TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Henri Pesonen

Bayesian Estimation and Quality Monitoring for Personal Positioning Systems

Julkaisu 1112 • Publication 1112

Tampere 2013

Tampereen teknillinen yliopisto. Julkaisu 1112
Tampere University of Technology. Publication 1112

Henri Pesonen

Bayesian Estimation and Quality Monitoring for Personal Positioning Systems

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Sähkötalo Building, Auditorium S4, at Tampere University of Technology, on the 8th of February 2013, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2013

ISBN 978-952-15-3008-1 (printed)
ISBN 978-952-15-3016-6 (PDF)
ISSN 1459-2045

Abstract

Personal positioning is a dynamic estimation problem where the ability to assess the quality of the positioning service is as important as obtaining accurate position estimates. When estimating the position of a person, as opposed to e.g. an airplane, the type of motion can change at any time as a pedestrian can board a bus, or a cyclist can board a train. Also the changing surroundings in urban navigation influence the observation noise as tall buildings blocking the line of sight to satellites are full of reflecting surfaces.

First we investigate classic robust estimation methods applied to the positioning problem, but then we focus on the Bayesian framework, as its generality allows us to take into account the abrupt changes in the state-space system. Gaussian mixture distributions and Markov chain indicator processes are used to model the changing systems. We evaluate the resulting systems mainly with sequential Monte Carlo methods, as this approach gives us an approximative joint posterior distribution of the errors and the state. We propose a general framework for the Bayesian receiver autonomous integrity monitoring in urban navigation based on the posterior probabilities.

We also use the Bayesian framework to solve the explicit effect of the sensor errors in a nominal system that estimates the state with the assumption of no changes in the models. We use the estimated cumulated effect of the errors in the time series to determine whether error is present in the system at any time. Finally, a variational Bayes algorithm is developed for detecting changes in the system noise covariances.

Preface

The research presented in this thesis was carried out at the Department of Mathematics at Tampere University of Technology during 2006-2011. During the writing of this thesis I have been financially supported by Tekes, Tampere University of Technology Graduate School, Tampere Graduate School in Information Science and Engineering, Emil Aaltonen foundation and Nokia foundation.

I would like to thank my supervisor Professor Robert Piché for his guidance and making it possible for me to do this research. I'm grateful for the support of my co-workers and colleagues at the department for the inspiring work environment. I also thank Simo Ali-löytty, Matti Raitoharju and rest of the researchers of the personal positioning algorithms research group for helpful discussions and activities outside of work. I'm especially grateful for the professional support and friendship of Lassi Paunonen. I would also want to express my gratitude for the pre-examiners Professor Joaquín Míquez and Professor Andreas Wieser for their comments and suggestions.

Lastly, I would like to thank my mother Tuire and my late father Ahti, my family and friends, for the love and support they have given me over the years. A special mention goes for my outstanding friends at Tynnyri. Most of all, I'm very grateful for the encouraging, supportive and loving Maiju for everything.

Tampere, January 2013,

Henri Pesonen

Contents

List of publications

This thesis consists of an introduction and the following publications, in chronological order:

P1. "Robust estimation techniques for GNSS positioning." In Proceedings of NAV07-The Navigation Conference and Exhibition, London, England, October 24–November 1, 2007.

P2. "Bayesian receiver autonomous integrity monitoring technique" (with Robert Piché). In *Proceedings of the Institute of Navigation International Technical Meeting 2009*, January 26–28, Anaheim, CA, 2009.

P3. "Outlier-robust Bayesian filter with integrity monitoring for GNSS positioning." In *Proceedings of European Navigation Conference - ENC-GNSS 2009*, May 3 – 6, Naples, Italy, 2009.

P4. "Bayesian positioning using Gaussian mixture models with time-varying component weights" (with Robert Piché). In *Proceedings of the 2011 Joint Statistical Meetings*, July 30– August 4, Miami, FL 2011.

P5. "A framework for Bayesian receiver autonomous integrity monitoring in urban navigation." *NAVIGATION*, 58(3):229–240, 2011.

P6. "Bayesian fault diagnosis method for linear systems with additive sensor errors" (with Robert Piché). In *2nd International Conference and Exhibition on Ubiquitous Positioning, Indoor Navigation, and Location Based Service UPINLBS 2012*, October 3–4, Helsinki 2012.

P7. "Variational Bayes algorithm for tracking linear systems with abrupt changes of the noise covariances" (with Robert Piché). Research report 100, Tampere University of Technology, Department of Mathematics, 2012.

CHAPTER 1

Introduction

This thesis consists of an introduction and six articles published in scientific conferences and journals, and one technical report. The purpose of this introductory chapter is not to repeat the derivations or results given in the publications [P1]–[P7], but rather to give a short unified background, and summarise the contribution in context.

The main contributions are:

P1: The performance of robust static and dynamic estimation methods in positioning problem with range and pseudorange measurements is investigated using simulated and real data. The publication demonstrates the need for robust estimators as even with a small amount of bad observations, robust methods perform better than the traditional methods.

P2: A Bayesian model comparison approach for detection and identification of additive biases in range measurement is presented. The introduced Bayesian method has a more straightforward interpretation of the results than the traditional RAIM/FDE.

P3: A Bayesian quality monitoring approach for dynamic systems based on Gaussian mixture filter is introduced. The proposed method can handle multiple outliers and its integrity monitoring is based on the estimation error instead of the presence of observation biases. Also, a fast novel merging method of the Gaussian mixture components is presented.

P4: A hierarchical approach for modeling varying environments in positioning systems is discussed, and an algorithm for solving the resulting problem is provided. The simulations show that the technique can be used to approximate the uncertainty parameter, in addition to providing performance comparable to optimal methods.

P5: A Bayesian framework for receiver autonomous integrity monitoring in urban navigation based on Bayesian filtering of the joint distribution of state and observation biases is introduced. The method is applicable to more general problems than the traditional integrity monitoring techniques. Also, because the integrity is determined by monitoring the probabilities of the large state errors, the integrity results are easily interpretable.

P6: A novel Bayesian fault diagnosis method for linear systems with observations contaminated with additive errors is developed. The method performs better than the standard detection-identification-adaptation method, with the expense of more computation power.

P7: A variational Bayes method for approximative batch estimation of linear of state-space systems with changes in the state transition or observation noise covariance is discussed and a heuristic online version of the method is provided. Both of the methods perform well against the standard methods.

The author's role in the shared publications:

- Publication P2: Based on the initial idea of the co-author, the author developed the method, wrote the computer code and the manuscript.
- Publications P4 and P6: The author came up with the main ideas, wrote the computer code and the manuscript.

- Publication P7: Based on the initial idea of the co-author, the author developed the methods, wrote the computer code and the manuscript.

1 Background

Personal positioning is a research area with a huge amount of different applications such as locating emergency callers, various map-services and location-aware gaming. Therefore different positioning methods have been researched intensively and will be researched in the near future as, with technological advancements, more people are carrying devices with the capability to solve its position coordinates using measurements from various sources. Currently the most popular positioning methods are based on the Global Positioning System (GPS) [39], [52], but there are many other sources providing signals that can be used for positioning. In addition to other satellite based positioning systems such as Glonass, Compass and Galileo, other sources providing data that could be used for positioning include cellular networks [48], [29], [42] and WLAN networks [41], [47]. Among the devices with the positioning capability, the mobile phone is possibly the most pervasive, and nowadays many phone models are integrated with various positioning services.

There is a vast number of scientific publications about the methods for positioning and several international conferences dedicated to the topic. Mathematically the positioning problem is formulated as a nonlinear filtering problem in which a positioning estimate is computed using noisy observations and a model for the receiver motion [63], [2]. An important part of the problem is to choose a realistic a model for the underlying system that is as simple as possible so that the problem would be feasible to solve in an environment where the computational resources are limited.

There is a large number of proposed techniques for the positioning problem which solve the problem very well under certain circumstances. Bayesian statistics provide a consistent and a theoretically optimal framework for solving the filtering problem recursively [33],[18], although in practice we are often required to use approximative methods [8], [19], [6], [P5].

Many of the measurements used in positioning systems are based on radio signals. These signals are sometimes affected by the surround-

ing environments as the signals can be reflected and attenuated e.g. when the receiver is in a building, or in an urban canyon. A well-known effect of the signal distortion is the multipath effect that is a main source of error in satellite positioning [39], [65], [23], [46], [66], [49]. It is difficult for the receiver to detect whether the signals are affected or not [68], but if these effects are not taken into account in the system models, the accuracy of the position estimates can be severely degraded.

Therefore it is of the utmost importance to develop methods that are able to provide as good as possible position service in situations where the signal quality may be degraded. A first possible approach for this is to develop estimation methods that perform well when observations contain outliers, i.e. severely degraded observations [51], [56], [43], [P1]. A second approach is to construct models that describe how the signals are degraded, e.g. they contain an additive bias [24],[32], [P3], [P5], or the observation noise can be described more accurately with the t-distribution than with a Gaussian [69], [1]. A third approach is to perform statistical tests to check whether there is evidence that supports a hypothesis that the underlying nominal assumptions may be wrong [7], [9], [26], [P6].

In addition, it is often not sufficient to solve only for the position coordinates, but it is important to provide an estimate of the quality of the service. In the case of poor quality of service we may be able to switch to a backup position system based on e.g. accelerometers or gyroscopes. In GPS, the methods used to monitor the quality, or the integrity, of the positioning service are traditionally referred to as receiver autonomous integrity monitoring (RAIM) [14], [30],[54]. The main application of RAIM has been in safety-critical aviation navigation [39]. However, the requirements for aviation are very different from those for personal urban navigation. In this thesis we investigate the Bayesian approach to provide quality monitoring service in addition to the position estimates for the personal positioning problem where the observations may, or may not, be contaminated with additive errors.

2 Problem statement

We consider the problem of estimating the state (position, velocity, acceleration, etc.) of a mobile set (MS) using noisy observations, possibly contaminated with additive failure components. The problem is described by a discrete-time state-space model

$$x_{k+1} = F_k x_k + w_k \tag{1}$$
$$y_k = h_k(x_k) + s_k + v_k \tag{2}$$
$$x_0 \sim \mathsf{N}\left(x_{0|0}, P_{0|0}\right), \tag{3}$$

where $x \sim \mathsf{N}(\mathsf{E}(x), \mathsf{V}(x))$ means that x is a Gaussian vector-valued random variable with mean $\mathsf{E}(x)$ and variance-covariance matrix $\mathsf{V}(x)$. We use notation $x_{k|k} := \mathsf{E}(x_k \mid y_{1:k})$ where $y_{1:k} := [y_1, \ldots, y_k]$ for the conditional mean. Analogous notation is used for the covariance $\mathsf{V}(x_k \mid y_{1:k})$. Throughout this work we assume that the state $x_k \in \mathbb{R}^{n_x}$ evolves according to the linear state transition model (1) with the state transition matrix $F_k \in \mathbb{R}^{n_x \times n_x}$ and the additive process noise $w_k \in \mathbb{R}^{n_x}$ that is assumed to be Gaussian

$$w_k \sim \mathsf{N}(0, Q_k). \tag{4}$$

The observations $y_k \in \mathbb{R}^{n_y}$ are described with the function $h_k(\cdot)$ of the state, and the additive Gaussian noise component

$$v_k \sim \mathsf{N}(0, R_k). \tag{5}$$

In addition, we assume that the observation may be contaminated with the additive sensor error component s_k. The stochastic processes w_k and v_k are assumed to be mutually independent white noise processes and independent of the initial state x_0 with the prior distribution (3).

2.1 Failure models

To to complete the specification of the statistical model (1)–(3) we need to model the additive failure components s_k. Let

$$h_k(\cdot) = \begin{bmatrix} h_{k,1}(\cdot) \\ \vdots \\ h_{k,m}(\cdot) \end{bmatrix}, \tag{6}$$

where $h_{k,1}(\cdot), \ldots, h_{k,m}(\cdot)$ is a partition of the observations such that the additive errors have a compatible partition

$$s_k = \begin{bmatrix} s_{k,1} \\ \vdots \\ s_{k,m} \end{bmatrix}, \tag{7}$$

where $s_{k,1}, \ldots, s_{k,m}$ are mutually independent. This kind of model would be natural for a situation in which we have pseudorange observations from several satellites, and depending whether the individual signals between the satellites and the receiver is unobstructed or not, there may be an additional error component present. Of course, we would have to assume that the visibility to one satellite is independent of the visibility to any other satellite. To model the presence of the sensor errors in the observations at time k, we use indicator variables $\lambda_{k,l} \in \{0, 1\}$ so that

$$s_{k,l} = \lambda_{k,l} r_{k,l}, \tag{8}$$

where $r_{k,l}$ is the magnitude of the error. We use two models for the indicator variables throughout the thesis. The first model for the indicator variable is the Bernoulli-distribution

$$\mathsf{P}(\lambda_{k,l} = 1) = \theta_k = 1 - \mathsf{P}(\lambda_{k,l} = 0) \tag{9}$$

where θ_k is the probability of an additive sensor error being present in $y_{k,l}$. The notation $\mathsf{P}(\cdot)$ is used for the probability mass functions of discrete random variables.

In the case of dynamic systems it is sometimes reasonable to model the indicator variable as a Markov chain [68] with transition probabilities

$$\mathsf{P}(\lambda_{k,l} = j \mid \lambda_{k-1,l} = i) = \theta_{ji}, \tag{10}$$

$$\mathsf{P}(\lambda_{0,l} = 1) = \theta_0. \tag{11}$$

Note that in both models, we assume for simplicity that the probabilities of the indicator variable are the same for each of the elements of the observation vector.

We use a linear state transition model for the sensor error size

$$r_{k+1,l} = \phi_{k,l}\, r_{k,l} + \epsilon_{k+1,l} \tag{12}$$

$$r_{0,l} \sim \mathsf{N}\left(r_{0|0,l}, P^r_{0|0,l}\right), \tag{13}$$

where $\epsilon_{k+1,l} \sim \mathsf{N}\left(0, \Sigma_{k+1,l}\right)$ is a Gaussian white noise process. The choice $\phi_{k,l} = 0$ results in the Gaussian white noise process for the error size that could be used to model outliers in the observation noise [55], [13]. The state transition model with the coefficient $\phi_{k,l} = 1$ is the Gaussian random walk that can be used to model the evolution of the multipath bias in GPS pseudorange measurements [23], [24], [P5].

The system (1)–(3) can be expressed using the sensor error models as

$$\begin{bmatrix} x_{k+1} \\ r_{k+1} \end{bmatrix} = \begin{bmatrix} F_k & 0 \\ 0 & \Phi_k \end{bmatrix} \begin{bmatrix} x_k \\ r_k \end{bmatrix} + \begin{bmatrix} w_k \\ \epsilon_{k+1} \end{bmatrix} \tag{14}$$

$$y_k = h_k(x_k) + \Lambda_k r_k + v_k \tag{15}$$

$$\begin{bmatrix} x_0 \\ r_0 \end{bmatrix} \sim \mathsf{N}\left(\begin{bmatrix} x_{0|0} \\ r_{0|0} \end{bmatrix}, \begin{bmatrix} P_{0|0} & 0 \\ 0 & P^r_{0|0} \end{bmatrix} \right) \tag{16}$$

where we have defined

$$\Phi_k := \begin{bmatrix} \phi_{k,1} & & \\ & \ddots & \\ & & \phi_{k,m} \end{bmatrix},$$

$$\Lambda_k := \begin{bmatrix} \lambda_{k,1} I_{n_{y_1}} & & \\ & \ddots & \\ & & \lambda_{k,m} I_{n_{y_m}} \end{bmatrix},$$

$$P^r_{0|0} = \begin{bmatrix} P^r_{0|0,1} & & \\ & \ddots & \\ & & P^r_{0|0,m} \end{bmatrix},$$

$$r_{0|0} := \begin{bmatrix} r^T_{0|0,1} & \cdots & r^T_{0|0,m} \end{bmatrix}^T,$$

$$\epsilon_k := \begin{bmatrix} \epsilon^T_{k,1} & \cdots & \epsilon^T_{k,m} \end{bmatrix}^T.$$

The augmented system (14)–(16) is a standard state-space system if the indicator variables would be known. Naturally one could consider the indicator variable as a part of the state and formulate a larger non-linear and non-Gaussian system, but we handle Λ_k separately due to the approximation methods discussed later on.

Furthermore, in the case $\phi_{i,j} = 0,\ \forall i, j$ we can write the system as

$$x_{k+1} = F_k x_k + w_k \tag{17}$$

$$y_k = h_k(x_k) + v_k(\Lambda_k) \tag{18}$$

$$x_0 \sim \mathsf{N}\left(x_{0|0}, P_{0|0}\right), \tag{19}$$

where the observation noise given Λ_k is a Gaussian white noise process

$$v_k(\Lambda_k) := \Lambda_k r_k + v_k \sim \mathsf{N}(0, R_k(\Lambda_k)) \tag{20}$$

$$R_k(\Lambda_k) := R_k + \Lambda_k \Sigma_k \Lambda_k^T \tag{21}$$

The system (17)–(19) is convenient if we are considering the additive sensor errors simply as nuisance parameters causing the deterioration of the observation quality.

Heavy-tailed distributions

Another approach for modeling faulty data would be to use non-Gaussian distributions for the error components. In the robust filter design similar to M-estimation [31, 27], very large outliers are modeled with the noise process

$$s_k + v_k \sim \mathscr{F} \tag{22}$$

where $\mathscr{F}$ is a member of a family of distributions with fat tails. The robust filter design is then based on finding a state estimator that performs the best should the noise have any of the distributions in the family $\mathscr{F}$ [50],[51], [56], [P1].

Outliers in the observation noise process can also be modeled simply using a heavy-tailed distribution such as Student-t distribution for the observation noise instead of the Gaussian noise [69, 1]. A sample drawn from a heavy-tailed distribution would result in more realizations that are far away from the bulk of the data.

Hierarchical modeling of varying environment

A Bayesian approach to the state-space estimation problem enables us to use hierarchical modeling to describe more complex real world phenomena. One application of hierarchical modeling is to describe the probability of the presence of the additive sensor error (9) as a time-evolving parameter. Given the model we can solve it jointly with the state [P5]. The approach is reasonable because the probability of the faulty observation is often dependent on the surrounding environments that change gradually when the MS is moving with a reasonably low velocity.

Now the probability mass function of $\lambda_{k,l}$ is defined with a hierarchical model depending on the time-varying unknown variable θ_k as follows

$$\mathsf{P}(\lambda_{k,l} = 0 \mid \theta_k) = 1 - \mathsf{P}(\lambda_{k,l} = 1 \mid \theta_k) = 1 - \theta_k. \tag{23}$$

In the state-transition model for the uncertainty parameter θ_k we have to take into account that $\theta_k \in [0,1]$. The parameter is modeled as

a Markov process, and we take the density $p(\theta_{k+1}|\theta_k)$ to be unimodal, with the mode near to the value of θ_k. A probability density fulfilling these criteria would be

$$\mathsf{beta}(\xi \mid \alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\xi^{\alpha-1}(1-\xi)^{\beta-1}, \tag{24}$$

that is a beta density with parameters α and β evaluated at ξ. The mode and variance of a beta distributed random variable are

$$\mathsf{mode}(\xi) = \frac{\alpha-1}{\alpha+\beta-2}, \quad \mathsf{V}(\xi) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta-1)}. \tag{25}$$

The beta density function is unimodal when $\alpha,\beta > 1$, and the variance of a beta distributed random variable depends on α and β, with variance $\to 0$ as $\alpha,\beta \to \infty$.

The model probability is modeled as having the state-transition density

$$p(\theta_{k+1} \mid \theta_k, S) = \mathsf{beta}(\theta_{k+1} \mid \theta_k(S-2)+1, (1-\theta_k)S+2\theta_k-1), \tag{26}$$

where S is a tuning parameter. The state transition density is illustrated in Figure 1. In Figure 2 we have drawn a few example sample paths of the process with a corresponding sample path of $\lambda_k \mid \theta_k$.

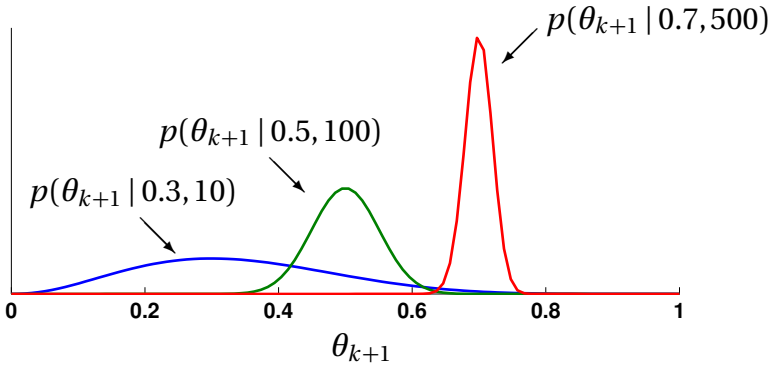

Figure 1: State-transition densities with different parameter values.

The mode and variance of (26) are

$$\mathsf{mode}(\theta_{k+1} \mid \theta_k, S) = \theta_k, \quad \mathsf{V}(\theta_{k+1} \mid \theta_k, S) = \frac{(1-\theta_k)\theta_k}{S-1}. \tag{27}$$

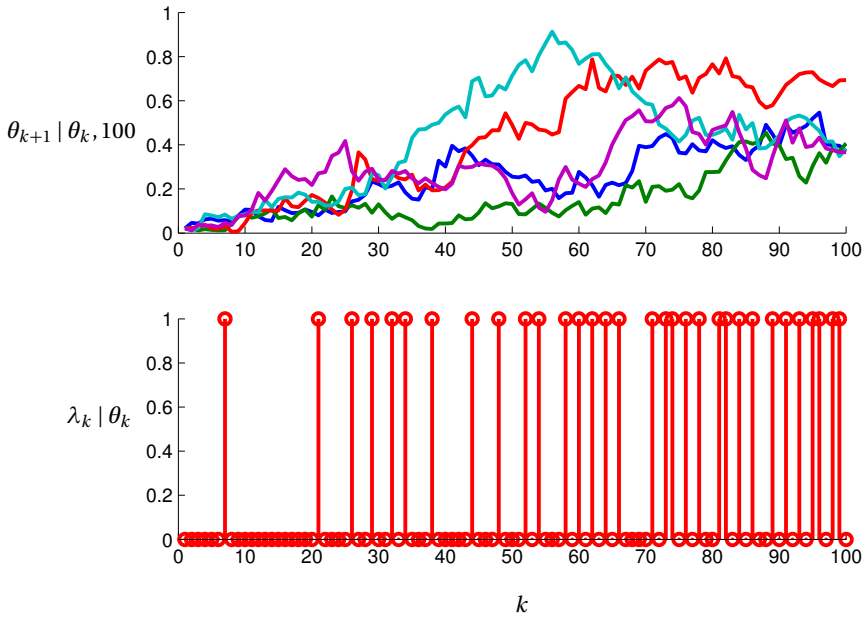

Figure 2: Five sample paths of the process $\theta_{k+1} \mid \theta_k, 100$ and a sample path of $\lambda_k \mid \theta_k$ corresponding to the red colored $\theta_{k+1} \mid \theta_k, 100$.

