymbol{x}_k} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_k$ as

$$\int p(\boldsymbol{y}_k|\boldsymbol{x}_k)\nabla_{\boldsymbol{x}_k} p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_k = -\int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})\nabla_{\boldsymbol{x}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k)d\boldsymbol{x}_k. \tag{B.7}$$

Use of (B.7) into (B.6) results in

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \hat{\boldsymbol{P}}_{k|k-1} \int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \nabla_{\boldsymbol{x}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k. \tag{B.8}$$

Finally, by noting that $\nabla_{\boldsymbol{x}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) = -\boldsymbol{H}_k^T \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k)$, we can rewrite (B.8) as

$$\begin{aligned}
\hat{\boldsymbol{x}}_{k|k} &= \hat{\boldsymbol{x}}_{k|k-1} - \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \hat{\boldsymbol{P}}_{k|k-1} \int \boldsymbol{H}_k^T \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k \\
&= \hat{\boldsymbol{x}}_{k|k-1} - \hat{\boldsymbol{P}}_{k|k-1} \boldsymbol{H}_k^T \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \int \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k \\
&= \hat{\boldsymbol{x}}_{k|k-1} + \hat{\boldsymbol{P}}_{k|k-1} \boldsymbol{H}_k^T g_k(\boldsymbol{y}_k)
\end{aligned} \tag{B.9}$$

where $g_k(\boldsymbol{y}_k) = -\left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$. This completes the derivation of (2.23).

B.2 Derivation of (2.24)

In the following, we denote $\bar{\boldsymbol{x}}_{k|k-1} = \boldsymbol{x}_k - \boldsymbol{x}_{k|k-1}$. By definition, $\mathbb{E}[\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T|\boldsymbol{y}_{1:k}]$ is given by

$$\begin{aligned}
&\mathbb{E}[\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T|\boldsymbol{y}_{1:k}] \\
&= \int \bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k \\
&= \int \boldsymbol{P}_{k|k-1}[\boldsymbol{P}_{k|k-1}]^{-1}\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T[\boldsymbol{P}_{k|k-1}]^{-1}\boldsymbol{P}_{k|k-1}p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k \\
&\qquad - \boldsymbol{P}_{k|k-1}[\boldsymbol{P}_{k|k-1}]^{-1}\left(\int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k\right)\boldsymbol{P}_{k|k-1} + \boldsymbol{P}_{k|k-1} \\
&= \boldsymbol{P}_{k|k-1}\left(\int [\boldsymbol{P}_{k|k-1}]^{-1}\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T[\boldsymbol{P}_{k|k-1}]^{-1}p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k\right. \\
&\qquad \left. - [\boldsymbol{P}_{k|k-1}]^{-1}\int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k\right)\boldsymbol{P}_{k|k-1} + \boldsymbol{P}_{k|k-1} \\
&= \boldsymbol{P}_{k|k-1}\left[\int \left([\boldsymbol{P}_{k|k-1}]^{-1}\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T[\boldsymbol{P}_{k|k-1}]^{-1}\right.\right. \\
&\qquad \left.\left. - [\boldsymbol{P}_{k|k-1}]^{-1}\right)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})d\boldsymbol{x}_k\right]\boldsymbol{P}_{k|k-1} + \boldsymbol{P}_{k|k-1} \\
&= \boldsymbol{P}_{k|k-1}\left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1}\left[\int \left([\boldsymbol{P}_{k|k-1}]^{-1}\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T[\boldsymbol{P}_{k|k-1}]^{-1}\right.\right. \\
&\qquad \left.\left. - [\boldsymbol{P}_{k|k-1}]^{-1}\right)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})p(\boldsymbol{y}_k|\boldsymbol{x}_k)d\boldsymbol{x}_k\right]\boldsymbol{P}_{k|k-1}^l + \boldsymbol{P}_{k|k-1}^l. \qquad \text{(B.10)}
\end{aligned}$$

By adopting the *Masreliez approximation*, i.e., $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{x}_k; \hat{\boldsymbol{x}}_{k|k-1}, \hat{\boldsymbol{P}}_{k|k-1})$, we can note that

$$\nabla_{\boldsymbol{x}_k}\nabla_{\boldsymbol{x}_k}^T p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \approx \left([\hat{\boldsymbol{P}}_{k|k-1}]^{-1}\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T[\hat{\boldsymbol{P}}_{k|k-1}]^{-1} - [\hat{\boldsymbol{P}}_{k|k-1}]^{-1}\right)p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})$$

and that (B.10) may be approximated by

$$\begin{aligned}
\hat{\mathbb{E}}[\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T|\boldsymbol{y}_{1:k}] = \hat{\boldsymbol{P}}_{k|k-1} + \hat{\boldsymbol{P}}_{k|k-1}\left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \\
\times\left[\underbrace{\int \nabla_{\boldsymbol{x}_k^t}\nabla_{\boldsymbol{x}_k}^T p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})p(\boldsymbol{y}_k|\boldsymbol{x}_k)d\boldsymbol{x}_k}_{\boldsymbol{J}}\right]\hat{\boldsymbol{P}}_{k|k-1}. \qquad \text{(B.11)}
\end{aligned}$$

As shown above, we denote the integral by $\boldsymbol{J}$. Consequently, we can write the (i,j) th element of $\boldsymbol{J}$ as

$$[\boldsymbol{J}]_{(i,j)} = \int\int \frac{\partial^2 p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{\partial \boldsymbol{x}_k^{(i)} \partial \boldsymbol{x}_k^{(j)}} p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k^{(i)} d\tilde{\boldsymbol{x}}_k \tag{B.12}$$

where the notations $d\boldsymbol{x}_k^{(i)}$, and $d\tilde{\boldsymbol{x}}_k$ implies integration over the i-th element, and all other remaining components of $\boldsymbol{x}_k$, respectively.

For convenience, we write equation (B.12) in an alternate form. To this end, we obtain after an application of integration by parts

$$[\boldsymbol{J}]_{(i,j)} = \int \left[- \int \frac{\partial p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{\partial \boldsymbol{x}_k^{(j)}} \frac{\partial p(\boldsymbol{y}_k|\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k^{(i)}} d\boldsymbol{x}_k^{(i)} \right] d\tilde{\boldsymbol{x}}_k. \tag{B.13}$$

Now, suppose that $\boldsymbol{H}_k \in \mathbb{R}^{n_y \times n_x}$ in (B.2) satisfies

$$\boldsymbol{H}_k = [\boldsymbol{H}_k^1, \boldsymbol{H}_k^2, \ldots, \boldsymbol{H}_k^{n_x}]$$

where $\boldsymbol{H}_k^m \in \mathbb{R}^{n_y \times 1}$ for $m = 1, \ldots, n_x$. However, recall that

$$\nabla_{\boldsymbol{x}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) = -\boldsymbol{H}_k^T \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k). \tag{B.14}$$

Thus, it can be seen that $\frac{\partial p(\boldsymbol{y}_k|\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k^{(i)}} = -[\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k)$, and that (B.13) may be rewritten as

$$\begin{aligned} [\boldsymbol{J}]_{(i,j)} &= \int \left[\int \frac{\partial p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{\partial \boldsymbol{x}_k^{(j)}} [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k^{(i)} \right] d\tilde{\boldsymbol{x}}_k \\ &= [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} \int \left[\int \frac{\partial p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})}{\partial \boldsymbol{x}_k^{(j)}} p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k^{(j)} \right] d\tilde{\boldsymbol{x}}_k. \end{aligned} \tag{B.15}$$