The most likely value of the model uncertainty θ_{k+1} corresponds to the model uncertainty at the previous time step θ_k, and increasing the value of the tuning parameter S reduces the probability of θ_{k+1} deviating significantly from θ_k. The hierarchical state-space model for the system (17)–(19) is illustrated by the directed acyclic graph (DAG) in Figure 3.

3 Estimation methods

From the whiteness and the mutual independence assumptions of the noise processes it follows that x_k, r_k and Λ_k are Markov processes

$$\begin{aligned} &p(x_k, r_k, \Lambda_k \mid x_{0:k-1}, r_{0:k-1}, \Lambda_{0:k-1}) \\ &= p(x_k \mid x_{k-1}) p(r_k \mid r_{k-1}) \mathsf{P}(\Lambda_k \mid \Lambda_{k-1}), \end{aligned} \tag{28}$$

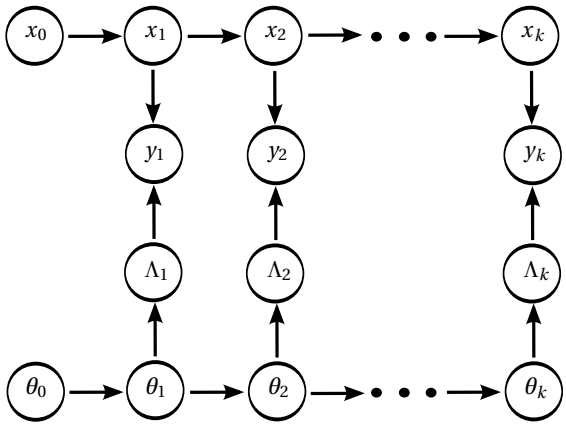

Figure 3: DAG of the hierarchical state-space model

and therefore

$$\begin{aligned} &p(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \\ &= p(x_0)p(r_0)\mathsf{P}(\Lambda_0)\prod_{i=1}^{k} p(x_i \mid x_{i-1})p(r_i \mid r_{i-1})\mathsf{P}(\Lambda_i \mid \Lambda_{i-1}), \end{aligned} \tag{29}$$

where

$$p(x_k \mid x_{k-1}) = p_{w_k}(x_k - F_{k-1}x_{k-1}) = \mathsf{N}(x_k \mid F_{k-1}x_{k-1}, Q_{k-1}) \tag{30}$$

$$p(r_k \mid r_{k-1}) = p_{\epsilon_k}(r_k - \Phi_{k-1}r_{k-1}) = \mathsf{N}(r_k \mid \Phi_{k-1}r_{k-1}, \Sigma_k), \tag{31}$$

when the state transition models are defined with additive white Gaussian process noises. The subscripted expression $p_x(\cdot)$ is occasionally used to emphasize that we are considering the probability density of the random variable x, but we omit the subscript whenever it is clear from the context what random variable we are considering. Also, the observations are conditionally independent

$$p(y_{1:k} \mid x_{0:k}, r_{0:k}, \Lambda_{0:k}) = \prod_{i=1}^{k} p(y_i \mid x_i, r_i, \Lambda_i). \tag{32}$$

The *likelihood function* (32) can be expressed as

$$\prod_{i=1}^{k} p_{v_i}(y_i - h_i(x_i) - \Lambda_i r_i) = \prod_{i=1}^{k} \mathsf{N}\left(y_i \mid h_i(x_i) + \Lambda_i r_i, R_i\right), \tag{33}$$

in the case of additive white Gaussian observation noise (2). If the noise is non-additive, then the form of the likelihood function can be significantly more complex.

3.1 Bayesian estimation

The problem (14)–(16) can be solved using the general framework of Bayesian statistics [33], [19], [61], [21], [6]. A complete solution for the system would be the joint posterior distribution of the state, errors and indicator variables given all the observations

$$\begin{aligned} p(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) &= \frac{p(y_{1:k} \mid x_{0:k}, r_{0:k}, \Lambda_{0:k}) p(x_{0:k}, r_{0:k}, \Lambda_{0:k})}{p(y_{1:k})} \\ &\propto \prod_{i=1}^{k} p(y_i \mid x_i, r_i, \Lambda_i) p(x_i \mid x_{i-1}) p(r_i \mid r_{i-1}) \mathsf{P}(\Lambda_i \mid \Lambda_{i-1}) \times \\ &\quad \times p(x_0) p(r_0) \mathsf{P}(\Lambda_0). \end{aligned} \tag{34}$$

In many of the algorithms discussed in the later sections, we are considering the joint distribution of the state and the errors

$$\begin{aligned} p(x_{0:k}, r_{0:k} \mid y_{1:k}) &= \sum_{\Lambda_{0:k}} p(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) \\ &= \sum_{\Lambda_{0:k}} p(x_{0:k}, r_{0:k} \mid y_{1:k}, \Lambda_{0:k}) p(\Lambda_{0:k} \mid y_{1:k}) \\ &\propto \sum_{\Lambda_{0:k}} \left(\prod_{i=1}^{k} p(y_i \mid x_i, r_i, \Lambda_i) p(x_i \mid x_{i-1}) p(r_i \mid r_{i-1}) \right) \times \\ &\quad \times p(x_0) p(r_0) p(\Lambda_{0:k} \mid y_{1:k}), \end{aligned} \tag{35}$$

where the notation $\sum_{\Lambda_{0:k}}$ is used for the summation over all possible values of $\Lambda_{0:k}$.

If we are interested only in the effect of the sensor errors on the actual state $x_{0:k}$ and not the values of the errors $r_{0:k}$, we treat them as nuisance parameters and integrate them out from the joint posterior.

The computational load of evaluating (34) can be overwhelming, especially with large k, as the dimensions of the state $x_{0:k}$, the errors

$r_{0:k}$ and the indicator variables $\Lambda_{0:k}$ grow at every time step. However, often we are interested only in the distribution of the most recent state

$$p(x_k, r_k \mid y_{1:k}) = \sum_{\Lambda_{0:k}} p(x_k, r_k \mid y_{1:k}, \Lambda_{0:k}) p(\Lambda_{0:k} \mid y_{1:k}), \tag{36}$$

called the *posterior filtering distribution,* that can be expressed recursively when (28) and (32) hold.

In this work we often solve the terms of (36) separately, because in our applications the posterior filtering distribution conditioned on the indicator variable history $\Lambda_{0:k}$ will be feasible to express approximately in closed form with a two-step process called *Bayesian filtering*. We have the initial state distribution (3). Then we use the *prediction step*

$$\begin{aligned} p(x_k, r_k \mid y_{1:k-1}, \Lambda_{0:k}) = \int & p(x_k, r_k \mid x_{k-1}, r_{k-1}) \times \\ & \times p(x_{k-1}, r_{k-1}, \mid y_{1:k-1}, \Lambda_{0:k}) \mathrm{d}(x_{k-1}, r_{k-1}), \end{aligned} \tag{37}$$

to get the prior predictive distribution. The second part of the Bayesian filtering is the *update step*

$$p(x_k, r_k \mid y_{1:k}, \Lambda_{0:k}) = \frac{p(y_k \mid x_k, r_k, \Lambda_{0:k}) p(x_k, r_k \mid y_{0:k-1}, \Lambda_{0:k})}{p(y_k \mid y_{1:k-1}, \Lambda_{0:k})}. \tag{38}$$

Sometimes it is beneficial to evaluate other marginal distributions $p(x_i \mid y_{1:k})$ of the posterior distribution. These are called posterior smoothing distributions, and from them we can obtain estimators of the state x_i with smaller variance than from the filtering distribution at ith time step. This is due to the fact that smoothing distributions are obtained with more information about the state. The linear Gaussian system is a special case in which we are able to find the posterior filtering and smoothing distributions $p(x_i \mid y_{1:k}, \Lambda_{1:k})$ in closed form using Kalman filter (KF) and Rauch-Tung-Striebel smoother (RTS) algorithms, respectively [38], [59], [4], [37]. In the general case, the posterior filtering and smoothing distributions are intractable and we have to resort to approximative methods.

The posterior distribution (34) is the complete solution for the dynamic estimation problem (14) – (16). In theory, the posterior distribution contains all the information about the system given the models, the prior distributions and the observations. In addition, the Bayesian approach to the problem enables the estimation of more complex systems, as we could use nonlinear models, non-additive noise, and even hierarchical models [33], [19], [P4].

Although it is the complete solution, the posterior filtering distribution (38) contains often too much information for many practical applications and a single point estimate of the state and summarizing statistics such as the variance or quantiles of the distribution would often be more appropriate. A Bayesian point estimate $\hat{x}_{k|k}$ can be derived as the point minimizing the expected loss of the estimate [33], [10]. The loss is quantified with the *loss function* $\mathscr{L}(x_k, \hat{x}_{k|k})$, and the optimal estimate can be obtained as the solution for the minimization problem

$$\hat{x}_{k|k} = \arg\min_{\theta_k} \int \mathscr{L}(x_k, \theta_k) p(x_k \mid y_{1:k}) \mathrm{d}x_k. \tag{39}$$

Throughout this work we use the loss function

$$\mathscr{L}(x_k, \theta_k) = ||x_k - \theta_k||^2_{P^{-1}_{k|k}}, \tag{40}$$

that is minimized with the posterior mean [33]. We use notations $||\cdot||$ for the Euclidian norm and $||x||_A := \sqrt{x^T A x}$. Other often used Bayesian estimators are the median and the maximum a posteriori estimate.

3.2 Kalman filter

An important special case, in which the conditional posterior distribution can be computed analytically, is the linear Gaussian system

$$\begin{bmatrix} x_{k+1} \\ r_{k+1} \end{bmatrix} = \begin{bmatrix} F_k & 0 \\ 0 & \Phi_k \end{bmatrix} \begin{bmatrix} x_k \\ r_k \end{bmatrix} + \begin{bmatrix} w_k \\ \epsilon_{k+1} \end{bmatrix} \tag{41}$$

$$y_k = \begin{bmatrix} H_k & \Lambda_k \end{bmatrix} \begin{bmatrix} x_k \\ r_k \end{bmatrix} + v_k \tag{42}$$

$$\begin{bmatrix} x_0 \\ r_0 \end{bmatrix} \sim \mathsf{N}\left(\begin{bmatrix} x_{0|0} \\ r_{0|0} \end{bmatrix}, \begin{bmatrix} P_{0|0} & 0 \\ 0 & P^r_{0|0} \end{bmatrix} \right) \tag{43}$$

where

$$\begin{bmatrix} w_k \\ \epsilon_{k+1} \end{bmatrix} \sim \mathsf{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q_k & 0 \\ 0 & \Sigma_{k+1} \end{bmatrix} \right)$$

and $v_k \sim \mathsf{N}(0, R_k)$ are mutually independent white noise processes. The posterior distribution $p(x_{0:k}, r_{0:k} \mid y_{1:k}, \Lambda_{0:k})$ of the system (41)–(43) is a Gaussian [8], and the posterior filtering distribution (38) can be evaluated in closed form recursively using the KF method given in Algorithm 1 [38], where we use the shorthand notation

$$\begin{aligned} H_k(\Lambda_k) &:= \begin{bmatrix} H_k & \Lambda_k \end{bmatrix} \\ A_k &:= \begin{bmatrix} F_k & 0 \\ 0 & \Phi_k \end{bmatrix} \\ B_k &:= \begin{bmatrix} Q_k & 0 \\ 0 & \Sigma_{k+1} \end{bmatrix} \\ X_k &:= \begin{bmatrix} x_k \\ r_k \end{bmatrix} \quad X_0 \sim \mathsf{N}\left(X_{0|0}, V_{0|0}\right). \end{aligned}$$

We use the notation $Z_k(\Lambda_{0:k})$ for the means and covariances Z_k conditioned on the indicator variable histories $\Lambda_{0:k}$.

The posterior filtering distribution is a Gaussian

$$p(x_k, r_k \mid y_{1:k}, \Lambda_{0:k}) = \mathsf{N}\left(x_k, r_k \mid X_{k|k}(\Lambda_{0:k}), V_{k|k}(\Lambda_{0:k})\right), \tag{44}$$

and hence so are the marginal distributions. We obtain the posterior filtering distribution of the state x_k

$$p(x_k \mid y_{1:k}, \Lambda_{0:k}) = \mathsf{N}\left(x_k \mid x_{k|k}(\Lambda_{0:k}), P_{k|k}(\Lambda_{0:k})\right) \tag{45}$$

Algorithm 1 KF

for $k = 1, \ldots, T$ **do**
  $X_{k|k-1}(\Lambda_{0:k-1}) = A_{k-1} X_{k-1|k-1}(\Lambda_{0:k-1})$
  $V_{k|k-1}(\Lambda_{0:k-1}) = A_{k-1} V_{k-1|k-1}(\Lambda_{0:k-1}) A_{k-1}^T + B_{k-1}$
  $S_k(\Lambda_{0:k}) = H_k(\Lambda_k) V_{k|k-1}(\Lambda_{0:k-1}) H_k(\Lambda_k)^T + R_k$
  $K_k(\Lambda_{0:k}) = V_{k|k-1}(\Lambda_{0:k-1}) H_k(\Lambda_k)^T S_k(\Lambda_{0:k})^{-1}$
  $X_{k|k}(\Lambda_{0:k}) = X_{k|k-1}(\Lambda_{0:k-1}) + K_k(\Lambda_{0:k})(y_k - H_k(\Lambda_k) X_{k|k-1}(\Lambda_{0:k-1}))$
  $V_{k|k}(\Lambda_{0:k}) = V_{k|k-1}(\Lambda_{0:k-1}) + K_k(\Lambda_{0:k}) H_k(\Lambda_k) V_{k|k-1}(\Lambda_{0:k-1})$
end for

by selecting the corresponding components from (44). It is also possible to find the Gaussian smoothing distributions $p(x_i, r_i \mid y_{1:k}, \Lambda_{0:k})$ for $i = 0, \ldots, k-1$, after the Gaussian distributions $p(x_i, r_i \mid y_{1:i}, \Lambda_{0:i})$ are found using the KF algorithm. The smoothing distributions

$$p(x_i, r_i \mid y_{1:k}, \Lambda_{0:k}) = \mathsf{N}\left(x_i, r_i \mid X_{i|k}(\Lambda_{0:k}), V_{i|k}(\Lambda_{0:k})\right) \tag{46}$$

are obtained with the RTS-smoother [59] given by Algorithm 2.

Algorithm 2 Rauch-Tung-Stribel smoother

for $k = T-1, \ldots, 0$ **do**
  $X_{k+1|k}(\Lambda_{0:k}) = A_k X_{k|k}(\Lambda_{0:k})$
  $V_{k+1|k}(\Lambda_{0:k}) = A_k V_{k|k}(\Lambda_{0:k}) A_k^T + B_k$
  $G_k = V_{k|k}(\Lambda_{0:k}) A_k^T V_{k+1|k}(\Lambda_{0:k})^{-1}$
  $X_{k|T}(\Lambda_{0:T}) = X_{k|k}(\Lambda_{0:k}) + G_k(X_{k+1|T}(\Lambda_{0:T}) - X_{k+1|k}(\Lambda_{0:k}))$
  $V_{k|T}(\Lambda_{0:T}) = V_{k|k}(\Lambda_{0:k}) + G_k(V_{k+1|T}(\Lambda_{0:T}) - V_{k+1|k}(\Lambda_{0:k})) G_k^T$
end for

Although many positioning systems rely on nonlinear equations, there are also linear observation equations. Examples of linear measurement equations would be Doppler measurements [39], coverage area measurements [41], and also position coordinates obtained by a positioning system. Using the position coordinates provided by existing positioning systems may cause the observation noise process to be non-white, depending on the algorithms that the system uses to estimate the position.

3.3 Extended Kalman filter

The observation equation (2) is often nonlinear in positioning. For example, the range measurements between MS and a base station (BS) with known coordinates $p_k^{(i)}$

$$y_{k,i} = h_{k,i}(x_k) + s_{k,i} + v_{k,i} = ||p_k^{(i)} - x_{k,1:d}|| + s_{k,i} + v_{k,i} \qquad (47)$$

can be obtained using time differences between the emission and the reception of a signal, or measuring received signal strengths and using a path loss model for the attenuation of the signal strength with distance [15]. At each time step we may obtain measurements from several sources. The observation vector used in positioning is

$$y_k = \begin{bmatrix} y_{k,1} & \cdots & y_{k,n_y} \end{bmatrix}^T. \qquad (48)$$

The notation $x_{k,1:d}$ refers to the d positioning coordinates of the state. In GPS positioning, the basic observation type is a biased range measurement between MS and a satellite with coordinates $p_k^{(i)}$

$$y_{k,i} = h_{k,i}(x_k) + s_{k,i} + v_{k,i} = ||p_k^{(i)} - x_{k,1:3}|| + x_{k,4} + s_{k,i} + v_{k,i}, \qquad (49)$$

where the bias $x_{k,4}$ is caused by the difference in MS and satellite clocks, and is the same for each observation at any time step, excluding the effect of the satellite clock error. In reality, the observation equation has more additive error components such as ionospheric and trophospheric delays that in principle could be estimated and their influence eliminated [39, 52]. Because the range measurement are derived from the signals traveling between MS and some other stations, the additive sensor error s_k in the range measurements can be caused by the attenuations and reflections of the signal between MS and the station.

As the observation equation is nonlinear, we are unable to use KF to solve the posterior distribution of the system (1)–(3), but instead resort to approximate methods. The standard approach is to approximate the nonlinear functions as linear, and apply KF to the approximate system. In cases where we are able to compute the Jacobian matrix of the measurement equation, the most popular

algorithm is the (first order) extended Kalman filter (EKF) [8], [16]. In EKF, the nonlinear measurement equations are approximated using the first order Taylor approximation about the predicted distribution mean $[x_{k|k-1}(\Lambda_{0:k-1})^T, r_{k|k-1}(\Lambda_{0:k-1})^T]^T$. The EKF algorithm is given in Algorithm 3. For simplicity we define

$$g_k(X_k, \Lambda_k) = h_k(x_k) + \Lambda_k r_k. \tag{50}$$

In the algorithm $g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k)$ is the Jacobian matrix of $g_k(\cdot, \Lambda_k)$ with respect to X_k evaluated at $X_{k|k-1}(\Lambda_{0:k-1})$.

Algorithm 3 EKF

for $k = 1, \ldots, T$ **do**
  $X_{k|k-1}(\Lambda_{0:k-1}) = A_{k-1} X_{k-1|k-1}(\Lambda_{0:k-1})$
  $V_{k|k-1}(\Lambda_{0:k-1}) = A_{k-1} V_{k-1|k-1}(\Lambda_{0:k-1}) A_{k-1}^T + B_{k-1}$
  $g_k(X_k, \Lambda_k) \approx g_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k) + g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k)(X_k - X_{k|k-1}(\Lambda_{0:k-1}))$
  $S_k(\Lambda_{0:k}) = g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k) V_{k|k-1}(\Lambda_{0:k-1}) g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k)^T + R_k$
  $K_k(\Lambda_{0:k}) = V_{k|k-1}(\Lambda_{0:k-1}) g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k)^T S_k(\Lambda_{0:k})^{-1}$
  $X_{k|k}(\Lambda_{0:k}) = X_{k|k-1}(\Lambda_{0:k-1}) + K_k(\Lambda_{0:k})(y_k - g_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k))$
  $V_{k|k}(\Lambda_{0:k}) = V_{k|k-1}(\Lambda_{0:k-1}) + K_k(\Lambda_{0:k}) g'_k(X_{k|k-1}(\Lambda_{0:k-1}), \Lambda_k) V_{k|k-1}(\Lambda_{0:k-1})$
end for

We approximate the posterior distribution at each time step as Gaussian with the mean and the covariance given by EKF

$$p(x_k \mid y_{1:k}, \Lambda_{0:k}) \approx \mathrm{N}\left(x_{k|k}(\Lambda_{0:k}), P_{k|k}(\Lambda_{0:k})\right). \tag{51}$$

It is important to note that the EKF algorithm does not evaluate the mean and the covariance of the posterior distribution directly, but instead approximates the nonlinear functions using Taylor series and hence we are able to evaluate the moments for this approximative system. In a sense this could be considered to be a one-step approximation for the filtering problem. With very strong nonlinearities in the observation functions $h_k(\cdot)$ the performance of EKF can be severely degraded. However, range measurements with long distances between MS and BS can be estimated locally very well with even first order Taylor approximation, and in this case EKF performs very accurately and is very fast.

There are Kalman-type filtering methods that are used to directly approximate the moments of the posterior distribution. The unscented

Kalman filter uses the unscented transformation to approximate the mean and covariance of a nonlinearly transformed Gaussian random variable with a weighted sum of the nonlinear function evaluations [35], [34], [36]. A similar approach is used in Kalman-type filters that evaluate the mean and the covariance of the posterior filtering distribution by approximating them with numerical integration, or cubature formulas. These filters are referred to as cubature Kalman filters [5], [58].

3.4 Gaussian mixture filter

In Section 3.2 we discussed the KF algorithm that can be used to evaluate the Gaussian posterior distribution $p(x_k, r_k \mid y_{1:k}, \Lambda_{0:k})$ in closed form. However, the indicator variable process Λ_k is generally unknown and also stochastic, thus making the evaluation of $p(x_k, r_k \mid y_{1:k})$ a more complicated process. The Markov chain model (10) results in a Gaussian mixture (GM) posterior distribution, and the posterior filtering distribution can be found using the GM filter (GMF) [64], [3],[2].

To evaluate the posterior filtering distribution (38) we are required to evaluate the posterior probability of the whole history of indicator variables $\mathsf{P}(\Lambda_{0:k} \mid y_{1:k})$. The models (9) and (10)–(11) for the indicator variables make it possible to do this in recursive form.

Due to the Markovian property of the indicator variable models, we can write the predictive probability mass function as

$$\begin{aligned}\mathsf{P}(\Lambda_{0:k} \mid y_{1:k-1}) &= \mathsf{P}(\Lambda_k \mid \Lambda_{0:k-1}, y_{1:k})\mathsf{P}(\Lambda_{0:k-1} \mid y_{1:k-1}) \\ &= \mathsf{P}(\Lambda_k \mid \Lambda_{k-1})\mathsf{P}(\Lambda_{0:k-1} \mid y_{1:k-1}). \end{aligned} \tag{52}$$

The posterior probability of the indicator history at time k can be evaluated by multiplying the posterior probability of the indicator

history from the time $k-1$ with the likelihood of the history and the transition probability

$$\begin{aligned}\mathsf{P}(\Lambda_{0:k} \mid y_{1:k}) &= \frac{p(y_k \mid \Lambda_{0:k}, y_{1:k-1})\mathsf{P}(\Lambda_{0:k} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\ &= \frac{p(y_k \mid \Lambda_{0:k}, y_{1:k-1})\mathsf{P}(\Lambda_k \mid \Lambda_{k-1})\mathsf{P}(\Lambda_{0:k-1} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})}. \end{aligned} \tag{53}$$

The predicted observation distribution

$$\begin{aligned}&p(y_k \mid \Lambda_{0:k}, y_{1:k-1}) \\ &= \int p(y_k \mid x_k, r_k, \Lambda_k) p(x_k, r_k \mid \Lambda_{0:k}, y_{1:k-1}) \mathrm{d}(x_k, r_k)\end{aligned} \tag{54}$$

is approximated as a Gaussian distribution

$$p(y_k \mid \Lambda_{0:k}, y_{1:k-1}) \approx \mathsf{N}(z_k(\Lambda_{0:k}) \mid 0, S_k(\Lambda_{0:k})) \tag{55}$$

where the mean and the covariance

$$z_k(\Lambda_{0:k}) = y_k - H_k(\Lambda_k) X_{k|k-1}(\Lambda_{0:k-1}) \tag{56}$$

$$S_k(\Lambda_{0:k}) = H_k(\Lambda_k) V_{k|k}(\Lambda_{0:k}) H_k(\Lambda_k)^T + R_k \tag{57}$$

are the innovation and the innovation covariance. The likelihood of the indicator history (55), measures how well the current observation is explained by a particular $\Lambda_{0:k}$ given the observation history $y_{1:k-1}$

The GMF algorithm evaluates the posterior distribution in closed form in the case of a linear dynamic system with additive sensor errors by running a growing bank of KFs and computing the probability for each of the Gaussian components. In the case of nonlinear systems, the posterior can be approximated analogously by running a bank of Kalman-type filters, such as EKFs.

The dimension of $\Lambda_{0:k}$ grows at each time step exponentially and a closed form solution of the posterior may become infeasible after only a few time steps. Hence approximation techniques have been developed for the problem. One solution is to remove mixture components with small probability $\mathsf{P}(\Lambda_{0:i} \mid y_{1:i})$ to control the number of components. Other approaches are often based on merging some of

the components according to a strategy. The simplest technique is to merge all the Gaussian components into a single one at each time step. This could be done by computing the mean and the covariance of the mixture distribution and approximating it with a Gaussian with corresponding mean and covariance [55], [62]. The mean and covariance of the GM posterior filtering distribution have closed form expressions

$$X_{k|k} = \sum_{\Lambda_{0:k}} \mathsf{P}(\Lambda_{0:k} \mid y_{1:k}) X_{k|k}(\Lambda_{0:k}) \tag{58}$$

$$V_{k|k} = \sum_{\Lambda_{0:k}} \mathsf{P}(\Lambda_{0:k} \mid y_{1:k}) \Big(V_{k|k}(\Lambda_{0:k}) + (X_{k|k}(\Lambda_{0:k}) - X_{k|k})(X_{k|k}(\Lambda_{0:k}) - X_{k|k})^T \Big). \tag{59}$$

Other merging techniques include the generalized pseudo-Bayesian approach of the first and the second order, and the interacting multiple model algorithm (IMM) [8], all of which keep the number of components constant at each time step, but have a different strategy of choosing which components to merge.

A novel sequential merging method

We consider the system (17)-(19) and the white indicator variable model (9). Let the approximate posterior filtering distribution at time $k-1$ be

$$p(x_{k-1} \mid y_{1:k-1}) \approx \mathsf{N}\left(x_{k-1|k-1}, P_{k-1|k-1}\right). \tag{60}$$

If we have m conditionally independent observations at time k, and each of them may have an additive error present, the posterior filter-

ing distribution will have 2^m components. The posterior distribution can be written as

$$
\begin{aligned}
p(x_k \mid y_{1:k}) &= \sum_{\lambda_{k,1}} \cdots \sum_{\lambda_{k,m}} \mathsf{P}(\lambda_{k,1}) \cdots \mathsf{P}(\lambda_{k,m}) p(x_k \mid y_{1:k}, \lambda_{k,1}, \ldots, \lambda_{k,m}) \\
&\propto \sum_{\lambda_{k,1}} \cdots \sum_{\lambda_{k,m}} \mathsf{P}(\lambda_{k,1}) \cdots \mathsf{P}(\lambda_{k,m}) \prod_{l=1}^{m} p(y_{k,l} \mid x_k, \lambda_{k,l}) p(x_k \mid y_{1:k-1}) \\
&\propto \sum_{\lambda_{k,2}} \cdots \sum_{\lambda_{k,m}} \mathsf{P}(\lambda_{k,2}) \cdots \mathsf{P}(\lambda_{k,m}) \prod_{l=2}^{m} p(y_{k,l} \mid x_k, \lambda_{k,l}) p(x_k \mid y_{1:k-1}, y_{k,1}),
\end{aligned}
$$

where $p(x_k \mid y_{1:k-1}, y_{k,1})$ is a two-component GM distribution. In the novel merging method, we merge two-component GM distributions $p(x_k \mid y_{1:k}, y_{k,1:l})$ into a single Gaussian using moment matching before evaluating the posterior $p(x_k \mid y_{1:k}, y_{k,1:l+1})$ [P3]. Note that the order in which observations are process has an influence on the approximative posterior.

3.5 Sampling based methods

Sequential Monte Carlo (SMC) methods, or particle filters, are sampling based methods that use a weighted sample to approximate the posterior distribution empirically [19], [60]. The approximation is done with a set of N weighted samples, or particles, $\{x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)}, w_{0:k}^{(i)}\}_{i=1}^{N}$, and the approximative posterior distribution is

$$
\begin{aligned}
&\hat{p}_N(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) \\
&= \sum_{i=1}^{N} w_{0:k}^{(i)} \delta\left((x_{0:k}, r_{0:k}, \Lambda_{0:k}) - \left(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)}\right)\right),
\end{aligned} \tag{61}
$$

where $\delta(\cdot)$ is the delta distribution. Optimally the particles would be generated from the posterior distribution, but in practice *importance sampling* based on the strong law of large numbers (SLLN) is used, as drawing a sample from the posterior may be impossible. Let $\pi(x_{0:k}, r_{0:k}, \Lambda_{0:k})$ be a distribution with a support that contains

the support of $p(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k})$. The mean of a function $g(\cdot)$ of random variables $x_{0:k}, r_{0:k}, \Lambda_{0:k}$ can be written as

$$\begin{aligned}
&\mathsf{E}(g(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \mid y_{1:k})) \\
&= \int g(x_{0:k}, r_{0:k}, \Lambda_{0:k}) p(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) \mathrm{d}(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \\
&= \frac{\int g(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \frac{p(x_{0:k}, r_{0:k}, \Lambda_{0:k} | y_{1:k})}{\pi(x_{0:k}, r_{0:k}, \Lambda_{0:k})} \pi(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \mathrm{d}(x_{0:k}, r_{0:k}, \Lambda_{0:k})}{\int \frac{p(x_{0:k}, r_{0:k}, \Lambda_{0:k} | y_{1:k})}{\pi(x_{0:k}, r_{0:k}, \Lambda_{0:k})} \pi(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \mathrm{d}(x_{0:k}, r_{0:k}, \Lambda_{0:k})}.
\end{aligned} \tag{62}$$

SLLN states that the sample mean converges almost surely to the expected value

$$\begin{aligned}
&\frac{\frac{1}{N} \sum_{i=1}^{N} g(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)}) \frac{p(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)} | y_{1:k})}{\pi(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)})}}{\frac{1}{N} \sum_{j=1}^{N} \frac{p(x_{0:k}^{(j)}, r_{0:k}^{(j)}, \Lambda_{0:k}^{(j)} | y_{1:k})}{\pi(x_{0:k}^{(j)}, r_{0:k}^{(j)}, \Lambda_{0:k}^{(j)})}} \\
&= \sum_{i=1}^{N} g(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)}) w_{0:k}^{(i)} \rightarrow \mathsf{E}(g(x_{0:k}, r_{0:k}, \Lambda_{0:k}) \mid y_{1:k})),
\end{aligned} \tag{63}$$

as the size of samples drawn from $\pi(\cdot)$ approaches infinity. Thus the sample $\{x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)}\}_{i=1}^{N}$ drawn from the importance distribution $\pi(x_{0:k}, r_{0:k}, \Lambda_{0:k})$ and the weights

$$w_{0:k}^{(i)} = \frac{\frac{p(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)} | y_{1:k})}{\pi(x_{0:k}^{(i)}, r_{0:k}^{(i)} \Lambda_{0:k}^{(i)})}}{\sum_{j=1}^{N} \frac{p(x_{0:k}^{(j)}, r_{0:k}^{(j)}, \Lambda_{0:k}^{(j)} | y_{1:k})}{\pi(x_{0:k}^{(j)}, r_{0:k}^{(j)} \Lambda_{0:k}^{(j)})}}, \tag{64}$$

can be used to estimate the expected value of any function $g(x_{0:k}, r_{0:k}, \Lambda_{0:k})$, and thus can be considered to be a representation of the posterior distribution.

The importance distribution can be chosen freely within certain mild restrictions [19], and this allows for the evaluation of the posterior distribution to be performed sequentially as follows. Due to the

Markovian property of the parameters, the posterior distribution can be written

$$\begin{aligned} & p(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) \\ & \propto p(y_k \mid x_k, r_k, \Lambda_k) p(x_k \mid x_{k-1}) p(r_k \mid r_{k-1}) \mathsf{P}(\Lambda_k \mid \Lambda_{k-1}) \times \\ & \quad \times p(x_{0:k-1}, r_{0:k-1}, \Lambda_{0:k-1} \mid y_{1:k-1}). \end{aligned} \tag{65}$$

The sequential importance sampling (SIS) algorithm is based on the selection of an importance distribution that can be factored in a certain way that allows the computation of the particle weights (64) recursively. In particular, the choice

$$\begin{aligned} & \pi(x_{0:k}, r_{0:k}, \Lambda_{0:k} \mid y_{1:k}) \\ & = p(x_k \mid x_{k-1}) p(r_k \mid r_{k-1}) \mathsf{P}(\Lambda_k \mid \Lambda_{k-1}) \pi(x_{0:k-1}, r_{0:k-1}, \Lambda_{0:k-1} \mid y_{1:k-1}). \end{aligned} \tag{66}$$

results in the particle weights

$$w_{0:k}^{(i)} \propto w_{0:k-1}^{(i)} p(y_k \mid x_k^{(i)}, r_k^{(i)}, \Lambda_k^{(i)}). \tag{67}$$

where the new samples $(x_{0:k}^{(i)}, r_{0:k}^{(i)}, \Lambda_{0:k}^{(i)})$ are obtained by propagating the old samples $(x_{k-1}^{(i)}, r_{k-1}^{(i)}, \Lambda_{k-1}^{(i)})$ according to the state transition distributions and adding them to $(x_{0:k-1}^{(i)}, r_{0:k-1}^{(i)}, \Lambda_{0:k-1}^{(i)})$. SIS is only a restricted version of the more general particle filter but is very popular due to its simplicity and general applicability. When we are interested only in the posterior filtering distribution, it is sufficient to store only the current particles $\{x_k^{(i)}, r_k^{(i)}, \Lambda_k^{(i)}, w_{0:k}^{(i)}\}_{i=1}^N$.

In practice SMC methods are plagued by the degeneracy phenomenon, where after a number of sequential steps of the algorithm, all but one particle will have almost zero normalized weight. The degeneracy problem is often solved in practice by using a procedure called resampling, where particles with low weights are eliminated and particles with large weights are multiplied. This can be done for example by sampling the particle indices from the multinomial distribution defined by the weights $\{w_{0:k}^{(i)}\}_{i=1}^N$, and forming the new particles according to the indices. After sampling, the same weight $w_{0:k}^{(i)} = \frac{1}{N}$ is assigned for all the particles. The resampling could be

performed at each time step as one does in the *bootstrap filter*, or after the effective sample size

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} \left(w_{0:k}^{(i)} \right)^2}, \tag{68}$$

which is used to measure the degeneracy of the weights, becomes smaller than some threshold value N_{Th}[40]. The SMC methods using resampling are referred to as *sampling importance resampling* (SIR) filters (Algorithm 4).

Algorithm 4 SIR particle filter for the system (14)–(16)

Set $w_0^{(i)} = \frac{1}{N}, i = 1, \ldots, N$
Draw $x_0^{(i)} \sim \mathsf{N}\left(x_{0|0}, P_{0|0}\right), r_0^{(i)} \sim \mathsf{N}\left(r_{0|0}, \Sigma_{0|0}\right), \Lambda_0^{(i)} \sim \mathsf{P}(\Lambda_0), i = 1, \ldots, N$
for $k = 1, \ldots, T$ **do**
  Draw $x_k^{(i)} \sim \mathsf{N}\left(F_{k-1} x_{k-1}^{(i)}, Q_{k-1}\right), r_0^{(i)} \sim \mathsf{N}\left(\Phi_{k-1} r_{k-1}^{(i)}, \Sigma_k\right),$
  $\Lambda_k^{(i)} \sim \mathsf{P}(\Lambda_k \mid \Lambda_{k-1}^{(i)}), i = 1, \ldots, N$
  Compute and normalize the weights
  $w_{0:k}^{(i)} \propto w_{0:k-1}^{(i)} p(y_k \mid x_k^{(i)}, r_k^{(i)}, \Lambda_k^{(i)}), i = 1, \ldots, N$
  if $N_{\text{eff}} < N_{\text{Th}}$ **then**
    Resample $x_k^{(i)}, r_k^{(i)}, \Lambda_k^{(i)}$ based on $w_{0:k}^{(i)}$
    Set $w_{0:k}^{(i)} = \frac{1}{N}, i = 1, \ldots, N$
  end if
end for

Marginalized particle filtering

As discussed previously, the conditional distribution of the state and the sensor errors given the indicator history $\Lambda_{0:k}$ can be solved in closed form for linear systems and approximatively for nonlinear systems

$$p(X_k \mid y_{1:k}, \Lambda_{0:k}) \approx \mathsf{N}\left(X_{k|k}(\Lambda_{0:k}), V_{k|k}(\Lambda_{0:k})\right). \tag{69}$$

When part of the problem can be solved in closed form, it is possible to use marginalized particle filters (MPFs, also known as Rao-Blackwellized filters) to approximate the posterior distribution more accurately with fewer particles [20].

In the state-space problem with the additive sensor errors we use SIR to approximate

$$\hat{p}_N(\Lambda_{0:k} \mid y_{1:k}) = \sum_{i=1}^{N} w_{0:k}^{(i)} \delta(\Lambda_{0:k} - \Lambda_{0:k}^{(i)}) \approx \mathsf{P}(\Lambda_{0:k} \mid y_{1:k}). \tag{70}$$

For each particle $\{\Lambda_{0:k}^{(i)}, w_{0:k}^{(i)}\}_{i=1}^{N}$, we can solve, at least approximately, the mean and the covariance of $p(x_k \mid y_{1:k}, \Lambda_{0:k}^{(i)})$. The unnormalized weight is computed for each of the particles as

$$\begin{aligned} w_{0:k}^{(i)} &\propto w_{0:k-1}^{(i)} p(y_k \mid y_{1:k-1}, \Lambda_{0:k}^{(i)}) \\ &= w_{0:k-1}^{(i)} \mathsf{N}\left(z_k\left(\Lambda_{0:k}^{(i)}\right) \mid 0, S_k\left(\Lambda_{0:k}^{(i)}\right)\right). \end{aligned} \tag{71}$$

The weights degenerate with time steps, and in practice one has to perform resampling the same way as for all SMC methods.

The approximate posterior filtering distribution obtained by MPF algorithm is a GM distribution. In a sense MPF is a GM filter where the unlikely components are pruned automatically in the resampling procedure.

Algorithm 5 Marginalized SIR particle filter for the system (14)–(16)

1: Set $w_0^{(i)} = \frac{1}{N}, i = 1, \ldots, N$
2: Draw $\Lambda_0^{(i)} \sim \mathsf{P}(\Lambda_0), i = 1, \ldots, N$
3: **for** $k = 1, \ldots, T$ **do**
4: Draw $\Lambda_k^{(i)} \sim \mathsf{P}(\Lambda_k \mid \Lambda_{k-1}^{(i)}), i = 1, \ldots, N$
5: Set $\Lambda_{0:k}^{(i)} = \left[\Lambda_{0:k-1}^{(i)}, \Lambda_k^{(i)}\right]$
6: Evaluate $p(X_k \mid y_{1:k}, \Lambda_{0:k}^{(i)})$ using KF
7: Compute and normalize the weights
$$w_{0:k}^{(i)} \propto w_{0:k-1}^{(i)} p(y_k \mid X_k^{(i)}, \Lambda_{0:k}^{(i)}), i = 1, \ldots, N$$
8: **if** $N_{\text{eff}} < N_{\text{Th}}$ **then**
9: Resample $p(X_k \mid y_{1:k}, \Lambda_{0:k}^{(i)})$ based on $w_{0:k}^{(i)}$
10: Set $w_{0:k}^{(i)} = \frac{1}{N}, i = 1, \ldots, N$
11: **end if**
12: **end for**

3.6 Variational Bayes

It is possible to approximate the posterior distribution of the state using the expectation maximization [17],[11] or variational method [12],[P7]. We study the system (17)–(19) with white indicator variable process (9) and linear observation equations. Similar approximative method could be used in the nonlinear case using the linearized equations.

In the variational Bayes method the calculus of variations is used to find the optimal approximate distribution

$$q(x_{0:k}, \lambda_{0:k,1:m}) \approx p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})$$

by requiring that the distribution can be factored as

$$q(x_{0:k}, \lambda_{0:k,1:m}) = q_{x_{0:k}}(x_{0:k}) \prod_{i=1}^{k} \prod_{l=1}^{m} q_{\lambda_{i,l}}(\lambda_{i,l}). \tag{72}$$

In the following we will again leave out the subscripts of the distributions for notational convenience. The marginal distributions $q(\lambda_{1,1}), \ldots, q(\lambda_{k,m})$ and $q(x_{0:k})$ are found such that they minimize the Kullback-Leibler (KL) divergence of the approximative distribution with respect to the posterior distribution

$$\begin{aligned} &\mathrm{KL}(q(x_{0:k}, \lambda_{0:k,1:m}) || p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})) \\ &= \sum_{\lambda_{0:k,1:m}} \int q(x_{0:k}) \prod_{i=0}^{k} \prod_{l=1}^{m} q(\lambda_{i,l}) \log \frac{q(x_{0:k}) \prod_{i=0}^{k} \prod_{l=1}^{m} q(\lambda_{i,l})}{p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})} \mathrm{d}x_{0:k}. \end{aligned} \tag{73}$$

The calculus of variations is used to minimize the KL divergence with respect to $q(\lambda_{i,l})$ by fixing the marginal distribution not containing $\lambda_{i,l}$

$$q(x_{0:k}, \lambda_{0:k,1:m} \backslash \lambda_{i,l}) := q(x_{0:k}, \lambda_{1,1}, \ldots, \lambda_{1,m}, \ldots, \lambda_{i,l-1}, \lambda_{i,l+1}, \ldots, \lambda_{k,m}).$$

The marginal distribution $q(\lambda_{i,l})$ minimizing the KL-divergence is [12, p. 466]

$$\log q(\lambda_{i,l}) = \mathsf{E}_{x_{0:k}, \lambda_{0:k} \backslash \lambda_{i,l}} \left(\log p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k}) \right) + \text{const.}, \tag{74}$$

where the expectation is taken with respect to $q(x_{0:k}, \lambda_{0:k,1:m} \backslash \lambda_{i,l})$.