For reasons shown below, we apply integration by parts a second time. In doing so, (B.15) becomes

$$[\boldsymbol{J}]_{(i,j)} = [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} \int \left[- \int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \frac{\partial p(\boldsymbol{y}_k|\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k^{(j)}} d\boldsymbol{x}_k^{(j)} \right] d\tilde{\boldsymbol{x}}_k. \tag{B.16}$$

If we recall (B.14), it can be seen that $\frac{\partial p(\boldsymbol{y}_k|\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k^{(j)}} = -\nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{x}_k)\boldsymbol{H}_k^j$, and that (B.16) may be written as

$$\begin{aligned}
[\boldsymbol{J}]_{(i,j)} &= [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} \int \left[\int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{x}_k) \boldsymbol{H}_k^j d\boldsymbol{x}_k^{(j)} \right] d\tilde{\boldsymbol{x}}_k \\
&= [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} \nabla_{\boldsymbol{y}_k}^T \left[\int p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) p(\boldsymbol{y}_k|\boldsymbol{x}_k) d\boldsymbol{x}_k \right] \boldsymbol{H}_k^j \\
&= [\boldsymbol{H}_k^i]^T \nabla_{\boldsymbol{y}_k} \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \boldsymbol{H}_k^j. \qquad \text{(B.17)}
\end{aligned}$$

By inspecting (B.17), it is clear that $\boldsymbol{J}$ can be written as

$$\boldsymbol{J} = \boldsymbol{H}_k^T \nabla_{\boldsymbol{y}_k} \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \boldsymbol{H}_k. \qquad \text{(B.18)}$$

However, if we let

$$G_k(\boldsymbol{y}_k) = \nabla_{\boldsymbol{y}_k} g_k(\boldsymbol{y}_k)^T$$

where $g_k(\boldsymbol{y}_k) = -\left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})$ as defined in the previous section. It can be shown that

$$\begin{aligned}
G_k(\boldsymbol{y}_k) &= \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-2} \nabla_{\boldsymbol{y}_k} p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \\
&\quad - \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \nabla_{\boldsymbol{y}_k} \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}) \\
&= g_k(\boldsymbol{y}_k) g_k^T(\boldsymbol{y}_k) - \left[p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1})\right]^{-1} \nabla_{\boldsymbol{y}_k} \nabla_{\boldsymbol{y}_k}^T p(\boldsymbol{y}_k|\boldsymbol{y}_{1:k-1}). \qquad \text{(B.19)}
\end{aligned}$$

Hence, if we substitute (B.19), and (B.18) into (B.11), we have

$$\hat{\mathbb{E}}[\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T|\boldsymbol{y}_{1:k}] = \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T \Big(g_k(\boldsymbol{y}_k) g_k^T(\boldsymbol{y}_k) - G_k(\boldsymbol{y}_k) \Big) \boldsymbol{H}_k \hat{\boldsymbol{P}}_{k|k-1} + \hat{\boldsymbol{P}}_{k|k-1} \qquad \text{(B.20)}$$

Observe that, equation (B.9) results in

$$\hat{\boldsymbol{P}}_{k|k} = \hat{\mathbb{E}}[\bar{\boldsymbol{x}}_{k|k-1}\bar{\boldsymbol{x}}_{k|k-1}^T|\boldsymbol{y}_{1:k}] - \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T g_k(\boldsymbol{y}_k) g_k^T(\boldsymbol{y}_k) \boldsymbol{H}_k \hat{\boldsymbol{P}}_{k|k-1}. \qquad \text{(B.21)}$$

Therefore, by substituting (B.21) into (B.20), we obtain

$$\hat{\boldsymbol{P}}_{k|k} = \hat{\boldsymbol{P}}_{k|k-1} - \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T G_k(\boldsymbol{y}_k) \boldsymbol{H}_k \hat{\boldsymbol{P}}_{k|k-1}.$$

This completes the derivation of (2.24).

B.3 Derivation of (2.25)

By definition, $\boldsymbol{x}_{k|k-1}$ is given by

$$\begin{aligned}\boldsymbol{x}_{k|k-1} &= \int \boldsymbol{x}_k p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1})d\boldsymbol{x}_k \\ &= \int \boldsymbol{x}_k \left(\int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1} \right) d\boldsymbol{x}_k \\ &= \int \left(\int \boldsymbol{x}_k p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) d\boldsymbol{x}_k \right) p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1}. \end{aligned} \tag{B.22}$$

From (B.1), we have $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k, \boldsymbol{Q}_k)$. Upon substitution into (B.22), we arrive at the desired result, that is

$$\begin{aligned}\boldsymbol{x}_{k|k-1} &= \int (\boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k)\, p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1} \\ &= \boldsymbol{F}_k\boldsymbol{x}_{k-1|k-1} + \boldsymbol{u}_k. \end{aligned}$$

B.4 Derivation of (2.26)

By definition $\boldsymbol{P}_{k|k-1}$ is given by

$$\begin{aligned}\boldsymbol{P}_{k|k-1} &= \int (\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})^T p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_k \\ &= \int (\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})^T \left(\int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1} \right) d\boldsymbol{x}_k \\ &= \int \left(\int (\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})(\boldsymbol{x}_k - \boldsymbol{x}_{k|k-1})^T p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) d\boldsymbol{x}_k \right) p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{1:k-1}) d\boldsymbol{x}_{k-1}. \end{aligned}$$

From (B.1), it can be readily seen that $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{F}_k\boldsymbol{x}_{k-1} + \boldsymbol{u}_k, \boldsymbol{Q}_k)$. Upon substitution, and noting that $\boldsymbol{P}_{k-1|k-1} = \mathbb{E}[(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1|k-1})(\boldsymbol{x}_{k-1} - \boldsymbol{x}_{k-1|k-1})^T | \boldsymbol{y}_{1:k-1}]$, we arrive at the desired result, that is

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_k\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_k^T + \boldsymbol{Q}_{k-1}.$$

Appendix C

Derivation of OID

For convenience, we reproduce the considered DSSM:

$$\boldsymbol{a}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{w}_k^1 \tag{C.1}$$

$$z_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + w_k^2 \tag{C.2}$$

$$y_k = z_k + e_k \tag{C.3}$$

where $\boldsymbol{F} = \beta \boldsymbol{I}_{P\times P}$, $\boldsymbol{a}_k = [a_k^1, \ldots, a_k^P]^T$, $\boldsymbol{G}_k(z_{k-P:k-1}) = [z_{k-1}, \ldots, z_{k-P}]$, $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$, $w_k^2 \sim \mathcal{N}(w_k^2; 0, Q_k^2)$, and $e_k \sim \mathcal{N}(e_k; 0, R_k)$. The OID satisfies

$$p(z_k, \boldsymbol{a}_k | z_{k-P:k-1}, \boldsymbol{a}_{k-1}, y_k) = p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, y_k) p(\boldsymbol{a}_k | \boldsymbol{a}_{k-1}, y_k). \tag{C.4}$$

In the following, we will derive analytical expressions for each of the constituent PDF's found in (C.4). The first density $p(z_k | z_{k-P:k-1}, \boldsymbol{y}_k)$ on the right hand side (RHS) of (C.4) satisfies

$$p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k, \boldsymbol{y}_k) = \frac{p(y_k | z_k) p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k)}{p(y_k | z_{k-P:k-1}, \boldsymbol{a}_k)} \tag{C.5}$$

where

$$p(y_k | z_{k-P:k-1}, \boldsymbol{a}_k) = \int p(y_k | z_k) p(z_k | z_{k-P:k-1}, \boldsymbol{a}_k) dz_k. \tag{C.6}$$