Because the posterior distribution is of the form

$$\begin{aligned} & p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k}) \\ & \propto \left(\prod_{i=1}^{k} \mathsf{N}\left(y_i \mid H_i x_i, R_i(\Lambda_i)\right) \mathsf{N}(x_i \mid F_{i-1} x_{i-1}, Q_{i-1}) \mathsf{P}(\Lambda_i) \right) \times \\ & \quad \times \mathsf{N}\left(x_0 \mid x_{0|0}, P_{0|0}\right) \mathsf{P}(\Lambda_0), \end{aligned} \tag{75}$$

we can evaluate the marginal $q(\lambda_{i,l})$ as

$$\begin{aligned} & \log q(\lambda_{i,l}) = \mathsf{E}_{x_{0:k}, \lambda_{0:k} \backslash \lambda_{i,l}} (\log p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})) + \text{const.} \\ & = \mathsf{E}_{x_{0:k}, \lambda_{0:k} \backslash \lambda_{i,l}} (\log p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})) + \text{const.} \\ & = (1-\lambda_{i,l}) \mathsf{E}_{x_i} \left(-\frac{1}{2} \log\det R_i - \frac{1}{2} \| y_{i,l} - H_{i,l} x_i \|^2_{R_{i,l}^{-1}} + \log(1-\theta_i) \right) \\ & \quad + \lambda_{i,l} \mathsf{E}_{x_i} \left(-\frac{1}{2} \log\det(R_i + \Sigma_{i,l}) - \frac{1}{2} \| y_{i,l} - H_{i,l} x_i \|^2_{(R_{i,l}+\Sigma_{i,l})^{-1}} + \log(\theta_i) \right) \\ & \quad + \text{const.} \end{aligned} \tag{76}$$

Taking the exponential, it can be shown that the optimal approximative $q(\lambda_{i,l})$ is of the same form as the prior $p(\lambda_{i,l})$, i.e. a Bernoulli-distribution

$$q(\lambda_{i,l}) = (1-\theta_{i,l})^{1-\lambda_{i,l}} \theta_{i,l}^{1-\lambda_{i,l}}, \tag{77}$$

and therefore $\mathsf{E}(\lambda_{i,l}) = \theta_{i,l}$. The probability of the sensor error being present is

$$\theta_{i,l} = \frac{\rho_{i,l}^{(2)}}{\rho_{i,l}^{(1)} + \rho_{i,l}^{(2)}}, \tag{78}$$

where

$$\begin{aligned} \log \rho_{i,l}^{(1)} = & -\frac{1}{2} \log\det R_{i,l} - \frac{1}{2} \| y_{i,l} - H_{i,l} x_{i|k} \|^2_{R_{i,l}^{-1}} \\ & - \frac{1}{2} \mathrm{tr}\left(H_{i,l}^T R_{i,l}^{-1} H_{i,l} P_{i|k} \right) + \log(1-\theta_i) \end{aligned} \tag{79}$$

$$\begin{aligned} \log \rho_{i,l}^{(2)} = & -\frac{1}{2} \log\det(R_{i,l} + \Sigma_{i,l}) - \frac{1}{2} \| y_{i,l} - H_{i,l} x_{i|k} \|^2_{(R_{i,l}+\Sigma_{i,l})^{-1}} \\ & - \frac{1}{2} \mathrm{tr}\left(H_{i,l}^T (R_{i,l} + \Sigma_{i,l})^{-1} H_{i,l} P_{i|k} \right) + \log \theta_i. \end{aligned} \tag{80}$$

After finding each of the marginal distributions $q(\lambda_{i,l})$, we evaluate the marginal distribution of the state $q(x_{0:k})$ as

$$\begin{aligned}
\log q(x_{0:k}) &= \mathsf{E}_{\lambda_{0:k,1:m}}\left(\log p(x_{0:k}, \lambda_{0:k,1:m} \mid y_{1:k})\right) + \text{const.} \\
&= \sum_{i=1}^{k}\sum_{l=1}^{m} -\frac{1}{2}\mathsf{E}_{\lambda_{i,l}}(1-\lambda_{i,l})\|y_{i,l} - H_{i,l}x_i\|^2_{R_{i,l}^{-1}} \\
&\quad -\frac{1}{2}\mathsf{E}_{\lambda_{i,l}}(\lambda_{i,l})\|y_{i,l} - H_{i,l}x_k\|^2_{(R_{i,l}+\Sigma_{i,l})^{-1}} \\
&\quad -\frac{1}{2}\|x_i - F_{i-1}x_{i-1}\|^2_{Q_{i-1}^{-1}} - \frac{1}{2}\|x_0 - x_{0|0}\|^2_{P_{0|0}^{-1}} + \text{const.} \\
&= \sum_{i=1}^{k} -\frac{1}{2}\|y_i - H_i x_i\|^2_{\Xi_i^{-1}} - \frac{1}{2}\|x_i - F_{i-1}x_{i-1}\|^2_{Q_{i-1}^{-1}} \\
&\quad -\frac{1}{2}\|x_0 - x_{0|0}\|^2_{P_{0|0}^{-1}} + \text{const.},
\end{aligned} \tag{81}$$

where

$$\begin{aligned}
\Xi_i^{-1} &= \begin{bmatrix} \Xi_{i,1}^{-1} & & \\ & \ddots & \\ & & \Xi_{i,m}^{-1} \end{bmatrix} \\
\Xi_{i,l}^{-1} &= \mathsf{E}_{\lambda_{i,1}}(1-\lambda_{i,1})R_{i,1}^{-1} + \mathsf{E}_{\lambda_{i,1}}(\lambda_{i,1})(R_{i,1}+\Sigma_{i,1})^{-1} \\
&= (1-\theta_{i,1})R_{i,1}^{-1} + \theta_{i,1}(R_{i,1}+\Sigma_{i,1})^{-1}.
\end{aligned} \tag{82}$$

It can be shown that the density $q(x_{0:k})$ is Gaussian and we can compute the marginals

$$q(x_i) = \mathsf{N}\left(x_{i|k}, P_{i|k}\right) \tag{83}$$

using KF and RTS-smoother, respectively described by Algorithms 1 and 2. The KF uses the observation noise matrices Ξ_i.

The set of equations (77) and (83) can be solved by a fixed-point iteration for which the convergence is guaranteed due to certain convexity properties of the error in the approximative distribution [12, p. 466]. This is the VB method that is summarized in Algorithm 6. Although convergence checks could be performed within the algorithm, we fix the number of iterations to M to control the computational costs. The resulting algorithm is very close to the EM-method for detecting change in the state transition model [17],[11],[44].

Algorithm 6 Variational Bayes method

$\theta_{i,l} \leftarrow 0, \quad i = 1,\ldots,k, \quad l = 1,\ldots,m$
for $m = 1,\ldots,M$ **do**
  $\Xi_{i,l}^{-1} \leftarrow (1-\theta_{i,l})R_{i,l}^{-1} + \theta_{i,l} W_{i,l}^{-1}, \quad i = 1,\ldots,k, \quad l = 1,\ldots,m$
  Evaluate $x_{i|i}, P_{i|i}, i = 1,\ldots,k$ using Algorithm 1
  Evaluate $x_{i|k}, P_{i|k}, i = k,\ldots,0$ using Algorithm 2
  for $i = 1,\ldots N$ **do**
    $\log \rho_{i,l}^{(1)} = -\frac{1}{2}\log\det R_{i,l} - \frac{1}{2}||y_{i,l} - H_{i,l}x_{i|k}||^2_{R_{i,l}^{-1}}$
      $-\frac{1}{2}\mathrm{tr}\left(H_{i,l}^T R_{i,l}^{-1} H_{i,l} P_{i|k}\right) + \log(1-\theta_i)$
    $\log \rho_{i,l}^{(2)} = -\frac{1}{2}\log\det(R_{i,l} + \Sigma_{i,l}) - \frac{1}{2}||y_{i,l} - H_{i,l}x_{i|k}||^2_{(R_{i,l}+\Sigma_{i,l})^{-1}}$
      $-\frac{1}{2}\mathrm{tr}\left(H_{i,l}^T (R_{i,l} + \Sigma_{i,l})^{-1} H_{i,l} P_{i|k}\right) + \log\theta_i$
    $\theta_{i,l} = \frac{\rho_{i,l}^{(2)}}{\rho_{i,l}^{(1)} + \rho_{i,l}^{(2)}}$
  end for
end for

4 Positioning quality

The problem of determining and enhancing the reliability of the positioning service in different situations is one of the fundamental problems in the art of positioning. Traditionally positioning quality, or integrity, is monitored using RAIM-techniques [14], [39]. The integrity is defined in [54] as

> a measure of trust which can be placed in the correctness of the information supplied by the total system. Integrity includes the ability of the system to provide timely warnings to the user when the system should not be used for intended operation.

Traditional RAIM is based on frequentist hypothesis testing, a theory that has been criticised as 'bad science' because of its convoluted approach (the 'null ritual') and its logical inconsistencies [10].

4.1 RAIM

RAIM is a technique for using an overdetermined system of equations to perform a check on the consistency of the system, originally used in GPS based aviation. The state contains at least four unknowns in GPS systems, i.e. three position coordinates and a receiver clock bias. Therefore, observations from at least five satellites are required for performing RAIM. Going through all the combinations of observations and finding the corresponding state estimates, the inconsistency of estimates would indicate the possibility of a faulty observation, if at most one faulty observation is assumed [39].

When six observations are observed, it is possible to identify and exclude the faulty observation, if at most a single one is present. This testing can be done for example using the reliability testing method [7]. This procedure is often referred to as RAIM with fault detection and exclusion (RAIM/FDE).

Required navigation parameters

The performance of RAIM is traditionally monitored with the required navigation performance (RNP) parameters of accuracy, availability, continuity and integrity.

Briefly, the RNP parameters are defined as follows [67]. Accuracy is the degree of conformance between the state estimates and the true state. The availability of the navigation system is the percentage of time that the services of the system are usable within the specified coverage area. Continuity is the probability that the position service is available for the duration of the phase of operation and integrity is the ability of the system to provide timely warnings to the user when the system is not to be used for positioning.

Traditionally availability and continuity are predictive parameters using the information about the orbits of the satellites.

4.2 Bayesian RAIM

Bayesian approach for RAIM is more straightforward than the traditional RAIM based on the testing of the observation consistency [53], [P2], [P3], [P5]. In previous sections we modeled the positioning problem as a dynamic estimation problem with additive sensor errors and discussed a few of the methods to approximate the posterior filtering distribution $p(x_k \mid y_{1:k})$ that contains all of the information provided by the models and the data. This posterior distribution includes all the information about the system, including the additive sensor errors, given the models and the received observations.

We have proposed a Bayesian framework for RAIM for personal positioning in urban environments [P5]. The proposed approach monitors the system performance solely based on the posterior distribution, and therefore the monitoring can be performed whenever the posterior distribution exists. There does not exist RAIM requirements for personal positioning as there are for aviation. However, the traditional RAIM is not applicable directly to the urban navigation setting, for example due to the following reasons.

- The assumption of at most one biased observation is too strict for urban navigation where multipath signals are common.
- The assumption that biased signals do not contain any useful information about the position is too strict.
- The predicted availability of RAIM is traditionally based only on the geometry and number of visible satellites, and not on the received observations. The geometry and number of visible satellites can be very poor in urban environments where large parts of the sky are blocked from view, and generally it is difficult to predict future visibility of the satellites in urban environments.
- Separate algorithms for positioning, error detection and error identification complicate the receiver architecture [54].
- Integrity is monitored indirectly through the observations and minimal biases that can be detected in the observations.

In the Bayesian framework, there are several approaches for the quality monitoring problem. Given the observation models

$$y_k = h_k(x_k) + \Lambda_k r_k$$

we can compute the probability for each $\Lambda_k^{(i)}, i = 1\ldots, 2^{n_y}$ for being the true model. Based on the probabilities of the models, we can investigate whether the observations indicate that one of the models is more probable than the others.

The posterior odds for $\Lambda_k^{(i)}$ against $\Lambda_k^{(j)}$ being the true model can be computed as

$$O_{ij} = \frac{\mathsf{P}\left(\Lambda_k^{(i)} \mid y_k\right)}{\mathsf{P}\left(\Lambda_k^{(j)} \mid y_k\right)} = \frac{p\left(y_k \mid \Lambda_k^{(i)}\right)}{p\left(y_k \mid \Lambda_k^{(j)}\right)} \frac{\mathsf{P}\left(\Lambda_k^{(i)}\right)}{\mathsf{P}\left(\Lambda_k^{(j)}\right)}, \tag{84}$$

where

$$\begin{aligned} &p\left(y_k \mid \Lambda_k\right) \\ &= \int p\left(y_k \mid x_k, \Lambda_k\right) p(x_k \mid \Lambda_k) \mathrm{d}x_k = \int p\left(y_k \mid x_k, \Lambda_k\right) p(x_k) \mathrm{d}x_k \end{aligned} \tag{85}$$

is the *evidence* given by the data y_k for the model Λ_k.

We suggested a Bayesian RAIM procedure based on the posterior odds in the case where at most a single observation channel is contaminated with a bias [P2]. The method is a snapshot RAIM procedure, i.e. it performs the integrity check at each time step with only the current set of observations, and no model for the dynamics of the error is used. Not taking the dynamics into consideration, and restricting to at most one bias within the observation vector, simplifies the problem significantly, as we have $n_y + 1$ possible models at each time step. We evaluate (84) for each of the models $\Lambda_k^{(i)}, i = 1, \ldots, n_y$ against $\Lambda_k^{(n_y+1)} = 0$ (the null model), and arrange the models according to their probability. In the case where the most probable model is not the null model, we check whether or not the correct contaminated observation could in fact be identified based on the geometry of the

problem. In the case the check fails, and the correct contaminated channel can't be identified, the system declares integrity failure.

In the cases where the geometric check does not fail or the null model has the best odds, we compute whether the odds are good enough to say a single model fits the data clearly the best. If the odds are good enough, the system declares sufficient integrity, but if the odds are not good, the system issues a warning that the integrity can't be guaranteed.

The described RAIM procedure was based solely on the probabilities of certain observations being contaminated. The main drawback of this method is that not all additive sensor errors cause the system to have performance worse than required, nor does the system with no errors necessarily perform within requirements.

Bayesian RAIM

When we have found the posterior distribution we can infer any kind of information from it. Our suggested Bayesian RAIM approach is simply based on computing the probabilities of the errors. Given the posterior, we are able to compute the probability of the true error being less than T, that can be for example the required accuracy for the current positioning task,

$$\mathsf{P}(x_k \in \Omega_T(x_{k|k}) \mid y_{1:k}) \tag{86}$$

where $\Omega_T(x_{k|k})$ contains the states within the error tolerance T

$$\Omega_T(x_{k|k}) = \{x_k : ||x_{k,1:d} - x_{k|k,1:d}|| < T\}. \tag{87}$$

The position error is computed in d-dimensions.

The probability (86) can be computed as the integral

$$
\begin{aligned}
P_T &= \mathsf{P}\left(x_k \in \Omega_T(x_{k|k}) \,|\, y_{1:k}\right) = \int_{\Omega_\mathrm{T}(x_{k|k})} p(x_k \,|\, y_{1:k}) \mathrm{d}x_k \\
&= \sum_{\Lambda_{0:k}} \mathsf{P}\left(\Lambda_{0:k} \,|\, y_{1:k}\right) \int_{\Omega_T(\hat{x}_k)} p(x_k \,|\, y_{1:k}, \Lambda_{0:k}) \mathrm{d}x_k \\
&= \sum_{\Lambda_{0:k}} \mathsf{P}\left(\Lambda_{0:k} \,|\, y_{1:k}\right) \mathsf{P}\left(x_k \in \Omega_T(x_{k|k}) \,|\, y_{1:k}, \Lambda_{0:k}\right), \qquad (88)
\end{aligned}
$$

so the probability of the error being smaller than a threshold can be obtained as the sum of the probabilities of errors within the tolerance given the indicator history multiplied by the probability of the indicator history. This approach takes automatically into account the possibility that the presence of additive sensor error does not necessarily cause the position error to be too large, and the absence of additive sensor errors does not necessarily ensure that the performance is within required limits.

In addition to the computation of accuracy, it is desirable also to compute whether the system performance is not within a specified alarm limit (AL). This probability is the integrity of the system, i.e. the RNP parameter most directly linked to the safety of the operation. Analogously to the accuracy, we can compute the integrity as

$$
\begin{aligned}
P_\mathrm{AL} &= \mathsf{P}\left(x_k \notin \Omega_\mathrm{AL}(x_{k|k}) \,|\, y_{1:k}\right) \\
&= \sum_{\Lambda_{0:k}} \mathsf{P}\left(\Lambda_{0:k} \,|\, y_{1:k}\right) \mathsf{P}\left(x_k \notin \Omega_\mathrm{AL}(x_{k|k}) \,|\, y_{1:k}, \Lambda_{0:k}\right). \qquad (89)
\end{aligned}
$$

The evaluated integrity is compared to the maximum integrity risk P_0 to decide whether or not to warn the user about a possibly too large position error. The principle of Bayesian RAIM is pictured in Figure 4.

If $P_\mathrm{AL} \leq P_0$, we do not warn the user about possibly too large error. Now the probability of misleading information (we say error is within limits and it is not) P_MI is equal to P_AL. On the other hand, if $P_\mathrm{AL} > P_0$ we warn the user about possibly too large error. The probability that the error is actually within the tolerance is the probability of false alarm and is equal to $P_\mathrm{FA} = 1 - P_\mathrm{AL}$.

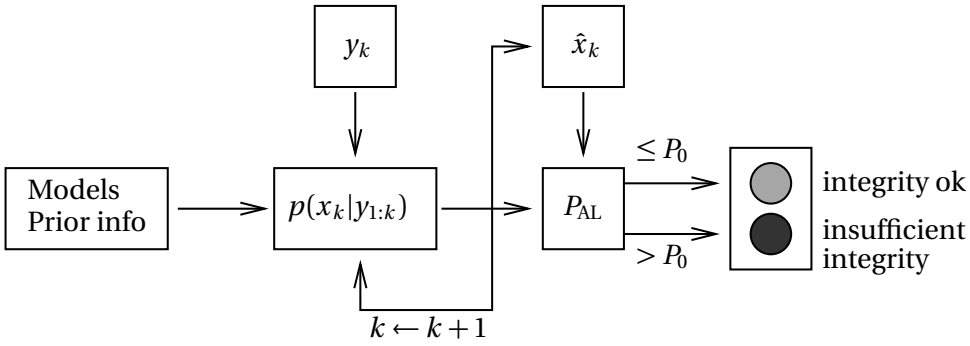

Figure 4: Diagram of the Bayesian RAIM algorithm.

Numerical integration

The computations of the integrity and accuracy involve the integration of the posterior distribution over $\Omega_{\mathrm{AL}}(x_{k|k})$, and $\Omega_{\mathrm{T}}(x_{k|k})$ respectively. Posterior distributions $p(x_k \mid y_{1:k}, \Lambda_{0:k})$ are approximated as GM. These can't be integrated analytically over general regions, but instead we have to rely on approximations [45], [57], [63], [22]. There are two approaches for approximating the integral. First, one can use numerical integration methods, i.e. quadrature or cubature rules that approximate the integral as a weighted sum of integrand evaluations at a set of nodes. Second, one can approximate the integration problem with a simpler one. We can approximate the integration region, or an integrand with something that we can analytically, or at least very accurately evaluate. In the work related to this thesis we use numerical approximations.

4.3 Failure diagnosis

In addition to RAIM methods, we can take the change detection approach for detecting abrupt changes in the observation sequence. Due to the recursive nature of the filtering algorithms, undetected sensor errors have an influence on the state estimates that may persist several time steps after its occurrence. For example, running KF algorithm without taking into account the possibility of errors

propagates the error in the mean according to Lemma 1. Proof for the lemma is not provided as it is analogous to the proof of Lemma 5 in [25].

Lemma 1. *Let the state space model be described by* (1)–(3) *where the observation equation is linear. The influence of the realized additive error sequence $s_{1:k}$ on the Kalman innovation and the posterior mean can be expressed explicitly as*

$$z_k = z_k(0_{1:k}) + \Delta z_k \tag{90}$$

and

$$x_{k|k} = x_{k|k}(0_{1:k}) + \Delta x_{k|k} \tag{91}$$

where

$$z_k(0_{1:k}) = y_k - H_k x_{k|k-1}(0_{1:k-1}) \tag{92}$$

and $x_{k|k}(0_{1:k})$ are the innovation and the mean of KF conditioned on $\Lambda_i = 0, i = 1,\ldots,k$. The sequences Δz_k and $\Delta x_{k|k}$ can be expressed recursively as

$$\Delta z_k = s_k - H_k F_{k-1} \Delta x_{k-1|k-1} \tag{93}$$

and

$$\Delta x_{k|k} = K_k s_k + C_k \Delta x_{k-1|k-1}, \tag{94}$$

where $C_k = (I_{n_x} - K_k H_k) F_{k-1}$.

The detection and the diagnosis of the constant biases in the system is often carried out using statistical tests [26],[9]. In whiteness tests such as the cumulative sum test, one computes whether or not the innovation process is a zero mean white noise process, as it is in the error-free case. In the generalized likelihood ratio test [70] one tests whether or not a constant bias appeared in the system at each of the time steps within a fixed window. Likelihood ratios of the models with the assumption of the bias appearing at the kth time step and the model without the bias are computed, and if the largest

test statistic is large enough, the bias is diagnosed by computing its maximum likelihood estimate.

In the Bayesian approach for the fault diagnosis problem we simply solve for the biases and the effect they cause using for example the estimation methods discussed in Sections 3.2 – 3.6 [P7].

Using Lemma 1 the system (41) – (43) can be transformed into the system

$$\begin{bmatrix} r_{k+1} \\ \Delta x_{k|k} \end{bmatrix} = \begin{bmatrix} \Phi_k & 0 \\ K_k \Lambda_{k+1} & C_k \end{bmatrix} \begin{bmatrix} r_k \\ \Delta x_{k-1|k-1} \end{bmatrix} + \begin{bmatrix} \epsilon_{k+1} \\ 0 \end{bmatrix} \tag{95}$$

$$z_k = \begin{bmatrix} \Lambda_k & -H_k F_{k-1} \end{bmatrix} \begin{bmatrix} r_k \\ \Delta x_{k-1|k-1} \end{bmatrix} + z_k(0_{1:k}), \tag{96}$$

$$r_0 \sim \mathsf{N}\left(r_{0|0}, P^r_{0|0}\right) \tag{97}$$

$$\Delta x_{0|0} = 0 \tag{98}$$

where $z_k(0_{1:k}) \sim N(0, S_k(0_{1:k}))$ is a white noise process independent of r_k, Λ_k and $\Delta x_{k|k}$. We can find the posterior filtering distribution for the system (95) – (97) in the Bayesian framework as

$$p(r_k, \Delta x_{k|k} \mid z_{1:k}) = \sum_{\Lambda_{0:k}} \mathsf{P}(\Lambda_{0:k} \mid z_{1:k})\, p(r_k, \Delta x_{k|k} \mid z_{1:k}, \Lambda_{0:k}), \tag{99}$$

which is a GM distribution. We have proposed a method to approximate (99) and report the estimates for $\Delta x_{k|k}$ as an addition to already implemented KF. A decision is made based on the estimate $\Delta x_{k|k}$ whether or not the KF estimate should be used for its intended purpose [P6].

5 Conclusions and future work

In this thesis we have studied the Bayesian solution of a positioning problem where observations may be contaminated with an additive sensor error. In the defined positioning problem, the state transition model is linear Gaussian, but observation equations can be linear or nonlinear. The additive sensor errors were modeled as a linear

Gaussian component multiplied by an indicator variable modeled as a Markov chain. This error model is reasonable for biases caused for example by multipath signals, but as we are describing failures, unexpected errors or biases, the models should be constructed for particular applications and systems. This is a worthy subject of study on its own that was out of scope for the current work.

In theory the problem can be solved in the Bayesian framework completely, because the posterior distribution of the state contains all the information about it given the models and the observations. However, as the solution is intractable in the general case due to exponential growth of computational requirements, we considered several techniques to approximate the posterior distribution. Many of the techniques approximate the posterior filtering distribution as a GM distribution where the Gaussian components are the distributions of the state given the observations and a particular indicator history of presence of errors in observations. The Gaussian components may vary significantly with different indicator histories and to keep the computational costs at a reasonable level, some of the components must be merged or deleted. The deleting or merging of the components may cause the approximative method to lose the information about the joint distribution of the errors and the state and therefore lead to wrong analysis about the state error. Further study on techniques that approximate GM posterior filtering distributions but are not prone to lose information about the joint distribution of the state and errors is required. Smoothing, i.e. waiting a few time steps to gather more data before merging or deleting components may improve the approximative posterior. Other possible approaches could be cost-based deleting or merging of the components so that components with low but reasonable probability will not be deleted immediately, if they describe MS motion that differ from the probable components.

Often the main criticism against the Bayesian approach is the requirement of prior distributions. In the current application we require priors for the magnitude and the presence of the sensor errors, in addition to the state prior. Naturally, the priors can have a major impact on the results, but we do not consider this being a major

issue, because information about error magnitudes can be experimentally found in any positioning system, and more importantly because the prior distributions should reflect the prior information that the user has. This means that if there is significant uncertainty about the magnitude of the errors, this should be reflected by large prior variances. The prior distributions will determine the influence of the observations at a certain time step, and therefore severe problems may arise if the magnitudes and the variances of the errors are significantly underestimated, or overestimated a priori. In the worst case, the filter performance can become extremely sensitive, or unresponsive, to additive errors, and the posterior distribution will not contain truthful information about the state.

In this thesis, we investigated a Bayesian approach to system failure diagnosis and RAIM. The Bayesian approach is attractive because it is more straightforward than traditional methods. Given that we have probabilistic models for all the system components, we can solve all the statistics that describe the system performance, at least in theory. Similar to classic GLR [70], [26], we are able to describe the effect of sensor errors on the KF and estimate it using the Bayesian approach. Based on the estimate, we can determine if the effect of sensor errors on the state estimate is negligible, or are further investigations required.

We also described a Bayesian RAIM method for urban navigation. RAIM was originally designed for aviation purposes and is not directly applicable to urban navigation. Therefore, we have investigated a more general Bayesian approach corresponding to the traditional RAIM that is applicable to any positioning problem in which we can formulate the models as probability distributions. In the simplest case, the Bayesian approach enables us to compute directly the posterior probabilities of the models describing the presence of sensor errors, and we can find the most probable error models. As the probability of the error model does not necessarily indicate anything about the error in the state, we have investigated the method for directly computing the posterior probability of the errors of certain size. In the Bayesian framework this is done by integrating the posterior distribution over a region defined by allowable error. We defined

how to evaluate the RAIM performance parameters in the Bayesian approach, but currently there do not exist requirements for the personal positioning quality, other than the emergency call positioning requirements [28].

There is a lot of future development required for the personal RAIM in urban environment. The Bayesian approach is very attractive due to the benefits discussed in this thesis and it should be further developed to be applicable in real positioning systems. The main problem of the Bayesian approach is that it is computationally quite demanding. The application of the Bayesian RAIM in handheld devices is not currently feasible, but simultaneous development of better approximative methods and more capable hardware could make the Bayesian approach applicable in the near future.

References

[1] G. Agamennoni, J. I. Nieto, and E. M. Nebot. An outlier-robust Kalman filter. In *Proceedings of 2011 IEEE International Conference on Robotics and Automation, Shanghai International Conference Center, May 9–13,2011 Shanghai, China*, 2011.

[2] S. Ali-Löytty. *Gaussian Mixture Filters in Hybrid Positioning*. PhD thesis, Tampere University of Technology, 2009.

[3] D. L. Alspach and H. W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, AC-17(4):439–448, August 1972.

[4] B. D. O. Anderson and J. B. Moore. *Optimal filtering*. Prentice Hall, Inc., 1979.

[5] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, 2009.

[6] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50 (2):174–188, February 2002.

[7] W. Baarda. *A Testing Procedure for Use in Geodetic Networks*. Netherlands Geodetic Commission, Publication on Geodesy, New Series 2, No. 5, Delft, Netherlands, 1968.

[8] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[9] M. Basseville and I. V. Nikiforov. *Detection of abrupt changes: theory and applications*. Information and system science series. Prentice Hall, NJ, 1993.

[10] J. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag New York, Inc., 2006.

[11] N. Bergman and F. Gustafsson. Three statistical batch algorithms for tracking manoeuvring targets. Technical report, Department of Electrical Engineering, Linköping University, 1999.

[12] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 2006.

[13] G. Box and G. Tiao. A Bayesian approach to some outlier problems. *Biometrika*, 55(1):119–129, 1968.

[14] R. G. Brown. A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods. *NAVIGATION : Journal of Institute of Navigation*, 39(3):101–116, 1992.

[15] *Wi-Fi location-based services 4.1 design guide*. Cisco Systems, Inc., 2008.

[16] F. Daum. Nonlinear filters: beyond the Kalman filter. *IEEE Aerospace & Electronical Systems Magazine*, 20(8):57–69, 2005.

[17] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[18] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.

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

[20] A. Doucet, N. J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):2001, March 2001.

[21] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis, Second Edititon*. Chapman & Hall/CRC Press, 2009.

[22] A. Genz and F. Bretz. *Computation of multivariate normal and t probabilities*, volume 195. Springer-Verlag Berlin Heidelberg, 2009.

[23] A. Giremus and J.-Y. Tourneret. Joint detection/estimation of multipath effects for the global positioning system. In *Proceedings of IEEE international conference on acoustics, speech and signal processing, Philadelphia, PA, March 2005*, pages 17–20, 2005.

[24] A. Giremus, J. Tourneret, and V. Calmettes. A particle filtering approach for joint detection/estimation of multipath effects on GPS measurements. *IEEE Transactions on Signal Processing*, 55 (4):1275–1285, April 2007.

[25] F. Gustafsson. The marginalized likelihood ratio test for detection abrupt changes. *IEEE Transactions on Automatic Control*, 41(1):66–78, January 1996.

[26] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons Ltd., 2000.

[27] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics : The Approach Based on Influence Functions*. John Wiley & Sons, Inc., 1986.

[28] D. N. Hatfield. A report on technical and operational issues impacting the provision of wireless enhanced 911 services. Technical report, Federal Communications Commission, 2002.

[29] G. Heinrichs, F. Dovis, M. Gianola, and P. Mulassano. Navigation and communication hybrid positioning with a common receiver architecture. In *Proceedings of the European Navigation Conference GNSS, 2004*, 2004.

[30] S. Hewitson and J. Wang. GNSS receiver autonomous integrity monitoring (RAIM) performance analysis. Technical report, The University of New South Wales, 2005.

[31] P. J. Huber. *Robust Statistics*. John Wiley & Sons, Inc., 1981.

[32] J. M. Huerta, J. Vidal, A. Giremus, and J.-Y. Tourneret. Joint particle filter and UKF position tracking in severe non-Line-of-Sight situations. *IEEE Journal of selected topics in signal processing*, 3(5):874–888, October 2009.

[33] A. H. Jazwinski. *Stochastic Processes And Filtering Theory*. Academic Press, Inc., 1970.

[34] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482, March 2000.

[35] S. J. Julier and J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Proceedings of AeroSense: the* 11*th internation symposium on aerospace/defence sensing, simulation and controls*, 1997.

[36] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, March 2004.

[37] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice Hall, NJ, 2000.

[38] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82, 1960.

[39] E. D. Kaplan, editor. *Understanding GPS : principles and applications*. Artech House, Inc., 1996.

[40] A. Kong, J. S. Liu, and W. H. Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994.

[41] L. Koski, T. Perälä, and R. Piché. Indoor positioning using WLAN coverage area estimates. In *Proceedings of 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Zurich, Switzerland, September*, 2010.

[42] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen. Positioning with coverage area estimates generated from location fingerprints. In *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2010 (WPNC'10), Dresden, Germany*, pages 99–106, 2010.

[43] B. D. Kovačević, Ž. M. Durović, and S. Glavaški. On robust Kalman filtering. *International Journal of Control*, 56(3):547–562, 1992.

[44] V. Krishnamurthy and J. B. Moore. On-line estimation of hidden Markov model parameters based on the Kullback-Leibler information measure. *IEEE Transactions on Signal Processing*, 41(8): 2557–2573, August 1993.

[45] A. Krommer and C. Ueberhuber. *Computational Integration*. SIAM, 1998.

[46] H. Kuusniemi. *User-Level Reliability and Quality Monitoring in Satellite-Based Personal Navigation*. PhD thesis, Tampere University of Technology, 2005.

[47] J. Kwon, B. Dundar, and P. Varaiya. Hybrid algorithm for indoor positioning using wireless LAN. In *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall.*, volume 7, pages 4625–4629, September 2004.

[48] C. Ma. Integration of GPS and cellular networks to improve wireless location performance. In *Proceedings of ION GPS/GNSS 2003*, pages 1585–1596, 2003.

[49] J. Marais and B. Godefroy. Analysis and optimal use of GNSS pseudo-range delays in urban canyons. In *Proceedings of IMACS Multiconference on Computational Engineering in System Applications (CESA), October 4-6, 2006, Beijing, China*, 2006.

[50] C. J. Masreliez. Approximate non-Gaussian filtering with linear state and observation relations. *IEEE Transactions on Automatic Control*, 20(1):107–110, February 1975.

[51] C. J. Masreliez and R. D. Martin. Robust Bayesian estimation for the linear model and robustifying the Kalman filter. *IEEE Transactions on Automatic Control*, 22(3):361–271, 1977.

[52] P. Misra and P. Enge. *Global Positioning System : Signals, Measurements, and Performance*. Ganga-Jamuna Press, 2006.

[53] P. B. Ober. Integrity according to Bayes. In *Position Location and Navigation Symposium, IEEE 2000, San Diego, CA, USA*, pages 325–332, 2000.

[54] P. B. Ober. *Integrity Prediction and Monitoring of Navigation Systems*. PhD thesis, Teknische Universiteit Delft, 2003.

[55] D. Peña and I. Guttman. Optimal collapsing of mixture distributions in robust recursive estimation. *Communications in Statistics: Theory and Methods*, 18(3):817–833, 1989.

[56] T. Perälä. Robust Extended Kalman filtering in hybrid positioning applications. Master's thesis, Tampere University of Technology, 2008.

[57] H. Pesonen and R. Piché. Numerical integration in Bayesian positioning. In *Proceedings of The 14th European Conference on Mathematics for Industry (ECMI 2006), Madrid, July 10-14, 2006.*, 2006.

[58] H. Pesonen and R. Piché. Cubature-based Kalman filters for positioning. In *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2010 (WPNC'10), Dresden, Germany, March 2010*, pages 45–49, 2010.

[59] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, August 1965.

[60] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle filters for tracking applications*. Artech House, Inc., 2004.

[61] C. P. Robert. *The Bayesian Choice : From Decision-Theoretic Foundations to Computational Implementation*. Springer Science+Business Media, LLC, 2007.

[62] A. R. Runnalls. Kullback-Leibler approach to Gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems*, 43(3):989–999, July 2007.

[63] N. Sirola. *Mathematical Methods for Personal Positioning and Navigation*. PhD thesis, Tampere University of Technology, 2007.

[64] H. Sorenson and D. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7:465–479, 1971.

[65] J. Soubielle, I. Fijalkow, P. Duvaut, and A. Bibaut. GPS positioning in a multipath environment. *IEEE Transactions on Signal Processing*, 50(1):141 – 150, January 2002.

[66] M. Spangenberg, J.-Y. Tourneret, V. Calmettes, and G. Duchâteau. Detection of variance changes and mean value jumps in measurement noise for multipath mitigation in urban canyons. In *Proceedings of Asilomar, 2008*, 2008.

[67] B. Tiemeyer. Evaluation of satellite navigation and safety case development. Technical report, Eurocontrol Experimental Center - report 370, 2002.

[68] N. Viandier, N. Nahimana, and D. Marais. GNSS performance enhancement in urban environment based on pseudo-range error model. In *IEEE/ION Position, Location and Navigation Symposium*, pages 377–382, 2008.

[69] M. West. Robust sequential approximate Bayesian estimation. *Journal of the Royal Statistical Society*, 43(2):157–166, 1981.

[70] A. S. Willsky and H. L. Jones. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control*, 21(1):108–121, 1976.

PUBLICATION 1

Henri Pesonen: Robust estimation techniques for GNSS positioning.
In *Proceedings of NAV07-The Navigation Conference and Exhibition,*
London, England, October 24–November 1, 2007.

NAV07-The Navigation Conference and Exhibition, 31.10.-1.11.2007, London, England

Robust Estimation Techniques for GNSS Positioning

Henri Pesonen
Department of Mathematics
Tampere University of Technology
henri.pesonen@tut.fi

Abstract

Tracking and navigation problems are often solved using estimation methods that are based on least-squares and Kalman filtering techniques. It is well known that these classic methods are sensitive to unexpectedly large measurement errors. In this article we discuss some robust static and dynamic estimation methods that are designed to be insensitive against outlying observations. Positioning simulations and results of a field test where robust techniques are applied to pedestrian positioning using GPS pseudorange measurements are presented. The results indicate that robust techniques have potential in GNSS positioning.

1 Introduction

GNSS positioning problems are often solved using estimation methods that are based on least squares estimation and Kalman filtering techniques. These methods can be shown to work optimally when the noises in the systems are Gaussian with known means and variances. The assumption of Gaussianity, even though there might be sound justification for making it, is sometimes made just because it is convenient that there exists methods that are in some sense optimal under it. The real measurement data often contain unexpectedly large errors that do not fit the assumed noise model. In GNSS measurements these kinds of errors could be the results of multipath or non-line-of-sight effects. It is well known that many of the classic methods are very sensitive to these kinds of errors, which are usually referred to as outliers, or blunder measurements.

There has been extensive study on methods that would behave as well as possible when the data is of good quality, but at the same time would be insensitive against occasional large errors. One approach to handling outliers is to try to detect them, modify the data or the model and subsequently estimate using only good data. Another approach is to compute a robust estimate using all the data and afterwards outliers could be detected as having the largest residuals. We consider only the second approach.

Methods for computing robust estimates have been considered for over 50 years. One of the most important contributions to this field is the M-estimation theory by Huber [2] [3], which is based on minimization of other loss functions than the sum of quadratic terms. M-estimation is discussed briefly in Section 2. The M-estimation theory can be used instead of the ordinary least squares in the case of static positioning.

The Kalman filter [4] and its extensions are the most used dynamic estimation methods in various problems, including GNSS navigation. Because of the popularity there is great interest to develop a robust Kalman filter-type dynamic estimation algorithms. Most of the work done in this area is heuristic by nature but can be shown to work in practice by simulations [9]. In this article we consider the one-step optimal Bayesian recursive estimator by Masreliez and Martin [7].

We test different stationary and dynamic estimators using simulated positioning scenarios in which we vary the number of available measurements and the amount of contamination. For simplicity, the provided simulations are on two-dimensional plane. We use range measurements to stationary pseudolites. The results are provided and discussed in Section 4.

In Section 5 we provide results of a test where the robust methods were applied to pedestrian positioning using GPS pseudorange measurements. In Section 6 we summarize the results of simulations and tests and provide some thoughts about future research on robust estimation methods for GNSS positioning.

2 Robust estimation

In the following two sections we consider estimation methods for static and dynamic linear estimation problems. Linear problems are considered as in this article we consider positioning with range and pseudorange measurements obtained from pseudolites or satellites located far away from the receiver. It is assumed that this results in almost linear estimation problems.

Consider the ordinary linear regression problem

$$y = Hx + v, \tag{1}$$

where y is the vector of observations, H is the design matrix of full rank and v is the observation error with variance $V(v) = R$. In the case of independent errors R is a diagonal matrix with elements σ_i. In LS estimation we minimize the quadratic cost function

$$\sum_i \rho(r_i(x)) = \frac{1}{2}\sum_i (r_i(x)/\sigma_i)^2\,, \tag{2}$$

where $r(x) = y - Hx$. The estimate that minimizes the quadratic cost function is

$$\hat{x} = (H^T R^{-1} H)^{-1} H^T R^{-1} y, \tag{3}$$

which is a linear combination of observations and as such the influence of outlying observations is not bounded in any way.

To introduce robustness into the estimation problem, Huber [2] suggested to minimize less rapidly increasing functions than (2). In more general form the estimation problem can be written as

$$\hat{x} = \arg\min_x \sum_i \rho(r_i(x)/\sigma_i). \tag{4}$$

Note that when $\rho(\cdot) = -\ln p_v$, (4) is a maximum likelihood estimation problem. This is why Huber referred his framework as M-estimation. When $\rho(\cdot)$ is a convex function, we can solve (4) from the equation

$$\sum_i \psi(r_i(x)/\sigma_i)\frac{dr_i(x)/\sigma_i}{dx} = 0. \tag{5}$$

and the solution is unique.

Although there does not exist a single best method for solving (5), one of the most popular techniques is to solve a sequence of weighted LS-problems. This is the iteratively reweighted least squares (IRLS) method which is given by Algorithm 1 [8]. It has to be noted that IRLS algorithm has only first degree convergence and needs reparameterization at each iteration, hence it is not computationally very atractive choice. Despite of this it is a much-used algorithm, possibly because of the available software for solving LS-problems. Also, with a good initial estimate, the algorithm often converges after only few iterations if the stopping criteria are not too strict.

One of the most used robust M-estimators is the one minimizing the original Huber's loss function [2]

$$\rho_H(r) = \begin{cases} k|r| - \frac{k^2}{2}, & |r| \geq k \\ \frac{r^2}{2}, & |r| < k \end{cases}. \tag{6}$$

Note that Huber's loss function corresponds to a density function that is Gaussian in the middle and double exponential in the tails. The Huber's ψ-function in Fig. 1 shows the influence of a residual to the estimate.

Another well-known loss function is the Tukey's bisquare

$$\rho_B(r) = \begin{cases} \frac{k^2}{6}, & |r| \geq k \\ \frac{k^2}{6}\left\{1 - \left[1 - \left(\frac{r}{k}\right)^2\right]^3\right\}, & |r| < k \end{cases} \tag{7}$$

which is not a convex function and as such the convergence to a global minimum is not quaranteed. Note that for residuals in the tails Tukey's loss function gives zero influence as illustrated by Figure 1. The effect is same as leaving out particular observations from the system and hence extra care has to be used when minimizing these kinds of loss functions. With a bad initial estimate the IRLS algorithm can ignore perfectly good observations and converge to a local minimum.

3 Robust dynamic estimation

In order to robustify the Kalman filter, Masreliez and Martin [7] approached the linear regression problem (1) using the Bayesian framework

$$y = Hx + v, \quad x \sim \Phi(\bar{x}, P), \quad v \sim F, \tag{8}$$

where Φ is Gaussian and F_v is symmetric and heavy-tailed. There is a restriction for the form of F_v that requires the existence of a transformation T such that the

Algorithm 1: IRLS

Set initial estimate $x^{(0)}$
Set error tolerance ϵ
$k = 0$
for $k = 1, 2, \ldots, N$ **do**
  $r^{(k-1)} = y - Hx^{(k-1)}$
  $w_i^{(k-1)} = \frac{\psi(r_i^{(k-1)}/\sigma)}{|r_i^{(k-1)}/\sigma|}$
  $W^{(k-1)} = \mathrm{diag}[w_1^{(k-1)}, \ldots, w_n^{(k-1)}]$
  $x^{(k)} = \left(H^T W^{(k-1)} H\right)^{-1} H^T W^{(k-1)} y$
  if $\|x^{(k)} - x^{(k-1)}\| < \epsilon$ **then**
    Stop
  end
end

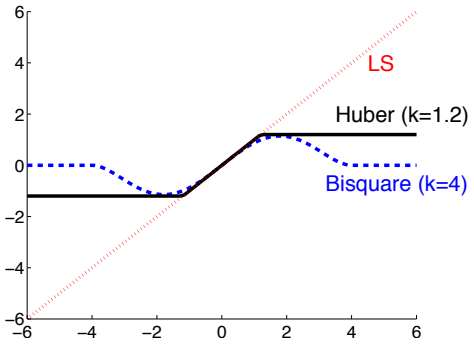

Figure 1: ψ-functions indicate the influence that a sample has on an M-estimator.

transformed innovation variable

$$u = T(H(x - \bar{x}) + v) \tag{9}$$

has a distribution F_u which has a density function that is an even function of all its parameters and all the marginal distributions F_{u_i} are members of a family of distributions $\mathcal{F}$. A transformation as such exists for example if the measurements errors are independent and have distributions from the same family of distributions.

Let F_0 be the least favorable member of the family of distributions $\mathcal{F}$ i.e. the distribution that minimizes the Fisher information for the best possible estimator θ_0 of a family of estimators Θ. Mathematically it is the saddle-point of the game

$$\min_{\theta\in\Theta} \max_{F\in\mathcal{F}} V(F, \theta) = V(F_0, \theta_0) = \max_{F\in\mathcal{F}} \min_{\theta\in\Theta} V(F, \theta), \tag{10}$$

where Θ is a family of estimators and θ_0 the min-max estimator. Masreliez and Martin [7] showed that if F_v is of required form, the error variance for an estimator

$$\hat{x} = \bar{x} + PH^T T^T \Psi(u), \tag{11}$$

where $[\Psi(u)]_i = \psi(u_i)$, is bounded from above as

$$\mathbb{E}\left[(\hat{x} - x)(\hat{x} - x)^T\right] \leq \left(I - PH^T T^T T H \mathbb{E}_{F_0}\left[\frac{\mathrm{d}\psi(u)}{\mathrm{d}u}\right]\right) P. \tag{12}$$

The ψ-function as defined in Section 2 is the derivative of the loss function corresponding to the ML-estimate of the least favorable distribution.