Evidently, it is necessary to evaluate $p(y_k|z_k)p(z_k|z_{k-P:k-1}, \boldsymbol{a}_k)$. From (C.2) and (C.3) we have

$$p(y_k|z_k)p(z_k|z_{k-P:k-1}, \boldsymbol{a}_k) = \mathcal{N}(y_k; z_k, R_k)\mathcal{N}(z_k; \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k, Q_k^2). \quad \text{(C.7)}$$

However, by using (A.1), we can rewrite (C.7) as

$$p(y_k|z_k)p(z_k|z_{k-P:k-1}, \boldsymbol{a}_k) = \mathcal{N}(y_k; \hat{y}_k, \hat{S}_k)\mathcal{N}(z_k; \hat{z}_k, \hat{P}_k) \quad \text{(C.8)}$$

where

$$\hat{y}_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k \quad \text{(C.9)}$$

$$\hat{S}_k = Q_k^2 + R_k \quad \text{(C.10)}$$

$$\hat{z}_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + \frac{Q_k^2}{R_k + Q_k^2}(y_k - \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k) \quad \text{(C.11)}$$

$$\hat{P}_k = \left(\frac{1}{R_k} + \frac{1}{Q_k^2}\right)^{-1}. \quad \text{(C.12)}$$

If we substitute into (C.6), it follows that

$$p(y_k|z_{k-P:k-1}, \boldsymbol{a}_k) = \mathcal{N}(y_k; \hat{y}_k, \hat{S}_k). \quad \text{(C.13)}$$

Therefore, we obtain after substituting (C.13) and (C.8) into (C.5)

$$p(z_k|z_{k-P:k-1}, \boldsymbol{a}_k, y_k) = \mathcal{N}(z_k; \hat{z}_k, \hat{P}_k) \quad \text{(C.14)}$$

where

$$\hat{z}_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + \frac{Q_k^2}{R_k + Q_k^2}(y_k - \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k)$$

$$\hat{P}_k = \left(\frac{1}{R_k} + \frac{1}{Q_k^2}\right)^{-1}. \quad \text{(C.15)}$$

This completes the derivation of $p(z_k|z_{k-P:k-1}, \boldsymbol{a}_k, y_k)$.

What remains is to derive an expression for $p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}, y_k)$. By virtue of Bayes' rule, this PDF is given by

$$p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}, y_k) = \frac{p(y_k|z_{k-P:k-1}, \boldsymbol{a}_k)p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1})}{p(y_k|z_{k-P:k-1}, \boldsymbol{a}_{k-1})} \quad \text{(C.16)}$$

where

$$p(y_k|z_{k-P:k-1}, \boldsymbol{a}_{k-1}) = \int p(y_k|z_{k-P:k-1}, \boldsymbol{a}_k)p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1})d\boldsymbol{a}_k. \tag{C.17}$$

In (C.16), the prior $p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1})$ satisfies

$$p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}) = \mathcal{N}(\boldsymbol{a}_k; \boldsymbol{F}\boldsymbol{a}_{k-1}, \boldsymbol{Q}_k^1) \tag{C.18}$$

whereas $p(y_k|z_{k-P:k-1}, \boldsymbol{a}_k)$ is in the form of (C.13). Therefore, we obtain after using (A.1)

$$p(y_k|z_{k-P:k-1}, \boldsymbol{a}_k)p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}) = \mathcal{N}(y_k; \phi_k, \widehat{\Sigma}_k)\mathcal{N}(\boldsymbol{a}_k; \hat{\boldsymbol{a}}_k, \boldsymbol{\Sigma}_k) \tag{C.19}$$

where

$$\phi_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{F}\boldsymbol{a}_{k-1} \tag{C.20}$$

$$\widehat{\Sigma}_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{Q}_k^1\boldsymbol{G}_k(z_{k-P:k-1})^T + Q_k^2 + R_k \tag{C.21}$$

$$\hat{\boldsymbol{a}}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{W}(y_k - \phi_k) \tag{C.22}$$

$$\boldsymbol{\Sigma}_k = \boldsymbol{Q}_k^1 - \boldsymbol{W}\boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{Q}_k^1 \tag{C.23}$$

and $\boldsymbol{W}^{(j)} = \boldsymbol{Q}_k^1\boldsymbol{G}_k(z_{k-P:k-1})^T[\widehat{\boldsymbol{\Sigma}}_k^{(j)}]^{-1}$. Finally, we obtain after substituting (C.19) into (C.16)

$$p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}, y_k) = \mathcal{N}(\boldsymbol{a}_k; \hat{\boldsymbol{a}}_k, \boldsymbol{\Sigma}_k). \tag{C.24}$$

This completes the derivation of the OID.

Appendix D

Derivation of PCRLB

For convenience, we reproduce the considered DSSM:

$$\boldsymbol{a}_k = \boldsymbol{F}\boldsymbol{a}_{k-1} + \boldsymbol{w}_k^1 \tag{D.1}$$

$$z_k = \boldsymbol{G}_k(z_{k-P:k-1})\boldsymbol{a}_k + w_k^2 \tag{D.2}$$

$$y_k = z_k + e_k \tag{D.3}$$

where $\boldsymbol{F} = \beta \boldsymbol{I}_{P\times P}$, $\boldsymbol{a}_k = [a_k^1, \ldots, a_k^P]^T$, $\boldsymbol{G}_k(z_{k-P:k-1}) = [z_{k-1}, \ldots, z_{k-P}]$, $\boldsymbol{w}_k^1 \sim \mathcal{N}(\boldsymbol{w}_k^1; \boldsymbol{0}, \boldsymbol{Q}_k^1)$, $w_k^2 \sim \mathcal{N}(w_k^2; 0, Q_k^2)$, and $e_k \sim \mathcal{N}(e_k; 0, R_k)$. As shown below, it is convenient to reformulate the considered TVAR model (D.1)-(D.3). To this end, we define $\boldsymbol{x}_{k-1}^1 = \boldsymbol{a}_{k-1}$, $\boldsymbol{x}_{k-1}^2 = [z_{k-1}, \ldots, z_{k-P}]^T$, and $\boldsymbol{x}_{k-1} = [\boldsymbol{x}_{k-1}^{1T}, \boldsymbol{x}_{k-1}^{2T}]^T$ so that (D.1)-(D.3) may be equivalently written as:

$$\boldsymbol{x}_k^1 = \boldsymbol{F}\boldsymbol{x}_{k-1}^1 + \boldsymbol{w}_k^1 \tag{D.4}$$

$$\boldsymbol{x}_k^2 = \boldsymbol{F}_k^2(\boldsymbol{x}_k^1)\boldsymbol{x}_{k-1}^2 + \tilde{\boldsymbol{w}}_k^2 \tag{D.5}$$

$$y_k = \boldsymbol{H}_k\boldsymbol{x}_k^2 + e_k \tag{D.6}$$

where

$$\boldsymbol{F}_k^2(\boldsymbol{x}_k^1) = \begin{bmatrix} \boldsymbol{a}_k^T \\ \boldsymbol{B} \end{bmatrix}, \quad \tilde{\boldsymbol{w}}_k^2 = \begin{bmatrix} w_k^2 \\ \boldsymbol{0}_{P-1\times 1} \end{bmatrix}, \quad \boldsymbol{H}_k = \begin{bmatrix} 1 & \boldsymbol{0}_{1\times P-1} \end{bmatrix} \tag{D.7}$$

and

$$\boldsymbol{B} = \left[\boldsymbol{I}_{P-1\times P-1} \quad \boldsymbol{0}_{P-1\times 1}\right]. \tag{D.8}$$