Consider the linear filtering problem

$$x_{k+1} = F_k x_k + \Gamma_k w_k \tag{13}$$
$$y_k = H_k x_k + v_k \tag{14}$$
$$x_0 \sim \Phi(\bar{x}_0, P_0), \tag{15}$$

where $w_k \sim \Phi(0, Q_k)$ and v_k is heavy-tailed, symmetric non-Gaussian for which the transformation T discussed previously exists. The Bayesian robust estimator for the linear model is applied to the filtering problem sequentially at each timestep k. This is carried out by heuristically approximating the posterior density with a Gaussian distribution. The robust filter is presented by Algorithm 2.

Algorithm 2: Robust Kalman-type filter

$\hat{x}_{0|0} \sim \Phi(\bar{x}_{0|0}, P_{0|0})$
for $k = 1, 2, \ldots$ **do**
  $\bar{x}_{k|k-1} = F_{k-1}\bar{x}_{k-1|k-1}$
  $P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + \Gamma_{k-1}Q_{k-1}\Gamma_{k-1}^T$
  $\bar{x}_{k|k} = \bar{x}_{k|k-1} + P_{k|k-1}H_k^T T^T \Psi(T(y_k - H_k\bar{x}_{k|k-1}))$
  $P_{k|k} = \left(I - P_{k|k-1}H_k^T T_k^T T_k H_k \mathbb{E}_{F_0}\left[\frac{\mathrm{d}\psi(u)}{\mathrm{d}u}\right]\right) P_{k|k-1}$
  Approximate $p_{\hat{x}_{k|k}} \approx \Phi(\bar{x}_{k|k}, P_{k|k})$
end

Two possible choices for family of distributions $\mathcal{F}$, as given by Masreliez and Martin [7], are the ϵ-contaminated family

$$\mathcal{F}_\epsilon = \{F \mid F = (1-\epsilon)\Phi(\cdot|0, 1) + \epsilon H, \\ H \text{ has a symmetric density}\}, \tag{16}$$

and the p-point family

$$\mathcal{F}_p = \Big\{ F \mid F(-y_p) = \Phi(-y_p|0,1) = \tfrac{p}{2}, \\ F \text{ has a symmetric density and is continuous in } \pm\tfrac{p}{2} \Big\}, \tag{17}$$

which is discussed also in Martin and Masreliez [6].

The least favorable density for ϵ-contaminated family is

$$f_{\epsilon,0}(x) = \begin{cases} \frac{1-\epsilon}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}x^2\right], & |x| < k \\ \frac{1-\epsilon}{\sqrt{2\pi}} \exp\left[-k|x| + \frac{1}{2}k^2\right], & |x| \geq k, \end{cases} \tag{18}$$

where k is a function of the fraction of contamination ϵ and can be solved from the equation

$$\frac{2}{k}\phi(k|0,1) - 2\Phi(-k|0,1) = \frac{\epsilon}{1-\epsilon}, \tag{19}$$

where ϕ is the Gaussian probability density function. The expectation needed for the upper bound of the variance is

$$\mathbb{E}_{F_{\epsilon,0}}\left[\frac{\mathrm{d}\psi(u)}{\mathrm{d}u}\right] = 1 - 2\Phi(-k|0,1). \tag{20}$$

For ϵ-contaminated family the error bound (12) is only an upper bound as it is not possible that the innovation has the least favorable distribution as its distribution.

The least favorable density for the p-point family of distributions is

$$f_{p,0}(x) = \begin{cases} K\cos^2\left(\frac{x}{2my_p}\right), & |x| \leq y_p \\ K\cos^2(\frac{1}{2m})\exp\left[\frac{2K}{p}\cos^2\left(\frac{1}{2m}\right)(y_p - |x|)\right], & |x| > y_p. \end{cases} \tag{21}$$

The normalization factor K can be solved using the property of all the distributions of the family that the distributions have the mass $1-p$ inside the interval $\left[-y_p, y_p\right]$, where the point y_p is determined by $\Phi(-y_p|0,1) = p/2$. The parameter value m minimizing the variance can be solved from

$$2m - p\left[1 + \tan^2\left(\frac{1}{2m}\right)\right]\left[2m + \tan\left(\frac{1}{2m}\right)\right] = 0. \tag{22}$$

Using the ψ-function of the p-point family of estimators

$$\psi_p(x) = \begin{cases} \frac{1}{my_p}\tan\left(\frac{x}{2my_p}\right) & , |x| \leq y_p \\ \frac{\mathrm{sign}(y_p)}{my_p}\tan\left(\frac{1}{2m}\right) & , |x| > y_p, \end{cases} \tag{23}$$

the expectation can now computed from

$$\begin{aligned} & \mathbb{E}_{F_{p,0}}\left[\frac{\mathrm{d}\psi(u)}{\mathrm{d}u}\right] \\ = \; & \frac{1}{2}\int_{-y_p}^{y_p} \frac{1}{(my_p)^2}\left(1 + \tan^2\left(\frac{x}{2my_p}\right)\right) f_{p,0}(x)\mathrm{d}x \end{aligned} \tag{24}$$

The above formulation of a robustified Kalman-type filter is just one of many. More general formulations than Algorithm 2 can be found for example in Kovačević et al. [5] and Schick and Mitter [10].

4 Simulations

We compare the different positioning methods using Matlab simulations. It is not a particularly easy task to choose the best way to compare the methods [11], as judged by different criteria several completely different filters can be the best one.

In this work we choose to look at the frequential behaviour of the estimation methods. Instead of considering the performance of the methods in one particular positioning scenario with a specific track and a set of observations, we generated 200 tracks of 120 epochs and a set of observations for each of the tracks and look at the frequential performance of the methods.

For simplicity we consider positioning with range measurements from a set of stationary pseudolites located far away at coordinates $s_1, \ldots, s_l$.

To generate the tracks we used the piecewise constant white noise acceleration model [1].

$$x_{k+1} = F_k x_k + \Gamma_k w_k, \tag{25}$$

$$\Gamma_k = \begin{bmatrix} \frac{1}{2}I_{2\times 2} \\ I_{2\times 2} \end{bmatrix}, \tag{26}$$

with

$$w_k \sim N\left(0, 0.2^2 I_{2\times 2}\right) \tag{27}$$

$$F = \begin{bmatrix} I_{2\times 2} & I_{2\times 2} \\ 0 & I_{2\times 2} \end{bmatrix}. \tag{28}$$

The state x_k contains the position and velocity coordinates.

Measurements were generated with a model

$$y_k = \begin{bmatrix} \|x_k - s_1\| \\ \vdots \\ \|x_k - s_l\| \end{bmatrix} + v_k, \tag{29}$$

where the observation noise

$$v_k \sim (1-\varepsilon)N(0, 10^2 I_{l\times l}) + \varepsilon N(0, 50^2 I_{l\times l}). \tag{30}$$

We generated several positioning scenarios by varying the fraction of contamination ε as well as the number of measurements.

In our tests we compared three static estimation methods. The performance of M-estimates with Huber's and Tukey's bisquare loss functions were compared against ordinary least squares solution. The tuning parameters for the M-estimators were chosen as $k = 1.345$ for Huber and $k = 4.5$ for bisquare.

Also three dynamic estimation methods were tested. Two robust filters using the assumptions of measurement errors $F \in \mathcal{F}_\epsilon$ and $\in \mathcal{F}_p$ (REKF$_\epsilon$ and REKF$_p$) were compared against the performance of the ordinary EKF. The parameter values were chosen as $\epsilon = 0.1$ and $p = 0.5$.

The results of simulations are given in Figures 2 - 3, where the bounds that contain 33%, 67% and 90% of the errors are plotted. Simulations show how extremely sensitive the classic methods are to even small amount of contaminated measurements. Robust filters offer good protection against unmodeled errors with only about four times more computation time than that of the Kalman filter. The M-estimation methods can give good results even in very contaminated scenarios but with the lack of prior information the static robust methods need more observations than the dynamic methods to be as robust as dynamic estimation methods. The robust filters perform almost identically on average as do two M-estimators as far as the estimation error is considered.

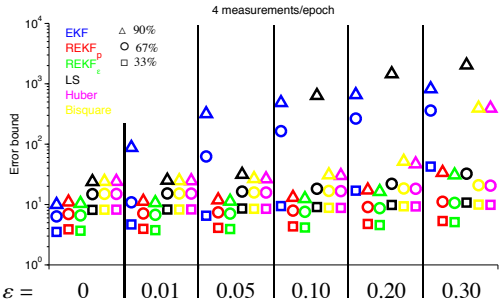

Figure 2: Error bounds containing 33%, 67% and 90% of the errors when contamination fraction is ε.

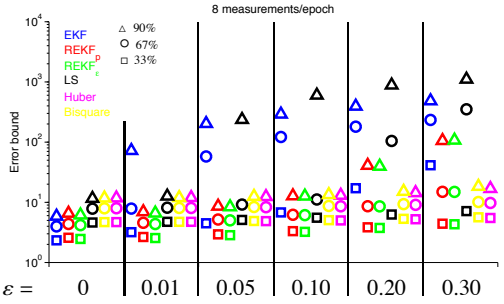

Figure 3: Error bounds containing 33%, 67% and 90% of the errors when contamination fraction is ε.

5 Tests

We tested the methods with real GPS-data. The pseudorange measurements were collected from a 380 second long walk around the campus area of Tampere University of Technology using the Holux GPSlim236 Wireless Bluetooth GPS Receiver. Now it has to be noted that unlike in simulations, we do not know the exact true track but instead compute the distance to a reference track that is assumed to be very close to the true track. Unfortunately, there is some error also in the reference track which in our tests was the postprocessed DGPS solution with data collected by two NovAtel's DL-4 GPS receivers. As bad signal environments would also affect the DGPS positioning, we obstruct the line-of-sight of the GPS receiver by covering it. This should results in noisier pseudorange measurements.

The state that we are solving now consists of position and velocity coordinates in ECEF coordinates and clock bias and clock drift in meters. The static estimation methods solve only for position and clock bias. The model

used for pseudorange measurement is

$$\rho_i = \|s_i - x_{1:3}\| + (b - b_i)c + I_i + T_i + v_i, \qquad (31)$$

where we assume that satellite clock bias b_i, the delays I_i, T_i correponding to ionophere and troposphere can be solved accurately. Also, for simplicity the model assumes independent errors $v_i \sim N\left(0, \left(1500/\sqrt{C/N_0}\right)^2\right)$, where C/N_0 is the carrier-to-noise ratio of the ith measurent.

The filters use the constant velocity and clock drift motion model

$$x_{k+1} = F_k x_k + w_k \qquad (32)$$

$$w_k \sim N\left(0, \begin{bmatrix} \frac{1}{3}\Delta t^3 Q_c & \frac{1}{2}\Delta t^2 Q_c & 0 \\ \frac{1}{2}\Delta t^2 Q_c & \Delta t Q_c & 0 \\ 0 & 0 & Q_{\text{clock}} \end{bmatrix}\right) \qquad (33)$$

where Δt is the length of the time interval between successive epochs, $Q_c = \text{diag}(0.1, 0.1, .01)$ in ENU coordinates and

$$Q_{\text{clock}} = \begin{bmatrix} 1/3\Delta t^3\sigma_c^2 & 1/2\Delta t^2\sigma_c^2 \\ 1/2\Delta t^2\sigma_c^2 & \Delta t\sigma_c^2 \end{bmatrix}, \sigma_c^2 = 2. \qquad (34)$$

M-estimators with Huber and bisquare loss functions use parameters $k = 0.5$ and $k = 2.5$ correspondingly. The robust Kalman-type filters use parameter values $p = 5$ and $\epsilon = 0.2$

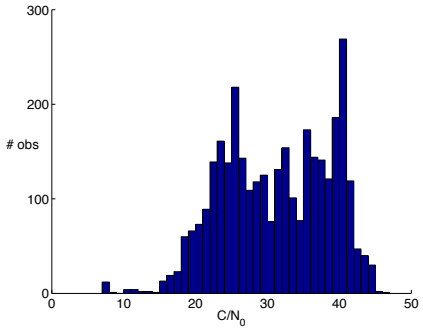

Figure 4: The distribution of measured carrier-to-noises.

From the collected pseudorange measurement data we selected different combinations of measurements which we used to compute the solutions. Different measurement sets were selected based on the carrier-to-noise ratios of the measurements. The results are collected in the Tables 2-5 in the form of the percentages of time when the estimate is within some bound from the reference position. The carrier-to-noise intervals and corresponding average number of available satellites and the percentage of epochs when only less than four measurements were available is collected in Table 1.

C/N_0	ave(SV)	% of epochs when # SV < 4
$(0, \infty)$	8.7	0
$[20, \infty)$	8.2	0
$[25, \infty)$	6.6	0.6
$[30, \infty)$	4.8	22
$(0, 40]$	6.9	0
$[20, 40]$	6.4	0
$[30, 40]$	2.9	58
$(0, 35]$	5.2	0.7
$(0, 30]$	3.7	50

Table 1: 0.33 CERP radius [m] when observations with carrier-to-noise ratios within certain intervals are used.

All the static estimation methods give very similar results but robust filters give often little better results than the Kalman filter, but the difference is not very large. It is debatable if the accuracy gained is worth the extra computation time. Our Matlab implementations of the robust filters required about four times more computation time than our implementation of the EKF.

C/N_0	EKF	REKF$_p$	REKF$_\epsilon$	LS	Huber	Bisquare
$(0, \infty)$	30.8	28.7	27.7	24.4	24.4	24.4
$[20, \infty)$	31.4	30.2	30.2	25.9	25.9	25.9
$[25, \infty)$	30.2	39.3	39.3	23.5	23.5	23.5
$[30, \infty)$	8.5	23.2	19.5	13.7	13.7	13.7
$(0, 40]$	28.4	32.3	32.9	13.4	13.4	13.4
$[20, 40]$	28.7	32.6	33.2	13.1	13.1	13.1
$[30, 40]$	16.8	7.6	17.1	3.7	3.7	3.7
$(0, 35]$	4.3	8.2	6.7	1.2	1.2	1.2
$(0, 30]$	1.2	0.9	1.2	0.3	0.3	0.3

Table 2: How often (%) the estimate is within 5m of the reference track.

6 Conclusions

In this article we reviewed robust methods that can be used in positioning applications. Both static and dynamic

C/N_0	EKF	REKF$_p$	REKF$_\epsilon$	LS	Huber	Bisquare
$(0,\infty)$	51.8	60.1	63.7	53.0	53.0	53.0
$[20,\infty)$	48.5	62.2	64.3	55.2	55.2	55.2
$[25,\infty)$	57.3	70.4	73.2	52.7	52.7	52.7
$[30,\infty)$	55.5	61.6	55.2	32.3	32.3	32.3
$(0,40]$	46.3	51.5	51.8	29.0	29.0	29.0
$[20,40]$	44.8	49.7	50.9	31.1	31.1	31.1
$[30,40]$	39.6	48.2	54.6	8.8	8.8	8.8
$(0,35]$	20.7	24.7	25.9	4.9	4.9	4.9
$(0,30]$	6.1	7.3	6.4	0.6	0.6	0.6

Table 3: How often (%) the estimate is within 10m of the reference track.

C/N_0	EKF	REKF$_p$	REKF$_\epsilon$	LS	Huber	Bisquare
$(0,\infty)$	79.6	80.8	81.7	71.6	71.6	71.6
$[20,\infty)$	74.7	80.8	82.9	72.6	72.6	72.6
$[25,\infty)$	83.5	94.5	93.3	70.7	70.7	70.7
$[30,\infty)$	80.8	93.0	92.7	46.3	46.3	46.3
$(0,40]$	75.3	76.5	77.7	47.9	47.9	47.9
$[20,40]$	67.1	75.0	76.2	50.0	50.0	50.0
$[30,40]$	61.3	64.0	73.5	13.7	13.7	13.7
$(0,35]$	55.5	51.2	52.7	15.2	15.2	15.2
$(0,30]$	13.4	18.6	15.2	0.9	0.9	0.9

Table 4: How often (%) the estimate is within 15m of the reference track.

C/N_0	EKF	REKF$_p$	REKF$_\epsilon$	LS	Huber	Bisquare
$(0,\infty)$	0	0	0	3.0	1.8	1.8
$[20,\infty)$	0	0	0	2.7	2.7	2.7
$[25,\infty)$	0	0	0	3.4	3.4	3.4
$[30,\infty)$	0	0	0	6.1	6.1	6.1
$(0,40]$	0	0	0	11.9	10.1	10.1
$[20,40]$	0	0.6	0	12.2	12.2	12.2
$[30,40]$	11.9	10.7	11.3	12.5	12.5	12.5
$[0,35]$	0	0	0	45.4	45.1	45.1
$(0,30]$	0	1.2	0.3	41.5	41.5	41.5

Table 5: How often (%) the estimate is **not** within 50m of the reference track.

cases were considered. In simulations it could be shown that the robust methods perform clearly better than the classic methods even with small amount of bad observations. This is because the simplified nature of the simulations, i.e. measurement noise in independent and outliers occur independently with a certain probability.

We tested the methods also with typical GPS pseudorange measurements in pedestrian positioning application. We used a DGPS solution as a reference track and collected the test data with consumer priced GPS receiver which was covered to obtain noisier measurements. Robust Kalman-type filters seem to give almost always better results than the Kalman filter but the difference between methods is not as distinctive as in simulations. This might be because of several reasons. First, all the robust methods discussed in this article assume zero mean noise but in reality pseudorange measurements contain interference that can not be modeled exactly but instead cause the noise to be biased. Second, the outliers are not independent but tend to occur during successive epochs e.g. while we are in an urban canyon and are not spread evenly in time.

For future work tests with extremely noisy data should be carried out. In that kind of environment, a good reference track would be very hard to obtain but easiest solution would be to evaluate the results graphically. To handle patchy outliers, dynamic estimators that are based on model selection could be applied. Also it would be interesting to apply robust methods to sensor-aided GNSS positioning and, in general to hybrid positioning.

References

[1] Yaakov Bar-Shalom, X. Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[2] Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35 (1), March 1964.

[3] Peter J. Huber. *Robust Statistics*. John Wiley & Sons, Inc., 1981.

[4] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82, 1960.

[5] Branko D. Kovačević, Željko M. Durović, and Sonja Glavaški. On robust Kalman filtering. *International Journal of Control*, 56(3):547–562, 1992.

[6] R. Douglas Martin and C. J. Masreliez. Robust estimation via stochastic approximation. *IEEE Transactions on Information Theory*, 21(3):263–271, May 1975.

[7] C. Johan Masreliez and R. Douglas Martin. Robust Bayesian estimation for the linear model and robustifying the Kalman filter. *IEEE Transactions on Automatic Control*, 22(3), 1977.

[8] M. R. Osborne. *Finite algorithms in optimization and data analysis*. John Wiley & Sons Ltd., 1985.

[9] Tommi Perälä and Robert Piché. Robust Extended Kalman filtering in hybrid positioning applications. In *Proceedings of the 4th Workshop on Positioning, Navigation and Communication (WPNC'07), March 22*, pages 55–64, 2007.

[10] Irvin C. Schick and Sanjoy K. Mitter. Robust estimation in the presence of heavy-tailed observation noise. *The Annals of Statistics*, 22(2):1045–1080, June 1994.

[11] Niilo Sirola, Simo Ali-Löytty, and Robert Piché. Benchmarking nonlinear filters. In *Nonlinear Statistical Signal Processing Workshop NSSPW06*, 2006.

PUBLICATION 2

Henri Pesonen and Robert Piché: Bayesian receiver autonomous integrity monitoring technique. In *Proceedings of the Institute of Navigation International Technical Meeting 2009,* January 26–28, Anaheim, CA, 2009.

Proc. of Institute of Navigation International Technical Meeting 2009, Anaheim, CA
Pages 420-425

Bayesian Receiver Autonomous Integrity Monitoring Technique

Henri Pesonen, *Tampere University of Technology*
Robert Piché, *Tampere University of Technology*

BIOGRAPHY

Henri Pesonen received his M.Sc. degree from Tampere University of Technology in 2006 and currently is pursuing his PhD studies. His research interests are robust and reliable positioning methods and Bayesian statistical methods.

Robert Piché is professor of mathematics at Tampere University of Technology. He has a Ph.D. in civil engineering from the University of Waterloo (Canada). His scientific interests include mathematical modelling and scientific computing with applications in navigation, finance, and mechatronics.

ABSTRACT

An integrity monitoring/failure detection and identification approach for GNSS positioning that is based on Bayesian model comparison theory is introduced. In the new method the user defines models for no-failure/failure cases and the most plausible model is chosen and used to estimate position. If a channel is contaminated and the corresponding model is chosen then the effect of this channel on the position estimate is attenuated. The posterior probability odds of two models can be used as a measure of how well the models can be distinguished from each other. In the proposed RAIM-technique if none of the model plausibilities stands out from the others, the user is made aware of the situation as the case might be that the effect of a good channel is attenuated and the contaminated one is modeled as a good one. The performances of traditional RAIM/FDE and the new method are compared via simulations. Results of a test with real GPS data are also presented.

INTRODUCTION

Quality monitoring and control techniques are important parts of any position estimation algorithm. As a result, receiver autonomous integrity monitoring (RAIM) has become a basic part of personal positioning receiver architectures [3,8,10]. Integrity of a positioning system refers to the ability of the system to warn the user when a given position estimate cannot be trusted. Autonomous means that the integrity monitoring is carried out using only the signals received by the system. Furthermore RAIM techniques have been enhanced to provide not only valuable information on the quality of the position estimate but also to offer means for detecting satellite failures and enable the exclusion of blunder observations.