Recall in section 2.3 that the computation of the PCRLB requires the evaluation of $\boldsymbol{D}_{k-1}^{11}$, $\boldsymbol{D}_{k-1}^{12}$, $\boldsymbol{D}_{k-1}^{21}$, and $\boldsymbol{D}_{k-1}^{22}$. Clearly, as seen from (2.88), (2.89), (2.90) and (2.91), the prior density $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ is assumed to be known, and twice differentiable with respect to its arguments $\boldsymbol{x}_k$, and $\boldsymbol{x}_{k-1}$. For the considered DSSM this is a generous assumption. Indeed, the prior $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ given by[1]

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) &= p(\boldsymbol{x}_k^1, \boldsymbol{x}_k^2|\boldsymbol{x}_{k-1}^1, \boldsymbol{x}_{k-1}^2) && \text{(D.9)} \\ &= p(\boldsymbol{x}_k^2|\boldsymbol{a}_k, \boldsymbol{x}_{k-1}^2)p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}) && \text{(D.10)} \end{aligned}$$

where

$$\begin{aligned} p(\boldsymbol{x}_k^2|\boldsymbol{a}_k, \boldsymbol{x}_{k-1}^2) &= \mathcal{N}(z_k; \boldsymbol{a}_k^T\boldsymbol{x}_{k-1}^2, \boldsymbol{Q}_k^2)\prod_{i=1}^{P-1}\delta\left(z_{k-i} - z_{k-i}\right) && \text{(D.11)} \\ p(\boldsymbol{a}_k|\boldsymbol{a}_{k-1}) &= \mathcal{N}(\boldsymbol{a}_k; \boldsymbol{F}\boldsymbol{a}_{k-1}, \boldsymbol{Q}_k^1) && \text{(D.12)} \end{aligned}$$

does not satisfy this assumption. Thus, to obtain the PCRLB we "regularize" (D.5) [55], and as mentioned before, the basic idea is to replace the non-differentiable Dirac delta function with a differentiable narrow Gaussian distribution. Therefore, we define $\tilde{\boldsymbol{x}}_k^2 = [z_k^1, \ldots, z_k^P]^T$, $v_k^i \sim \mathcal{N}(v_k^i; 0, \epsilon)$ for $i = 1, \ldots, P-1$, and approximate (D.5) with

$$\tilde{\boldsymbol{x}}_k^2 = \boldsymbol{F}_k^2(\boldsymbol{x}_k^1)\tilde{\boldsymbol{x}}_{k-1}^2 + \tilde{\boldsymbol{w}}_k^\epsilon \tag{D.13}$$

where

$$\tilde{\boldsymbol{w}}_k^\epsilon = \left[w_k^2, v_k^1, \ldots, v_k^{P-1}\right]^T \tag{D.14}$$

and $\tilde{\boldsymbol{Q}}_k^2 = \mathbb{E}\left[\tilde{\boldsymbol{w}}_k^\epsilon\tilde{\boldsymbol{w}}_k^{\epsilon T}\right]$. Notice for $\epsilon = 0$ the "regularized" process equation (D.13) reduces to the original process equation (D.5). Therefore, by choosing ϵ arbitrarily

[1]Recall that $\boldsymbol{x}_{k-1}^1 = \boldsymbol{a}_{k-1}$, and $\boldsymbol{x}_{k-1}^2 = [z_{k-1}, \ldots, z_{k-P}]^T$.

close to zero ($0 < \epsilon \lll 1$), we can see that (D.13) provides an arbitrarily close approximation of (D.5). Now, let us proceed with the derivation of a PCRLB which for $\epsilon \rightarrow 0$ converges to the PCRLB of the original TVAR model. For convenience, we present the prior $p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2)$, and the likelihood $p(y_k|\tilde{\boldsymbol{x}}_k^2)$ for the regularized TVAR model[2]:

$$p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2) = \mathcal{N}(\tilde{\boldsymbol{x}}_k^2; \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2, \tilde{\boldsymbol{Q}}_k^2)\mathcal{N}(\boldsymbol{a}_k; \boldsymbol{F}\boldsymbol{a}_{k-1}, \boldsymbol{Q}_k^1) \tag{D.15}$$

$$p(y_k|\tilde{\boldsymbol{x}}_k^2) = \mathcal{N}(y_k; \boldsymbol{H}_k\tilde{\boldsymbol{x}}_k^2, R_k) \tag{D.16}$$

In the following, we present the derivations of $\boldsymbol{D}_{k-1}^{11}$, $\boldsymbol{D}_{k-1}^{12}$, $\boldsymbol{D}_{k-1}^{21}$, $\boldsymbol{D}_{k-1}^{22}$, and $\boldsymbol{J}_0$.

Derivation of $\boldsymbol{D}_{k-1}^{11}$:

From (2.88), we have

$$\boldsymbol{D}_{k-1}^{11} = -\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2)\right]. \tag{D.17}$$

where $\tilde{\boldsymbol{x}}_k = [\boldsymbol{x}_k^{1T}, \tilde{\boldsymbol{x}}_k^{2T}]^T$. We obtain using (D.15)

$$\boldsymbol{D}_{k-1}^{11} = \mathbb{E}\left[-\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{a}_k, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{a}_{k-1}, \tilde{\boldsymbol{x}}_{k-1}^2)\right] \tag{D.18}$$

$$= \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} (\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1})^T [\boldsymbol{Q}_k^1]^{-1} (\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1})\right]$$

$$+ \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right] \tag{D.19}$$

$$= \begin{bmatrix} \boldsymbol{F}^T[\boldsymbol{Q}_k^1]^{-1}\boldsymbol{F} & \boldsymbol{0}_{P\times P} \\ \boldsymbol{0}_{P\times P} & \mathbb{E}\left[[\boldsymbol{F}_k^2(\boldsymbol{a}_k)]^T[\tilde{\boldsymbol{Q}}_k^2]^{-1}\boldsymbol{F}_k^2(\boldsymbol{a}_k)\right] \end{bmatrix}. \tag{D.20}$$

Derivation of $\boldsymbol{D}_{k-1}^{12}$:

From (2.89), we have

$$\boldsymbol{D}_{k-1}^{12} = -\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2)\right]. \tag{D.21}$$

[2]Here, the notation is $\boldsymbol{x}_k^1 = \boldsymbol{a}_k$, and $\tilde{\boldsymbol{x}}_k^2 = [z_k^1, \ldots, z_k^P]^T$.

We obtain using (D.15)

$$\begin{aligned}
\boldsymbol{D}_{k-1}^{12} &= \mathbb{E}\left[-\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{a}_k, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{a}_{k-1}, \tilde{\boldsymbol{x}}_{k-1}^2)\right] && \text{(D.22)}\\
&= \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} \left(\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1}\right)^T [\boldsymbol{Q}_k^1]^{-1} \left(\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1}\right)\right]\\
&\quad + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right] && \text{(D.23)}\\
&= \begin{bmatrix} -\boldsymbol{F}^T[\boldsymbol{Q}_k^1]^{-1} & \boldsymbol{0}_{P\times P} \\ \boldsymbol{0}_{P\times P} & \boldsymbol{0}_{P\times P} \end{bmatrix}\\
&\quad + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right] && \text{(D.24)}
\end{aligned}$$

Now, let us simplify the second term of (D.24)

$$\begin{aligned}
&\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_{k-1}}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_{k-1}} \nabla_{\tilde{\boldsymbol{x}}_k}^T \frac{1}{2} \left([\boldsymbol{Q}_k^2]^{-1} \left(z_k^1 - \boldsymbol{a}_k^{1T} \tilde{\boldsymbol{x}}_{k-1}^2\right)^2 + \epsilon^{-1} \sum_{m=2}^{P} \left(z_k^m - z_{k-1}^{m-1}\right)^2 \right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_{k-1}} \begin{bmatrix} \nabla_{\boldsymbol{a}_k}^T & \nabla_{\tilde{\boldsymbol{x}}_k^2}^T \end{bmatrix} \frac{1}{2} \left([\boldsymbol{Q}_k^2]^{-1} \left(z_k^1 - \boldsymbol{a}_k^{1T} \tilde{\boldsymbol{x}}_{k-1}^2\right)^2 + \epsilon^{-1} \sum_{m=2}^{P} \left(z_k^m - z_{k-1}^{m-1}\right)^2 \right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_{k-1}} \begin{bmatrix} -[\boldsymbol{Q}_k^2]^{-1} \left(z_k^1 - \boldsymbol{a}_k^{1T} \tilde{\boldsymbol{x}}_{k-1}^2\right) \tilde{\boldsymbol{x}}_{k-1}^{2T} & \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \end{bmatrix}\right]\\
&= \mathbb{E}\left[\begin{bmatrix} \nabla_{\boldsymbol{a}_{k-1}^1} \\ \nabla_{\tilde{\boldsymbol{x}}_{k-1}^2} \end{bmatrix} \begin{bmatrix} -[\boldsymbol{Q}_k^2]^{-1} \left(z_k^1 - \boldsymbol{a}_k^{1T} \tilde{\boldsymbol{x}}_{k-1}^2\right) \tilde{\boldsymbol{x}}_{k-1}^{2T} & \left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1} \end{bmatrix}\right]\\
&= \begin{bmatrix} \boldsymbol{0}_{P\times P} & \boldsymbol{0}_{P\times P} \\ \mathbb{E}\left[\boldsymbol{a}_k \tilde{\boldsymbol{x}}_{k-1}^{2T} [\boldsymbol{Q}_k^2]^{-1}\right] & \mathbb{E}\left[-[\boldsymbol{F}_k^2(\boldsymbol{a}_k)]^T [\tilde{\boldsymbol{Q}}_k^2]^{-1}\right] \end{bmatrix}. && \text{(D.25)}
\end{aligned}$$

We obtain upon substitution of (D.25) into (D.24)

$$\boldsymbol{D}_{k-1}^{12} = \begin{bmatrix} -\boldsymbol{F}^T[\boldsymbol{Q}_k^1]^{-1} & \boldsymbol{0}_{P\times P} \\ \mathbb{E}\left[\boldsymbol{a}_k \tilde{\boldsymbol{x}}_{k-1}^{2T} [\boldsymbol{Q}_k^2]^{-1}\right] & \mathbb{E}\left[-[\boldsymbol{F}_k^2(\boldsymbol{a}_k)]^T [\tilde{\boldsymbol{Q}}_k^2]^{-1}\right] \end{bmatrix}. \qquad \text{(D.26)}$$

Derivation of $\boldsymbol{D}_{k-1}^{21}$:

From (2.90), we have

$$\boldsymbol{D}_{k-1}^{21} = -\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_{k-1}} \ln p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2)\right] = \left[\boldsymbol{D}_{k-1}^{12}\right]^T. \tag{D.27}$$

The substitution of (D.26) into (D.27) readily yields

$$\boldsymbol{D}_{k-1}^{21} = \begin{bmatrix} -[\boldsymbol{Q}_k^1]^{-1}\boldsymbol{F} & \mathbb{E}\left[[\boldsymbol{Q}_k^2]^{-1}\tilde{\boldsymbol{x}}_{k-1}^2 \boldsymbol{a}_k^T\right] \\ \boldsymbol{0}_{P\times P} & \mathbb{E}\left[-[\tilde{\boldsymbol{Q}}_k^2]^{-1}\boldsymbol{F}_k^2(\boldsymbol{a}_k)\right] \end{bmatrix}. \tag{D.28}$$

Derivation of $\boldsymbol{D}_{k-1}^{22}$:

From (2.91), we have

$$\boldsymbol{D}_{k-1}^{22} = -\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{x}_k^1, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{x}_{k-1}^1, \tilde{\boldsymbol{x}}_{k-1}^2)\right] - \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \ln p(\boldsymbol{y}_k | \tilde{\boldsymbol{x}}_k)\right]. \tag{D.29}$$

We obtain using (D.15), and (D.16)

$$\begin{aligned} \boldsymbol{D}_{k-1}^{22} &= \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} - \ln p(\boldsymbol{a}_k, \tilde{\boldsymbol{x}}_k^2 | \boldsymbol{a}_{k-1}, \tilde{\boldsymbol{x}}_{k-1}^2)\right] + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} - \ln p(\boldsymbol{y}_k | \tilde{\boldsymbol{x}}_k^2)\right] \qquad \text{(D.30)} \\ &= \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2}\left(\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1}\right)^T [\boldsymbol{Q}_k^1]^{-1}\left(\boldsymbol{a}_k - \boldsymbol{F}\boldsymbol{a}_{k-1}\right)\right] \\ &\quad + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2}\left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1}\left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right] \\ &\quad + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2}\left(\boldsymbol{y}_k - \boldsymbol{H}_k\tilde{\boldsymbol{x}}_k^2\right)^T [\boldsymbol{R}_k]^{-1}\left(\boldsymbol{y}_k - \boldsymbol{H}_k\tilde{\boldsymbol{x}}_k^2\right)\right] \qquad \text{(D.31)} \end{aligned}$$

$$\begin{aligned} &= \begin{bmatrix} [\boldsymbol{Q}_k^1]^{-1} & \boldsymbol{0}_{P\times P} \\ \boldsymbol{0}_{P\times P} & \boldsymbol{0}_{P\times P} \end{bmatrix} \\ &\quad + \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k} \frac{1}{2}\left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T [\tilde{\boldsymbol{Q}}_k^2]^{-1}\left(\tilde{\boldsymbol{x}}_k^2 - \boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right] \\ &\quad + \begin{bmatrix} \boldsymbol{0}_{P\times P} & \boldsymbol{0}_{P\times P} \\ \boldsymbol{0}_{P\times P} & \boldsymbol{H}_k^T\boldsymbol{H}_k\boldsymbol{R}_k^{-1} \end{bmatrix}. \end{aligned} \tag{D.32}$$