Traditional RAIM methods are based on conventional frequentist hypothesis testing, a theory that has been criticised for its convoluted approach and for logical inconsistencies [2]. In frequentist hypothesis testing, one seeks to reject the null hypothesis based on the improbability of the data given that the null hypothesis is true. But often what we are really interested in is whether one hypothesis is better that the other given the data.

Bayesian model comparison allows us to think in this more direct fashion: we compare the probabilities of a model being true given the data and select the model that best describes the data. Bayesian techniques have been used in integrity monitoring by Ober [10] who introduced mixture error models which lead to exact position-domain results in addition to performing data-based integrity monitoring. However, the method introduced relies on improper prior probability densities which should not be used in the particular case of mixture estimation.

We propose to use Bayesian model comparison as an autonomous integrity monitoring/fault detection technique. We refer to it as BRAIM in the rest of this article. The main advantage of the new proposed method is the natural interpretation of the results which appear as odds or probabilities of an assumption being true. Also, the algorithm is computationally light.

We compare the performance of the proposed method to the reliability testing method by [1], which has been often applied to RAIM [5,9]. The technique was designed to be used as a statistical reliability testing procedure in geodetic networks but can be used also in positioning to detect and exclude a failure among the observations. The method performs two tests. First, a global test is carried out to detect a failure by a RAIM method known as least squares RAIM. Second, if the global test detected a failure, a local test is used to identify the faulty

observation, after which it can be excluded from the measurement set. Hence the method is sometimes referred to as RAIM/failure detection and exclusion (RAIM/FDE) [8] and we adopt this acronym in this article.

In this article we first introduce briefly the concept of Bayesian model comparison problem, after which we describe Bayesian model comparison-based BRAIM method. We compare the performance of RAIM/FDE and BRAIM using simulations and a test with GPS data and present conclusions.

BAYESIAN MODEL COMPARISON

This section summarizes general Bayesian model comparison theory, see for example [11] for details. Suppose that we have models M_i all of which we consider to be reasonable for the problem we are interested in. Note that we don't necessarily believe that any of the models is the truth. The goal is to choose the most plausible model given the data. We assume that the problem is not new to us so that using our knowledge of the underlying situation, we can assign prior probabilities for the models $P(M_0),\ldots,P(M_n)$. The posterior probability of a model M_i being the model that produced data D is

$$P(M_i \mid D) = \frac{P(D \mid M_i)P(M_i)}{P(D)} \tag{1}$$

which we use to compute the posterior ratio of two models

$$O_{ij} = \frac{P(M_i \mid D)}{P(M_j \mid D)} = \underbrace{\frac{P(D \mid M_i)}{P(D \mid M_j)}}_{B_{ij}} \times \underbrace{\frac{P(M_i)}{P(M_j)}}_{P_{ij}} \tag{2}$$

The factor P_{ij} is the prior odds ratio of M_i to M_j. This a priori information represents our personal opinion about the relative plausibility of the models given the background information. Often the prior probabilities for two models are taken to be equal (P_{ij} = 1), representing the case where we don't favor one model over another, but this is not necessary. The second factor B_{ij}, called the *Bayes factor* represents the evidence in favor of M_i as opposed to M_j [7]. The evidence for model M_i is

$$P(D \mid M_i) = \int p(D \mid \theta_i, M_i)\, p(\theta_i \mid M_i)\mathrm{d}\theta_i \tag{3}$$

where θ_i is a vector of unknown parameters in the model M_i. The prior probability densities $p(\theta_i \mid M_i)$ are needed to compute the evidence. This sometimes could cause a problem as this information may not be available. On the other hand in many problems some a priori knowledge is available, for example in dynamic problems where models for the evolution of θ_i are readily available. Prior probabilities are a powerful tool for incorporating that information into the model. The posterior odds ratios are used to make decisions. The choice of a meaningful scale depends on the area of application. Jeffreys [6] suggests the following scale for general scientific investigations

O_{ij}	$\log_{10}O_{ij}$	Probability for M_i against M_j
[1,3.2)	[0,0.5)	Not worth more than a mention
[3.2,10)	[0.5,1)	Substantial
[10,31.6)	[1,1.5)	Strong
[31.6,100	[1.5, 2)	Very strong
[100,∞)	[2,∞)	Decisive

Table1. Scales for odds of probability for M_i as suggested by Jeffreys [6].

BAYESIAN INTERGRITY MONITORING TECHNIQUE

In this section we apply the Bayesian model comparison theory described in the previous section to develop an integrity monitoring/failure detection identification technique for GNSS positioning. For the sake of simplicity and possibility of analytical formulations we

$$\begin{aligned} M_0 &: y = H_0 x_0 + v \\ M_i &: y = H_0 x_0 + v + b_i e_i \\ &= \underbrace{\begin{pmatrix} H_0 & e_i \end{pmatrix}}_{H_i} \underbrace{\begin{pmatrix} x_0 \\ b_i \end{pmatrix}}_{x_i} + v, \quad i = 1,\ldots,n \end{aligned} \tag{4}$$

where e_i is the i^{th} column of $n \times n$ identity matrix, x_0 is the parameter of m state variables (position, velocity, etc.) and b_i is the bias. Model M_0 corresponds to the situation of no failure component in any of the measurements and in each model M_i the i^{th} measurement has an unknown bias b_i which is taken to be independent of x_0. In general form the measurement equation under the model M_i is

$$y = H_i x_i + v \tag{5}$$

If the prior of the parameter x_i is normal with mean μ_i and covariance P_i and the measurement error has a normal distribution with mean 0 and covariance R, we can write the evidence as

$$\begin{aligned} P(y \mid M_i) &= \int p(y \mid x_i, M_i)\, p(x_i \mid M_i)\mathrm{d}x_i \\ &= c_i^* \exp(g_i^*(z_i)) \end{aligned} \tag{6}$$

where

$$z_i = y - H_{i\ i} \tag{7}$$

$$c_i^* = \frac{\sqrt{\det(2\pi(A_i^T \Sigma_i^{-1} A_i)^{-1})}}{\sqrt{\det(2\pi\Sigma_i)}} \tag{8}$$

$$A_i = \begin{pmatrix} H_i \\ I \end{pmatrix} \tag{9}$$

$$\Sigma_i = \begin{pmatrix} R & 0 \\ 0 & P_i \end{pmatrix} = \begin{pmatrix} R & 0 & 0 \\ 0 & P_{x_0} & 0 \\ 0 & 0 & \sigma_b^2 \end{pmatrix} \tag{10}$$

$$g_i^*(z_i) = -\frac{1}{2} z_i^T (H_i P_i H_i^T + R)^{-1} z_i \tag{11}$$

Introducing $S=HP_{x0}H^T+R$, the constant c_i^* can be written as

$$c_i^* = \sqrt{\frac{\det\left(H_0^T R^{-1} H_0 + P_{x_0}^{-1}\right)}{(2\pi)^n \det(R)\det(P_{x_0})}} \times \underbrace{\begin{cases} 1 & ,i=0 \\ (\sigma_b^2 e_i^T S^{-1} e_i + 1)^{-1/2} & ,i \neq 0 \end{cases}}_{c_i}$$

and $g_i^*(\cdot)$ can be expressed as

$$g_i^*(z_i) = -\frac{1}{2} z_i^T S^{-1} z_i + \underbrace{\frac{1}{2} \begin{cases} 0 & ,i=0 \\ \dfrac{(z_i^T S^{-1} e_i)^2}{e_i^T S^{-1} e_i + \sigma_b^{-2}} & ,i \neq 0 \end{cases}}_{g_i(z_i)}$$

Using the above notation, the Bayes factor for the i^{th} model can be computed as

$$B_{ij} = \frac{c_i}{c_j} \times \exp(g_i(z_i) - g_j(z_j)) \tag{12}$$

As an example, compared to the null model, the Bayes factor for i^{th} model is

$$B_{i0} = c_i \exp(g_i(z_i)) \tag{13}$$

when the mean of the bias b_i is zero.

To compute the posterior odds ratio O_{ij} we still need to model the prior odds ratio P_{ij}. We assume that a measurement from a particular channel is contaminated with probability ε and clean with probability 1- ε and the quality of one channel is independent of another. The models that we have constructed in this section correspond to ones with 'no bad channels' and 'exactly one bad channel'. In our model, different channels are contaminated with the same probability so that all the ratios $P_{ij}=1$, $i,j > 0$. Prior odds ratios P_{i0} can be computed as

$$P_{i0} = \frac{P(n-1 \text{ channels are clean and 1 is contaminated})}{P(n \text{ channels are clean})}$$

$$= \frac{P(\text{channel is clean})^{n-1} P(\text{channel is contaminated})}{P(\text{channel is clean})^n}$$

$$= \frac{(1-\varepsilon)^{n-1}\varepsilon}{(1-\varepsilon)^n} = \frac{\varepsilon}{1-\varepsilon}$$

and the posterior odds ratio O_{i0} can be expressed

$$O_{i0} = \frac{\varepsilon c_i g_i(z_i)}{1-\varepsilon}$$
$$O_{ij} = \frac{c_i}{c_j} \times \exp(g_i(z_i) - g_j(z_j)) \quad j \neq 0 \tag{14}$$

The most plausible model can be found by comparing posterior odds, as for most plausible model M: $O_{ij} \geq 1$, for all j.

We can analyze further the properties of the posterior odds O_{ij}. First of all, the maximum odds for $O_{0i} = O_{i0}^{-1}$, for all i is achieved when $y = H_0 y_0$. Thus

$$O_{i0} \leq \frac{1-\varepsilon}{\varepsilon} \sqrt{\sigma_b^2 e_i^T S^{-1} e_i + 1} \tag{15}$$

Let $S_{ii}^{-1} = e_i^T S^{-1} e_i$ and $z = y - H_i\mu_i = y - H_0\mu_0$, and let $T < 1$ be a threshold parameter. Then

$$O_{i0} \leq T$$

$$g_i(z) \leq \ln\left(\frac{(1-\varepsilon)T}{\varepsilon c_i}\right) \tag{16}$$

$$\left|z^T S^{-1} e_i\right| \leq \sqrt{\ln\left(\frac{(1-\varepsilon)T}{\varepsilon}\sqrt{\sigma_b^2 S_{ii}^{-1} + 1}\right)\left(S_{ii}^{-1} + \sigma_b^{-2}\right)}$$

If inequality (16) holds for all i then M_0 is the most plausible model and the odds for it against any other model are at least $1/T$.

For simplicity assume that μ_0 is close to the actual unknown x_0. Then given that M_0 is the most plausible mode, the size of a bias Δ in the k^{th} measurement is bounded as

$$\left|\Delta\right| \leq \frac{1}{S_{ki}^{-1}} \sqrt{\ln\left(\frac{(1-\varepsilon)T}{\varepsilon}\sqrt{\sigma_b^2 S_{ii}^{-1} + 1}\right)\left(S_{ii}^{-1} + \sigma_b^{-2}\right)}, \ \forall i.$$

A larger bias in k^{th} observation causes one of the i odds $O_{i0} > 1$. Because of this O_{i0} can be used to detect whether there is a blunder observation among the observation set and the odds O_{i0} are a sensible measure of quality of the null model in a practical sense.

Similar analysis can be carried for all the models. We focus on the posterior odds of a correct model that is the posterior odds for model O_{ij} when there is a bias component in the i^{th} measurement. From the equation (14) we see that the odds depend on bias as

$$g_i(\Delta e_i) - g_j(\Delta e_i) = \Delta^2 \left[\frac{\left(S_{ii}^{-1}\right)^2}{S_{ii}^{-1} + \sigma_b^{-2}} - \frac{\left(S_{ij}^{-1}\right)^2}{S_{jj}^{-1} + \sigma_b^{-2}}\right] \tag{17}$$

We want that a larger bias in i^{th} observation would cause O_{ij} to be larger, however from (17) using the fact that S^{-1}

is a symmetric positive definite matrix it can be shown that this can be guaranteed only if

$$S_{ii}^{-1} > S_{ij}^{-1}, \forall j \qquad (18)$$

If this holds, we can identify a blunder in i^{th} if it is large enough. And because the odds for the correct model increase quadratically with the size of the realized bias element, the odds are a sensible measure of the correctness of the model choice.

We assume that integrity will be attained if a model corresponding to a contaminated channel is selected and that the effect of a contaminated measurement is thereby attenuated. The decision of integrity is therefore based solely on the model space and not on the resulting positioning space. As a result the BRAIM method is based on posterior odds ratios that one model stands out as the best model. The test is declared inconclusive if none of the models stands out. The test is a failure if (18) does not hold and M_0 is not the most plausible model because in this case there is no guarantee that the most plausible model handles the correct observation as a blunder. Otherwise the system is assumed to be working within prescribed standards. The threshold T for posterior odds for different situations can be based on Table 1. The BRAIM algorithm is illustrated in Figure 1.

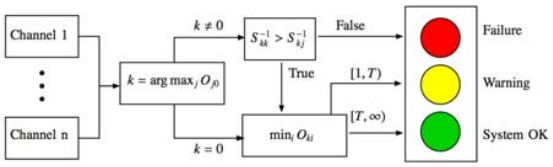

Fig.1. Diagram of the proposed BRAIM method.

Once a model is selected, the information about the state is contained in the normal posterior distribution $p(x_i \mid y, M_i) = N(m_i, C_i)$, where

$$m_i = {}_i + P_i H_i^T (H_i P_i H_i^T + R)^{-1} (y - H_i {}_i)$$
$$C_i = P_i - P_i H_i^T (H_i P_i H_i^T + R)^{-1} H_i P_i$$

In the case of models M_i, $i > 0$, the state vector x_i contains the bias element in addition to other state variables. This means that it will be estimated along with other parameters.

Note that in the case of warning or failure the resulting model does not necessarily result in particularly bad position estimate. It is important to note that the only conclusion that can be drawn is that none of the compared models stand out as the best one given the data. Instead of issuing a warning or failure message, one could proceed to further data analysis, expanding the set of models until one model does stand out. This expansion could be for example models with more than one contaminated channel but this is left for future study.

TESTS

The performance of the new proposed method is now compared to that of the classic method of RAIM/FDE as it is discussed for example in [8] in a special case of only one possible outlying observation at a time. Positioning scenarios with various numbers (n) of satellites and sets of measurements with different noise variances are generated. We generate all the observation noises from $N(0,10^2)$ for the duration of 10 epochs and after that one randomly selected satellite generates contaminated observations for the next 10 epochs. The contaminated observation noise has distribution $N(0,\sigma_c^2)$ (Table 2)

Test	n	σ_c^2
A_1,A_2	5	100^2,200^2
B_1,B_2	6	100^2,200^2
C_1,C_2	7	100^2,200^2

Table2. Test parameters.

The track of the target was generated using a constant velocity model [4] using $\sigma_c^2 = 0.01$ with an initial state $(0,0,0,1,0,0)^T$. The satellites were generated uniformly on a rectangle $[-10^5,10^5]\times[-10^5,10^5]\times[10^5,10^5+10^2]$.

The prior probability distribution for x_0, which contains position and velocity were propagated using two different motion models from the posterior probability distribution $p(x_0 \mid M_k, y)$ obtained in previous epoch. The model M_k refers to the most plausible model in that epoch. The prior probability for b_i is always taken to be independent of position and velocity and distributed as $N(0,\sigma_b^2)$. The motion models can be written as

$$x_0^{k+1} = \begin{pmatrix} I & I \\ 0 & I \end{pmatrix} x_0^k + Q_j, \; j = 1, 2$$

where $Q_1=100^2\mathrm{I}$ is large in the sense that it results in a prior that influences the results very little and $Q_2=I$ is smaller so that the resulting prior does have an influence.

The parameters of the methods α, β (probabilities of Type I and II errors in RAIM/FDE), ε and σ_b^2 are varied and the performance is reported as the fraction of epochs in which correct faulty channel was identified vs. the fraction of epochs in which no good channels were identified as faulty.

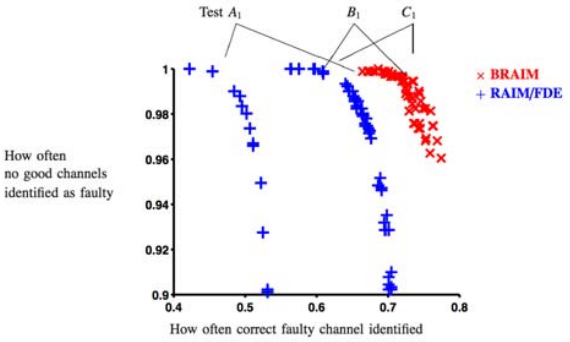

Fig.2. Method performance, more informative prior and smaller observation noise.

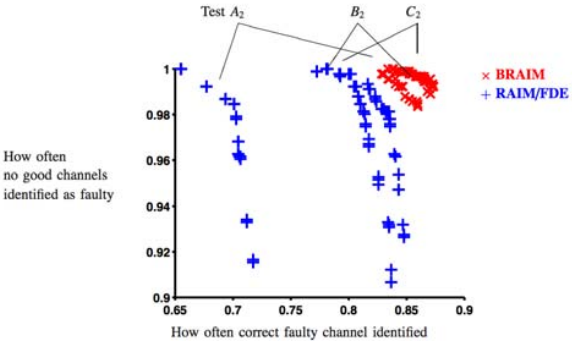

Fig.3. Method performance, more informative prior and larger observation noise.

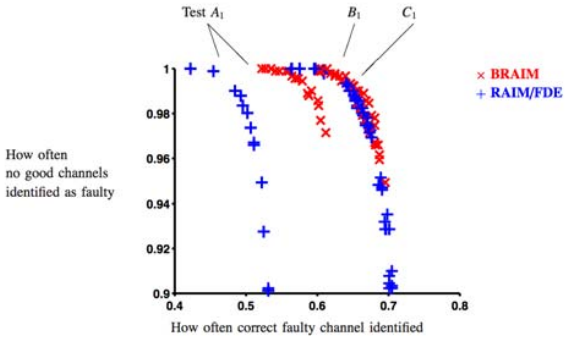

Fig.4. Method performance, less informative prior and smaller observation noise.

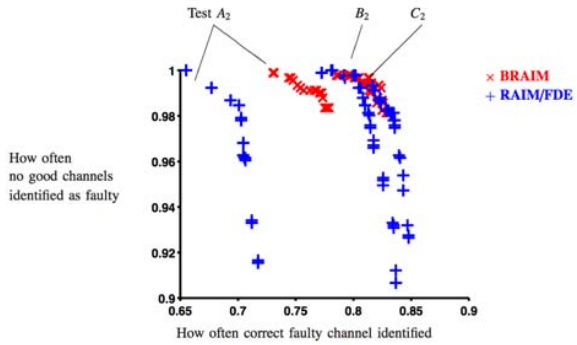

Fig.5. Method performance, less informative prior and larger observation noise.

The results of the simulations are given in Figures 2 - 5 which correspond to different scenarios. Figures 4 and 5 indicate that if prior information for BRAIM is not taken advantage of, the methods have similar performance when there are 6 or more satellites. On the other hand if use of prior information is made, then the BRAIM can perform significantly better than the traditional method, as can be seen from Figures 2 and 3.

	System OK (%)	**Warning (%)**	**Failure (%)**
Correct decision	74	26	0
Wrong decision	33	67	0

Table3. Test C_1 results with T=10, $\sigma_b^2=80^2$, ε=0.6 (small-variance prior for parameters).

	System OK (%)	**Warning (%)**	**Failure (%)**
Correct decision	69	31	0
Wrong decision	33	67	0

Table4. Test C_1 results with T=10, $\sigma_b^2=80^2$, ε=0.6 (large-variance prior for parameters).

The rates of BRAIM algorithm issuing *system OK* and *warning* flags are given in Tables 3 and 4 in the cases where correct or wrong identification were made. The results show that when correct model is chosen, the system is most often recognized to be working properly and almost no false *warning* flags are given. When wrong model is chosen, system most often issues a *warning* in these particular tests with reported parameters.

The new method was also applied to a real GPS-data test drive in Tampere. The 800 epochs long test route was in an urban area with a relatively clear view of the sky. The test was carried out by including data from at most one satellite with a poor carrier-to-noise ratio (C/N). Although poor C/N of a measurement does not mean that the measurement from that particular satellite is contaminated

and high C/N does not mean that a observation is of good quality this situation can be close to the at-most-one bad observation situation that we are considering in this article. The error of the estimated position is illustrated by Figure 6 where the error of the BRAIM estimate and ordinary Kalman filtered position are given. Several significant errors are excluded when the BRAIM method is used.

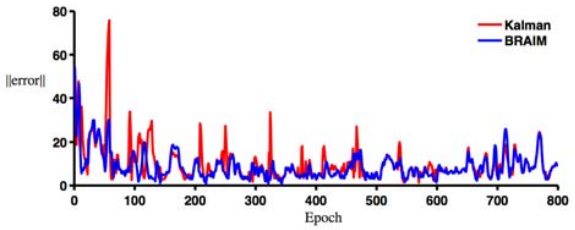

Fig.6. Errors of Kalman filtered position estimate versus the error given by the BRAIM method on a real GPS data vehicular test.

CONCLUSIONS

In the current report we applied Bayesian model comparison theory to GNSS integrity monitoring problem and introduced Bayesian receiver autonomous integrity monitoring technique (BRAIM). It was shown through simulations that the new proposed method obtains similar performance to traditional RAIM/FDE processing method. Better performance can be achieved if good prior information for the unknown parameters is available. The clearest advantage of the new proposed method is its foundations in Bayesian statistics, so that method parameters can be interpreted more easily than the traditional concepts of significance, power of the test etc. Drawback of the method is the requirement to have prior distributions for parameters and prior odd ratios for the models, but on the other hand this can be considered an advantage as this information may well be available (e.g. through filtering) and Bayesian theory enables to use this information.

The method can be developed further by formulating more realistic models than the current ones based on normal distributions and the generalization of the method to handle more than one faulty channel. In this paper we have not discussed position-domain integrity information; such information could be obtained by computing credibility regions, as is standard in Bayesian statistics [10,11].

ACKNOWLEDGMENTS

This work was carried out in the project Future GNSS Applications and Techniques (FUGAT) funded by the Finnish Funding Agency for Technology and Innovation (Tekes)

REFERENCES

[1] W. Baarda, "A testing procedure for use in geodetic networks," *Netherlands Geodetic Commission, Publication on Geodesy*, New Series 2, No. 5, Delft, Netherlands, 1968.