Now, we will simplify the second term of (D.32)

$$\begin{aligned}
&\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{x}}_k}^{\tilde{\boldsymbol{x}}_k}\frac{1}{2}\left(\tilde{\boldsymbol{x}}_k^2-\boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T[\tilde{\boldsymbol{Q}}_k^2]^{-1}\left(\tilde{\boldsymbol{x}}_k^2-\boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_k}\nabla_{\tilde{\boldsymbol{x}}_k}^T\frac{1}{2}\left([\boldsymbol{Q}_k^2]^{-1}\left(z_k^1-\boldsymbol{a}_k^{1T}\tilde{\boldsymbol{x}}_{k-1}^2\right)^2+\epsilon^{-1}\sum_{m=2}^{P}\left(z_k^m-z_{k-1}^{m-1}\right)^2\right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_k}\begin{bmatrix}\nabla_{\boldsymbol{a}_k}^T & \nabla_{\tilde{\boldsymbol{x}}_k^2}^T\end{bmatrix}\frac{1}{2}\left([\boldsymbol{Q}_k^2]^{-1}\left(z_k^1-\boldsymbol{a}_k^{1T}\tilde{\boldsymbol{x}}_{k-1}^2\right)^2+\epsilon^{-1}\sum_{m=2}^{P}\left(z_k^m-z_{k-1}^{m-1}\right)^2\right)\right]\\
&= \mathbb{E}\left[\nabla_{\tilde{\boldsymbol{x}}_k}\left[-[\boldsymbol{Q}_k^2]^{-1}\left(z_k^1-\boldsymbol{a}_k^{1T}\tilde{\boldsymbol{x}}_{k-1}^2\right)\tilde{\boldsymbol{x}}_{k-1}^{2T}\quad\left(\tilde{\boldsymbol{x}}_k^2-\boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T[\tilde{\boldsymbol{Q}}_k^2]^{-1}\right]\right]\\
&= \mathbb{E}\left[\begin{bmatrix}\nabla_{\boldsymbol{a}_k^1}\\ \nabla_{\tilde{\boldsymbol{x}}_k^2}\end{bmatrix}\left[-[\boldsymbol{Q}_k^2]^{-1}\left(z_k^1-\boldsymbol{a}_k^{1T}\tilde{\boldsymbol{x}}_{k-1}^2\right)\tilde{\boldsymbol{x}}_{k-1}^{2T}\quad\left(\tilde{\boldsymbol{x}}_k^2-\boldsymbol{F}_k^2(\boldsymbol{a}_k)\tilde{\boldsymbol{x}}_{k-1}^2\right)^T[\tilde{\boldsymbol{Q}}_k^2]^{-1}\right]\right]\\
&= \begin{bmatrix}\mathbb{E}\left[\tilde{\boldsymbol{x}}_{k-1}^2\tilde{\boldsymbol{x}}_{k-1}^{2T}[\boldsymbol{Q}_k^2]^{-1}\right] & \left[\mathbb{E}\left[-[\boldsymbol{Q}_k^2]^{-1}\tilde{\boldsymbol{x}}_{k-1}^2\right]\quad\boldsymbol{0}_{P\times P-1}\right]\\ \begin{bmatrix}\mathbb{E}\left[-[\boldsymbol{Q}_k^2]^{-1}\tilde{\boldsymbol{x}}_{k-1}^{2T}\right]\\ \boldsymbol{0}_{P-1\times P}\end{bmatrix} & [\tilde{\boldsymbol{Q}}_k^2]^{-1}\end{bmatrix}.
\end{aligned}\tag{D.33}$$

The substitution of (D.33) into (D.32) results in

$$\boldsymbol{D}_{k-1}^{22} = \begin{bmatrix}[\boldsymbol{Q}_k^1]^{-1}+\mathbb{E}\left[\tilde{\boldsymbol{x}}_{k-1}^2\tilde{\boldsymbol{x}}_{k-1}^{2T}[\boldsymbol{Q}_k^2]^{-1}\right] & B\\ B^T & \boldsymbol{H}_k^T\boldsymbol{H}_k\boldsymbol{R}_k^{-1}+[\tilde{\boldsymbol{Q}}_k^2]^{-1}\end{bmatrix}\tag{D.34}$$

where $B=\left[\mathbb{E}\left[-[\boldsymbol{Q}_k^2]^{-1}\tilde{\boldsymbol{x}}_{k-1}^2\right]\quad\boldsymbol{0}_{P\times P-1}\right]$.

Derivation of $\boldsymbol{J}_0$:

From (2.92), we have

$$\boldsymbol{J}_0=\mathbb{E}\left[\Delta_{\boldsymbol{x}_0}^{\boldsymbol{x}_0}-\ln p(\boldsymbol{x}_0)\right].\tag{D.35}$$

The initial state is assumed to follow

$$p(\boldsymbol{x}_0)=\mathcal{N}(\boldsymbol{x}_0,\boldsymbol{0},\hat{\boldsymbol{Q}}_0).\tag{D.36}$$

This results in

$$\begin{aligned} \boldsymbol{J}_0 &= \mathbb{E}\left[\Delta_{\boldsymbol{x}_0}^{\boldsymbol{x}_0} - \ln p(\boldsymbol{x}_0)\right] && \text{(D.37)} \\ &= \mathbb{E}\left[\Delta_{\boldsymbol{x}_0}^{\boldsymbol{x}_0} \frac{1}{2} (\boldsymbol{x}_0)^T \hat{\boldsymbol{Q}}_0^{-1} (\boldsymbol{x}_0)\right] && \text{(D.38)} \\ &= \hat{\boldsymbol{Q}}_0^{-1}. && \text{(D.39)} \end{aligned}$$

Appendix E

Derivation of (4.26)

For convenience, we present some PDF's that are applicable to the following derivation. The first density is based on the *Masreliez approximation*, that is, we approximate $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2)$ with a single Gaussian distribution:

$$\widehat{p}(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2) = \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k-1}^1, \boldsymbol{P}_{k|k-1}^1). \tag{E.1}$$

The second applicable density is $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{x}_k^1)$, and this can be readily obtained from (4.2):

$$p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{x}_k^1) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^2; \overline{\boldsymbol{w}}_k^{2,(j)} + \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_k^1, \boldsymbol{Q}_k^{2,(j)}). \tag{E.2}$$

To begin our derivation, we write $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$ as

$$\begin{aligned} p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) &= \int p(\boldsymbol{x}_k^2, \boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2) d\boldsymbol{x}_k^1 \\ &= \int p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{x}_k^1) p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2) d\boldsymbol{x}_k^1. \end{aligned} \tag{E.3}$$

Thus, by substituting (E.1), and (E.2) into (E.3), we obtain an approximation of $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$ that is in the form of

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \sum_{j=1}^{N} p_j \int \Big(\mathcal{N}(\boldsymbol{x}_k^2; \overline{\boldsymbol{w}}_k^{2,(j)} + \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_k^1, \boldsymbol{Q}_k^{2,(j)}) \times \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k-1}1, \boldsymbol{P}_{k|k-1}^1) \Big) d\boldsymbol{x}_k^1. \quad \text{(E.4)}$$

Then, by using (A.1), we can write (E.4) as

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \sum_{j=1}^{N} p_j \int \mathcal{N}(\boldsymbol{x}_k^1; \boldsymbol{x}_{k|k}^{1,(j)}, \boldsymbol{P}_{k|k}^{1,(j)}) \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{S}_{k|k-1}^{2,(j)}) d\boldsymbol{x}_k^1 \quad \text{(E.5)}$$

where

$$\begin{aligned}
\boldsymbol{x}_{k|k}^{1,(j)} &= \boldsymbol{x}_{k|k-1}^1 + \boldsymbol{W}_k^{(j)}(\boldsymbol{x}_k^2 - \overline{\boldsymbol{w}}_k^{2,(j)} - \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) - \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k|k-1}^1) \\
\boldsymbol{P}_{k|k}^{1,(j)} &= \boldsymbol{P}_{k|k-1}^1 - \boldsymbol{W}_k^{(j)}\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k|k-1}^l \\
\boldsymbol{x}_{k|k-1}^{2,(j)} &= \overline{\boldsymbol{w}}_k^{2,(j)} + \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k|k-1}^l \\
\boldsymbol{S}_{k|k-1}^{2,(j)} &= \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k|k-1}^1\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T + \boldsymbol{Q}_k^{2,(j)}
\end{aligned}$$

and $\boldsymbol{W}_k^{(j)} = \boldsymbol{P}_{k|k-1}^1\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T(\boldsymbol{S}_{k|k-1}^{2,(j)})^{-1}$. Performing the integration in (E.5) produces

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{S}_{k|k-1}^{2,(j)}).$$

This completes our derivation of (4.26).

Appendix F

Derivation of (4.30)

For convenience, we reproduce some PDF's that are applicable to the following derivation. The first density is the likelihood $p(\boldsymbol{x}_k^1|\boldsymbol{x}_{1:k-1}^2)$ which under the assumption of $h(\boldsymbol{x}_k^2) = \boldsymbol{H}\boldsymbol{x}_k^2$ satisfies

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k^2) = \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_k^2, \boldsymbol{R}_k). \tag{F.1}$$

The second density is the finite dimensional approximation of the prior distribution $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$. This is in the form of (4.26):

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) = \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{S}_{k|k-1}^{2,(j)}) \tag{F.2}$$

where

$$\boldsymbol{x}_{k|k-1}^{2,(j)} = \overline{\boldsymbol{w}}_k^{2,(j)} + \boldsymbol{F}^2(\boldsymbol{x}_{k-n:k-1}^2) + \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{x}_{k|k-1}^1 \tag{F.3}$$

$$\boldsymbol{S}_{k|k-1}^{2,(j)} = \boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)\boldsymbol{P}_{k|k-1}^1\boldsymbol{A}^2(\boldsymbol{x}_{k-n:k-1}^2)^T + \boldsymbol{Q}_k^{2,(j)}. \tag{F.4}$$

To begin, we use Bayes rule to write $p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k)$ as

$$p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k) = \frac{p(\boldsymbol{y}_k|\boldsymbol{x}_k^2)p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)}{\int p(\boldsymbol{y}_k|\boldsymbol{x}_k^2)p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)d\boldsymbol{x}_k^2}. \tag{F.5}$$

Then, by using (F.1) and (F.2), we can approximate $p(\boldsymbol{y}_k|\boldsymbol{x}_k^2)p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2)$ by

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k^2)p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) \approx \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_k^2, \boldsymbol{R}_k)\sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^2; \boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{S}_{k|k-1}^{2,(j)}). \quad \text{(F.6)}$$

Next, we apply (A.1) so that (F.6) can be written as

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k^2)p(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2) \approx \sum_{j=1}^{N} p_j \mathcal{N}(\boldsymbol{x}_k^n; \hat{\boldsymbol{x}}_k^{2,(j)}, \hat{\boldsymbol{P}}_k^{2,(j)})\mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T + \boldsymbol{R}_k) \quad \text{(F.7)}$$

where

$$\begin{aligned} \hat{\boldsymbol{x}}_k^{2,(j)} &= \boldsymbol{x}_{k|k-1}^{2,(j)} + \boldsymbol{W}_k^{(j)}(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(j)}) \\ \hat{\boldsymbol{P}}_k^{2,(j)} &= \boldsymbol{S}_{k|k-1}^{2,(j)} - \boldsymbol{W}_k^{(j)}\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)} \end{aligned}$$

and $\boldsymbol{W}_k^{(j)} = \boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T + \boldsymbol{R}_k)^{-1}$. Finally, by substituting (F.7) into (F.5) we obtain

$$\widehat{p}(\boldsymbol{x}_k^2|\boldsymbol{x}_{1:k-1}^2, \boldsymbol{y}_k) = \sum_{j=1}^{N} \overline{p}_j \mathcal{N}(\boldsymbol{x}_k^2; \hat{\boldsymbol{x}}_k^{2,(j)}, \hat{\boldsymbol{P}}_k^{2,(j)}) \quad \text{(F.8)}$$

where

$$\overline{p}_j = \frac{p_j \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(j)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(j)}\boldsymbol{H}^T + \boldsymbol{R}_k)}{\sum_{n=1}^{N} p_n \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{H}\boldsymbol{x}_{k|k-1}^{2,(n)}, \boldsymbol{H}\boldsymbol{S}_{k|k-1}^{2,(n)}\boldsymbol{H}^T + \boldsymbol{R}_k)}.$$

This completes the derivation of (4.30).

Bibliography

[1] R. G. Albano, L. Franchina, and S. A. Kosmopoulos, "Bit error performance evaluation of double-differential QPSK in faded channels characterized by Gaussian plus impulsive noise and doppler effects," *IEEE Trans. Veh. Technol.*, vol. 49, no. 1, pp. 148–158, Jan. 2000.

[2] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximation," *IEEE Trans. Automat. Contr.*, vol. 17, no. 4, pp. 439–448, Aug. 1972.

[3] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.

[4] C. Andrieu, M. Davy, and A. Doucet, "Efficient particle filtering for jump Markov systems. Applications to time-varying autoregressions," *IEEE Trans. Signal Processing*, vol. 51, no. 7, pp. 1762–1770, July 2003.

[5] C. Andrieu and A. Doucet, "Particle filtering for partially observed Gaussian state space models," *J. Royal Statist. Soc. B*, vol. 64, pp. 827–836, 2002.

[6] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[7] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction.* New York: Wiley, 2001.

[8] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Trans. Broadcast.*, vol. 48, no. 3, pp. 223–229, 2002.

[9] E. Costa and S. Pupolin, "M-QAM-OFDM system performance in the presence of a nonlinear amplifier and phase noise," *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 462–472, Mar. 2002.

[10] M. Daly, "Marginalized particle filtering for blind system identification," Master's thesis, Dept. of Elec. and Comp. Eng., McMaster University, Hamilton, ON, Canada, 2004.

[11] P. M. Djurić, J. Kotecha, F. Esteve, and E. Perret, "Sequential parameter estimation of time-varying non-Gaussian autoregressive processes," *EURASIP Journal on Applied Signal Processing*, vol. 2002:8, pp. 865–875.

[12] A. Doucet, "On sequential Monte Carlo sampling methods for Bayesian filtering," University of Cambridge, Department of Engineering, Signal Processing Group, England, Tech. Rep. TR. 310, 1998.

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

[14] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, Dec. 2000.

[15] A. Doucet, N. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49, no. 3, pp. 613–624, Mar. 2001.

[16] K. B. Eom, "Time-varying autoregressive modeling of HRR radar signatures," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 3, pp. 974–988, 1999.

[17] A. Farina, B. Ristic, and D. Benvenuti, "Tracking a ballistic target: Comparison of several nonlinear filters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 3, pp. 854–867, 2002.

[18] W. Fong, S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing with application to audio signal enhancement," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 438–449, Feb. 2002.

[19] T. Ghirmai, P. M. Djurić, M. F. Bugallo, and J. Kotecha, "Multisample receivers for time-varying channels using particle filtering," in *Proc. IEEE Workshop Signal Processing Advances in Wireless Communications*, Rome, Italy, June 2003.

[20] N. Gordon, D. Salmond, and A. Smith, "Novel approach to non-linear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[21] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 425–437, Feb. 2002.

[22] M. G. Hall, A. V. Oppenheim, and A. D. Wilsky, "Time varying parametric modeling of speech," *Signal Processing*, vol. 5, pp. 276–285, 1983.

[23] S. Haykin, K. Huber, and Z. Chen, "Bayesian sequential state estimation for MIMO wireless communications," *Proc. IEEE*, vol. 92, no. 3, pp. 439–454, Mar. 2004.

[24] W. C. Jakes, *Microwave Mobile Communications.* New York: Wiley, 1974.

[25] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[26] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, Series D, pp. 35–45, 1960.

[27] S. M. Kay, *Modern Spectral Estimation.* Englewood Cliffs, NJ, Prentice-Hall, USA, 1988.

[28] G. Kitagawa, "Non-Gaussian state-space modeling of nonstationary time series," *J. Am. Stat. Assoc.*, vol. 82, no. 400, pp. 1032–1063, 1987.

[29] ——, "Monte Carlo filter and smoother for non-Gaussian non-linear state space models," *J. Comp. Graph. Stat.*, vol. 5, no. 1, pp. 1–25, 1996.

[30] C. Komninakis, C. Fragouli, A. H. Sayed, and R. D. Wesel, "Multi-input Multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Trans. Signal Processing*, vol. 50, no. 5, pp. 1065–1076, May 2002.

[31] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputation method and Bayesian missing data problems," *J. Am. Stat. Assoc.*, vol. 89, no. 425, pp. 278–288, 1994.

[32] J. H. Kotecha and P. M. Djurić, "Gaussian particle filtering," *IEEE Trans. Signal Processing*, vol. 51, no. 10, pp. 2592–2601, Oct. 2003.

[33] J. S. Liu and R. Chen, "Blind deconvolution via sequential imputation," *J. Am. Stat. Assoc.*, vol. 90, pp. 567–576, 1995.

[34] ——, "Sequential Monte Carlo methods for dynamic systems," *J. Am. Stat. Assoc.*, vol. 93, pp. 1032–1044, 1998.

[35] ——, "Mixture Kalman filters," *J. Roy. Statistic. Soc. B*, vol. 62, pp. 493–508, 2000.

[36] Z. Liu, X. Ma, and G. B. Giannakis, "Space-time coding and Kalman filtering for time-selective fading channels," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 183–186, Feb. 2002.

[37] C. J. Masreliez, "Approximate non-Gaussian filtering with linear state and observation relations," *IEEE Trans. Automat. Contr.*, vol. 20, no. 1, pp. 107–110, Jan. 1975.

[38] J. Miguez and P. M. Djurić, "Blind equalization of frequency-selective channels by sequential importance sampling," *IEEE Trans. Signal Processing*, vol. 52, no. 10, pp. 2738–2748, Oct. 2004.

[39] R. Negi and J. Cioffi, "Pilot tone selection for channel estimation in a mobile OFDM system," *IEEE Trans. Consumer Electron.*, vol. 44, no. 3, pp. 1122–1128, Aug. 1998.

[40] A. Papoulis and P. Unnikrishna, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York: McGraw-Hill, 2001.

[41] D. Petrovic, W. Rave, and G. Fettweis, "Phase Noise suppression in OFDM using a Kalman filter," in *Proc. IEEE Wireless Personal Multimedia Communications*, Yokosuka, Japan, Oct. 2003.

[42] L. Piazzo and P. Mandarini, "Analysis of phase noise effects in OFDM modems," *IEEE Trans. Commun.*, vol. 50, no. 10, pp. 1696–1705, Oct. 2002.

[43] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filter," *J. Am. Stat. Assoc.*, vol. 94, pp. 590–599, 1999.

[44] T. Pollet, M. V. Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise," *IEEE Trans. Commun.*, vol. 43, no. 234, pp. 191–193, Feb./March/April 1995.

[45] H. V. Poor, "Active interference suppression in CDMA overlay systems," *IEEE J. Select. Areas Commun.*, vol. 19, no. 1, pp. 4–20, Jan. 2001.

[46] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.

[47] T. S. Rappaport, *Wireless Communications: Principles and Practise.* New Jersey: Prentice-Hall PTR, 1996.

[48] B. Ristic, A. Farina, D. Benvenuti, and M. S. Arulampalam, "Performance bounds and comparison of nonlinear filters for tracking a ballistic object on re-entry," *IEE Proceedings -, Radar, Sonar and Navigation*, vol. 150, no. 2, pp. 65–70, Apr. 2003.

[49] J. K. O. Ruanaidh and W. J. Fitzgerald, *Numerical Bayesian Methods Applied to Signal Processing.* New York: Springer-Verlag, 1996.

[50] T. Schon, F. Gustafsson, and P. J. Nordlund, "Marginalized particle filters for nonlinear state-space models," Department of Electrical Engineering, Linkoping University, SE-581 83 Linkoping, Sweden, Tech. Rep. LiTH-ISY-R-2548, Oct. 1998.

[51] D. Sengupta and S. Kay, "Efficient estimation of parameters for non-Gaussian autoregressive processes," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 6, pp. 785–794, June 1989.

[52] N. Shephard, "Partial non-Gaussian times series models," *Biometrika*, vol. 81, pp. 115–131, 1994.

[53] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, July 1971.

[54] H. Tanizaki, *Nonlinear Filters: Estimation and Applications*, 2nd ed. New York: Springer-Verlag, 1996.

[55] P. Tichavský, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-rao bounds for discrete-time nonlinear filtering," *IEEE Trans. Signal Processing*, vol. 46, no. 5, pp. 1386–1396, May 1998.

[56] L. Tomba, "On the effect of Wiener phase noise in OFDM systems," *IEEE Trans. Commun.*, vol. 46, no. 5, pp. 580–583, May 1998.

[57] L. Tomba and W. A. Krzymien, "Sensitivity of the MC-CDMA access scheme to carrier phase noise and frequency offset," *IEEE Trans. Veh. Technol.*, vol. 48, no. 5, pp. 1657–1665, 1999.

[58] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," Cambridge University Engineering Department, Tech. Rep. CUED/FINFENG/ TR-380, Tech. Rep., 2000.

[59] J. Vermaak, C. Andrieu, A. Doucet, and S. J. Godsill, "Particle methods for Bayesian modeling and enhancement of speech signals," *IEEE Trans. Speech Audio Processing*, vol. 10, no. 3, pp. 173–185, Mar. 2002.

[60] M. Šimandl, J. Královec, and P. Tichavský, "Filtering, predictive, and smoothing Cramér-rao bounds for discrete-time nonlinear dynamic systems," *Automatica*, vol. 37, no. 11, pp. 1703–1716, Nov. 2001.

[61] X. Wang and R. Chen, "Adaptive Bayesian multiuser detection for synchronous CDMA with Gaussian and impulsive noise," *IEEE Trans. Signal Processing*, vol. 47, no. 7, pp. 2013–2028, 2000.

[62] X. Wang and H. V. Poor, "Joint channel estimation and symbol detection in Rayleigh flat-fading channels with impulsive noise," *IEEE Commun. Lett.*, vol. 1, no. 1, pp. 19–21, Jan. 1997.

[63] P. H. Y. Wu and A. Duel-Hallen, "Multiuser detectors with disjoint Kalman channel estimators for synchronous CDMA mobile radio channels," *IEEE Trans. Commun.*, vol. 48, no. 5, pp. 752–756, May 2000.

[64] S. Wu and Y. Bar-Ness, "A phase noise suppression algorithm for OFDM-based WLANs," *IEEE Commun. Lett.*, vol. 6, no. 12, pp. 535–537, Dec. 2002.

[65] ——, "OFDM systems in the presence of phase noise: consequences and solutions," *IEEE Trans. Commun.*, vol. 52, no. 11, pp. 1988–1996, Nov. 2004.

[66] W. R. Wu, "Maximum likelihood identification of glint noise," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 1, pp. 41–51, Jan. 1996.