[2] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, Springer-Verlag New York, Inc. 2006.

[3] R. G. Brown, "A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods," *Navigation: Journal of Institute of Navigation*, 39(3): 101-116, 1992.

[4] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering: with MATLAB exercises and solutions*, John Wiley & Sons, Inc. 1997.

[5] S. Hewitson, H. K. Lee and J. Wang, "Localizability analysis for GPS/Galileo receiver autonomous integrity monitoring," *The Journal of Navigation*, 57: 245-259, 2004.

[6] H. Jeffreys, *Theory of Probability*, Oxford University Press, 3rd edition, 1961.

[7] R. E. Kass and A. Raftery, "Bayes factors," *Journal of the American Statistical Association*, 90(430): 773-795.

[8] H. Kuusniemi, *User-Level Reliability and Quality Monitoring in Satellite-Based Personal Navigation*, PhD thesis, Tampere University of Technology, 2005.

[9] H. Leppäkoski, H. Kuusniemi and Jarmo Takala, "RAIM and complementary Kalman filtering for GNSS reliability enhancement," in *IEEE Position Location and Navigation Conference PLANS* 2006, April 25-28, San Jose CA. 2006.

[10] P. B. Ober, *Integrity Prediction and Monitoring of Navigation Systems*, PhD thesis, Teknische Universiteit Delft, 2003.

[11] C. P. Robert, *The Bayesian Choice: From Decision-theoretic Foundations to Computational Implementation*, Springer Science+Business Media, LLC, 2007.

PUBLICATION 3

Henri Pesonen: Outlier-robust Bayesian filter with integrity monitoring for GNSS positioning. In *Proceedings of European Navigation Conference - ENC-GNSS 2009*, May 3 – 6, 2009.

Outlier-robust Bayesian Filter with Integrity Monitoring for GNSS Positioning

Henri Pesonen, *Tampere University of Technology*

BIOGRAPHY

Henri Pesonen received his M.Sc. degree from Tampere University of Technology in 2006 and is pursuing his Ph.D. studies. His research interests are robust and reliable positioning methods and Bayesian statistics.

ABSTRACT

We present an outlier-robust positioning Bayesian filter based on Gaussian mixture distributions and a fast sequential technique to approximate the posterior distribution. Also, a receiver autonomous integrity monitoring (RAIM) scheme based on Bayesian decision theory is introduced. The performances of the proposed methods are compared via simulations in urban personal positioning scenarios. The simulations show that the proposed techniques work well as outlier-robust positioning methods and that the integrity monitoring technique further enhances the performance.

1 INTRODUCTION

Filtering is the main approach for solving navigation and tracking problems in which the goal is to find an estimate of the kinetic state (position, velocity, etc.) of an object using current and past measurement data. Currently the standard positioning method for satellite navigation is the extended Kalman filter (EKF) [1], which is fast, reliable and accurate given a clear view of the sky. However, in urban environments the signals are affected by reflections and attenuations. As a result of these NLOS and multipath effects, the statistical properties of the positioning signals can differ from their nominal values and the use these signals with EKF can lead to significantly degraded performance. This is a well-known problem and many estimation methods have been developed that are robust against deviations from the model assumptions. Huber's robust M-estimation techniques have been used as a static estimation methods [2, 3] and to compute weights for the innovations in Kalman filter so that the faulty observations have only little influence on the estimate [4]. Similar methods that modify the Kalman innovation has been considered in [5, 3]. Different approaches were considered in [6], where an interacting multiple model filter was introduced to enable positioning under spoofing, and in [7] where a detection-identification-adaptation method, which enhances Kalman filter algorithm with recursive hypothesis testing, was introduced.

Another way to deal with failures in the positioning system is to monitor the quality of the signals and the position estimate, and if necessary and possible, to enhance the quality. The process of monitoring the quality and ensuring that the performance is within tolerable limits is called receiver autonomous integrity monitoring (RAIM). Various statistical testing procedures are currently the basis of RAIM techniques, and recently also Bayesian methods have been proposed [8, 9]. Traditional RAIM techniques were developed for aviation applications and make the assumption of only one possible faulty observation at a time. However, for personal positioning in urban environment, this assumption is too restrictive. Furthermore, RAIM techniques have traditionally been so called snapshot techniques in which the quality of the system is investigated at each epoch based only on the current observations.

In this work we use a Bayesian filtering method that is desensitized against failures and can be used naturally to monitor integrity. Bayesian filtering computes a probability distribution of the unknown kinematic state given all the data gathered up to that point, from which it is possible to find not only the optimal estimate of the state but also a measure of the quality of the estimate. If we model the faulty observations using Gaussian mixture densities, we can solve the problem using the Gaussian mixture filter (GMF) [10, 11], which have many applications in positioning [12]. In this paper we propose two approxima-

tions of GMF as outlier-robust positioning methods, and a RAIM-technique based on Bayesian decision theory.

The paper is organized as follows. First, in Section 2 we formulate a model based on Gaussian mixture probability distributions for the positioning problem that takes into account the possibility of faulty observations. In Section 3 we present the problem of assessing the integrity of the system as a decision theoretic problem which is solved using numerical integration. In Section 4 we test the performance of the positioning and integrity monitoring methods in urban positioning simulations. Finally, in Section 5 we conclude the article with some discussion.

2 PROBLEM FORMULATION

We consider a Bayesian filtering approach for the problem of positioning when there is a possibility of a faulty observations occurring. The basic state-space model we are considering for the positioning problem is

$$\begin{aligned} x_{k+1} &= F_k x_k + w_k \\ y_k &= h_k(x_k) + v_k, \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^{n_x}$ is the state of the system and $y_k \in \mathbb{R}^{n_k}$ is a vector of measurements at time t_k. Noise processes w_k, v_k are modeled as white, mutually independent and independent of the initial state x_0. In Bayesian filtering we compute the posterior probability distribution $p(x_k|y_{1:k})$ of the state given all the available, current and past observations. The posterior distribution can be solved recursively by alternating *prediction* and *update* steps. The prediction step is defined as

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})\mathrm{d}x_{k-1} \tag{2}$$

and the update step is defined as

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})\mathrm{d}x_k}, \tag{3}$$

which in general can't be computed analytically. If the $h(\cdot)$ is linear and w_k, v_k are jointly Gaussian, then the posterior distribution is a Gaussian with mean and covariance given by the well-known Kalman filter recursions [13].

If the observation function is nonlinear, then the posterior distribution can be approximated as a Gaussian computed by the extended Kalman filter (EKF) algorithm, which linearizes the observation function around the prior mean [14]. In satellite navigation EKF is currently the standard solution, due to its low computational demands and the fact that generally it works very well.

One situation in which the performance of EKF can be degraded significantly is when faulty observations or outliers occur [5, 3]. Faulty observations are measurements that contain less information about the unknown parameters than the regular observations, sometimes they are taken to contain no information at all about the parameters to be estimated. In the latter case it is sometimes justified to not to use them at all in estimation process. However, in this case, the faulty observations have to be identified reliably. In this work we take a Bayesian approach to model the faulty observations [15]. If the ith observation at time t_k is good, then we model it to have a sampling distribution

$$p(y_{k,i}|x_k) = N^{h_{k,i}(x_k)}_{\sigma^2_{k,i}}(y_{k,i}), \tag{4}$$

where $N^{\mu}_{\Sigma}(\cdot)$ denotes a normal distribution with mean μ and covariance Σ and $h_{k,i}$ is the ith element of the observation function at time t_k. If the jth observation is taken to be a faulty one, then we model it as a sample from a large-variance distribution

$$p(y_{k,j}|x_k) = N^{h_{k,j}(x_k)}_{c^2\sigma^2_{k,j}}(y_{k,j}), \quad c > 1. \tag{5}$$

Now, if there is uncertainty as to whether any observation is faulty or not, but the ith observation has a prior probability ϵ of being faulty and $1-\epsilon$ to be a good one, we model the sampling sampling distribution of $y_{k,i}$ with a Gaussian mixture distribution

$$\begin{aligned} p(y_{k,i}|x_k) = (1-\epsilon)&N^{h_{k,i}(x_k)}_{\sigma^2_{k,i}}(y_{k,i}) \\ &+ \epsilon N^{h_{k,i}(x_k)}_{c^2\sigma^2_{k,i}}(y_{k,i}). \end{aligned} \tag{6}$$

We assume that the observations given x_k are independent. Then the complete sampling distribution is a mixture of 2^{n_k} components

$$\begin{aligned} p(y_k|x_k) &= \prod_{i=1}^{n_k} p(y_{k,i}|x_k) \\ &= \sum_{r_k \in \mathcal{R}_k} \alpha^{r_k} N^{h_k(x_k)}_{R_k^{r_k}}(y_k), \end{aligned} \tag{7}$$

where

$$\mathcal{R}_k = \{(r_{k_1}, \ldots, r_{k_{n_k}}) | r_{k_j} \in \{0, 1\}, \forall j\}, \tag{8}$$

is the set of vectors r_k in which ith element is 1 if the ith observation is good and 0 if it's faulty. The coefficients

$$\alpha^{r_k} = (1-\epsilon)^{\sum_{j=1}^{n_k} r_{k_j}} \epsilon^{n_k - \sum_{j=1}^{n_k} r_{k_j}}$$

are the weights for sampling distribution which have a combination of good and faulty observations as given by r_k. They can also be interpreted as the a priori probabilities of the models defined by the diagonal covariance matrices

$$\begin{aligned} R_k^{r_k} = &\lceil \sigma_{k,1}^2 r_{k_1}, \ldots, \sigma_{k,k_{n_k}}^2 r_{n_k} \rfloor \\ &+ \lceil c^2 \sigma_{k,1}^2 (1-r_{k_1}), \ldots, c^2 \sigma_{k,n_k}^2 (1-r_{k_{n_k}}) \rfloor, \end{aligned}$$

of $y_k | x_k, r_k$ when the partitioning into of good and faulty observations is given by r_k.

Computing the posterior distribution is done using GMF and first order Taylor polynomial approximations of the nonlinear observations function $h_k(\cdot)$ around the prior mean $\bar{x}_{k|k-1}$ as in EKF

$$h_k(x_k) \approx h_k(\bar{x}_{k|k-1}) + H_k(x_k - \bar{x}_{k|k-1}), \quad (9)$$

Let the posterior distribution of x_{k-1} be

$$p(x_{k-1}|y_{1:k-1}) = N_{P_{k-1|k-1}}^{\bar{x}_{k-1|k-1}}(x_{k-1}), \quad (10)$$

then the prior distribution of x_k can be computed using the state transition model as

$$p(x_k|y_{1:k-1}) = N_{P_{k|k-1}}^{\bar{x}_{k|k-1}}(x_k), \quad (11)$$

where

$$\begin{aligned} \bar{x}_{k|k-1} &= F_{k-1} \bar{x}_{k-1|k-1} \\ P_{k|k-1} &= F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1}. \end{aligned} \quad (12)$$

Then the posterior distribution is a Gaussian mixture distribution

$$p(x_k|y_{1:k}) = \sum_{r_k \in \mathcal{R}_k} \beta^{r_k} N_{P_{k|k}^{r_k}}^{\bar{x}_{k|k}^{r_k}}(x_k), \quad (13)$$

where the mean and the covariance of the Gaussian components are

$$\begin{aligned} \bar{x}_{k|k}^{r_k} &= \bar{x}_{k|k-1} + K_k^{r_k}(y_k - h_k(\bar{x}_{k|k-1})) \\ P_{k|k}^{r_k} &= P_{k|k-1} - K_k^{r_k} H_k P_{k|k-1}. \end{aligned} \quad (14)$$

Each of the components depend on a Kalman gain

$$K_k^{r_k} = P_{k|k-1} H_k^T (S_k^{r_k})^{-1}, \quad (15)$$

where

$$S_k^{r_k} = H_k P_{k|k-1} H_k^T + R_k^{r_k} \quad (16)$$

is the approximated covariance of the innovation $y_k - h_k(x_k|y_{1:k-1})$ given the model defined by r_k. The weights β^{r_k} also depend on the realized observations, the sampling distribution models and the a priori weights α^{r_k}

$$\beta_k^{r_k} = \frac{\alpha^{r_k} N_{S_k^{r_k}}^{h_k(\bar{x}_{k|k-1})}(y_k)}{\sum_{r_k \in \mathcal{R}_k} \alpha^{r_k} N_{S_k^{r_k}}^{h_k(\bar{x}_{k|k-1})}(y_k)} \quad (17)$$

If (13) were used to compute the prior distribution at the next time-step, then the number of the components would grow from 2^{n_k} to $2^{n_k} \cdot 2^{n_{k+1}}$ which in effect makes the computation of the posterior infeasible in practice. This is why approximation techniques have been developed. Including merging similar mixture components and deleting components with small weights. In this work we use two different methods both of which produce a single Gaussian each time step. This keeps the algorithm computationally light.

The first method consists of merging all mixture components into a single Gaussian at the end of the filter update step by matching the mean and the variance of the Gaussian mixture distribution. This is optimal in the Kullback-Leibler sense [16]. For a Gaussian mixture distribution (13) the mean and covariance are [14]

$$\begin{aligned} \bar{x}_{k|k} &= \sum_{r_k \in \mathcal{R}_k} \beta_k^{r_k} \bar{x}_{k|k}^{r_k} \\ P_{k|k} &= \sum_{r_k \in \mathcal{R}_k} \beta_k^{r_k} (P_{k|k}^{r_k} + (\bar{x}_{k|k} - \bar{x}_{k|k}^{r_k})(\bar{x}_{k|k} - \bar{x}_{k|k}^{r_k})^T) \end{aligned} \quad (18)$$

The second method we introduce, which to our knowledge is original, consists of merging the observations sequentially into a single Gaussian using moment matching. This method is computationally light at the expense of losing information about the overall mixture distribution at each merging.

$$\begin{aligned} p(x_k|y_{1:k}) &\propto \prod_{i=2}^{n_k} p(y_{k,i}|x_k) p(y_{k,1}|x_k) p(x_k|y_{1:k-1}) \\ &\tilde{\propto} \prod_{i=3}^{n_k} p(y_{k,i}|x_k) p(y_{k,2}|x_k) p(x_k|y_{1:k-1,1}) \\ &\tilde{\propto} \cdots \tilde{\propto} p(y_{k,y_{n_k}}|x_k) p(x_k|y_{1:k-1,n_k-1}), \end{aligned} \quad (19)$$

where, given

$$p(x_k|y_{1:k-1,m-1}) = N^{\bar{x}_{k|k-1,m-1}}_{P_{k|k-1,m-1}}(x_k) \tag{20}$$

and denoting $p(x_k|y_{1:k-1}) = p(x_k|y_{1:k-1,0})$

$$\begin{aligned}
p(x_k|y_{1:k-1,m}) &= N^{\bar{x}_{k|k-1,m}}_{P_{k|k-1,m}}(x_k) \\
\bar{x}_{k|k-1,m} &= \sum_{i=0}^{1} \beta_k^i \bar{x}^i_{k|k-1,m} \\
P_{k|k-1,m} &= \sum_{i=0}^{1} \beta_k^i (P_{k|k-1,1} + (\bar{x}_{k|k-1,m} - \\
&- \bar{x}^i_{k|k-1,m})(\bar{x}_{k|k-1,m} - \bar{x}^i_{k|k-1,m})^T)
\end{aligned} \tag{21}$$

The mixture components are computed as

$$\begin{aligned}
\bar{x}^i_{k|k-1,m} &= \bar{x}_{k|k-1,m-1} + \\
&\quad K^i_{k,m}(y_{k,m} - h_{k,m}(\bar{x}_{k|k-1,m-1})) \\
P^i_{k|k-1,m} &= (I - K^i_{k,m} H^T_{k,m}) P_{k|k-1,m-1},
\end{aligned} \tag{22}$$

where $H_{k,m}$ is the mth row vector of H_k and

$$\begin{aligned}
K^i_{k,m} &= P_{k|k-1,m-1} H^T_{k,m} (S^i_k)^{-1} \\
S^i_k &= H_{k,m} P_{k|k-1,m-1} H^T_{k,m} + (c^2)^{1-i} \sigma^2_{k,m}
\end{aligned} \tag{23}$$

Finally the posteriori weights are computed as

$$\begin{aligned}
\beta^i_k &= \frac{\alpha^i_k N^{h_{k,m}(\bar{x}_{k|k-1,m-1})}_{S^i_k}(y_{k,m})}{\sum_{i=0}^{1} \alpha^i_k N^{h_{k,m}(\bar{x}_{k|k-1,m-1})}_{S^i_k}(y_{k,m})} \\
\alpha^i_k &= (1-\epsilon)^{(1-i)} \epsilon^i
\end{aligned} \tag{24}$$

3 INTEGRITY MONITORING

In Bayesian filtering, the posterior distribution contains all the knowledge about the uncertain state. However, in applications the posterior needs to be summarized by a few numbers, for example, the posterior distribution's mean as a point estimate of the state and the covariance as in indicator of the quality of the estimate.. Here, we describe a system in which we report a point estimate of the state and also a decision as to whether the system has sufficient 'integrity', a measure of trust which can be placed on the correctness of the position estimate. The method described here is similar to the one described in [8, 17] but the approach is different.

In [9] we described a Bayesian integrity monitoring method, which was based on formulation of models for faulty observations and comparing the posteriori weights (17) of different models. If a single model stood out, i.e. correct set of faulty measurements were detected with high probability, a decision was made that a correct set of faulty observations was identified and integrity was achieved. This approach does not take into account that faulty observations do not necessarily cause the position estimate to be erroneous, nor does good data always give a position estimate that is accurate. Here we introduce a direct method that deals directly with the position error, and the probabilities of good/faulty data models are taken into account automatically. This is done using the complete posterior distribution of the state x_k at time t_k to monitor the integrity of the system.

The integrity monitoring scheme described here is based on decisions whether the error of the estimate $\hat{x}_k$ is within acceptable limit. We define the set of acceptable estimates as

$$\Omega_T(\hat{x}_k) = \{x_k : ||x_k - \hat{x}_k|| < T\}, \tag{25}$$

where T is a user defined parameter. If the true error is within this region, the estimate is trustworthy and the positioning system has sufficient integrity. Although the true error is unknown, we have an approximation of the true posterior probability distribution of the state.

We formulate the integrity monitoring problem as a decision theoretic problem [18]. We have two actions to choose from

a_1 : Report 'system has sufficient integrity'

a_2 : Report 'system doesn't have sufficient integrity'.

For these actions we define a cost function

$L(x_k, a)$	$x_k \in \Omega_T(\hat{x}_k)$	$x_k \notin \Omega_T(\hat{x}_k)$
a_1	0	K_1
a_2	K_0	0

, (26)

where K_0 and K_1 are user-defined costs of *false alarm* and *missed failure detection*. Arguably $K_1 > K_0$ as often false trust in the position estimate is more harmful that occasional false warning. The action a_i is chosen which minimizes the expected loss

$$\begin{aligned}
\mathbb{E}(L(x_k, a)|y_{1:k}) &= \int L(x_k, a) p(x_k|y_{1:k}) \mathrm{d}x_k \\
&= \begin{cases} K_1(1 - P(x_k \in \Omega_T(\hat{x}_k)|y_{1:k})), & a = a_1 \\ K_0 P(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) & a = a_2. \end{cases}
\end{aligned} \tag{27}$$

The optimal action is a_1 if

$$P(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) \geq \frac{K_1}{K_0 + K_1} \tag{28}$$

and a_2 otherwise. To find the optimal action, we need to compute the probability $\Omega_T(\hat{x}_k)$

$$\begin{aligned} P(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) &= \int_{\Omega_T(\hat{x}_k)} p(x_k|y_{1:k})\mathrm{d}x_k, \\ &= \sum_{r_k} \beta^{r_k} \int_{\Omega_T(\hat{x}_k)} N^{r_k}_{\bar{x}^{r_k}_{k|k}, P^{r_k}_{k|k}}(x_k)\mathrm{d}x_k \end{aligned} \tag{29}$$

If we merge the components in a manner described in Section 2, then the integral is of a single Gaussian over the hypersphere $\Omega_T(\hat{x}_k)$.

In general case an integral of a Gaussian over $\Omega_T(\hat{x})$ can't be computed analytically but has to be solved approximatively. However, in the important case of horizontal error, which is considered here, the integration region is a disk $\Omega_T(\hat{z}_k)$, where z_k is the horizontal position. In this case we can compute the integral numerically fast and accurately. For simplicity, we assume that $\hat{z}_k = 0$ as integration over an arbitrary disk can be transformed to an integration over origin-centered disk. We get

$$\begin{aligned} &\int_{\Omega_T(0)} N^{\mu}_{\Sigma}(z)\mathrm{d}z \\ &= \int_{-T}^{T} \int_{-\sqrt{T^2-z_1^2}}^{\sqrt{T^2-z_1^2}} N^{\mu_2|z_1}_{\Sigma_{22}|z_1}(z_2)\mathrm{d}z_2 N^{\mu_1}_{\Sigma_{11}}(z_1)\mathrm{d}z_1, \end{aligned} \tag{30}$$

where

$$\begin{aligned} \mu_2|z_1 &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(z_1 - \mu_1) \\ \Sigma_{22}|z_1 &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}. \end{aligned} \tag{31}$$

We denote

$$F(z_1) = \int_{-\sqrt{T^2-z_1^2}}^{\sqrt{T^2-z_1^2}} N^{\mu_2|z_1}_{\Sigma_{22}|z_1}(z_2)\mathrm{d}z_2, \tag{32}$$

which we can compute fast and accurately using error functions. The integration (30) is then

$$\int_{-T}^{T} F(z_1) N^{\mu_1}_{\Sigma_{11}}(z_1)\mathrm{d}z_1. \tag{33}$$

The integral (33) can be solved approximately using some one dimensional quadrature rule which is a weighted sum of integrand values evaluated at predefined points.

$$\int_{-T}^{T} F(z_1) N^{\mu_1}_{\Sigma_{11}}(z_1)\mathrm{d}z_1 \approx \sum_i w_i F(z_1^i) N^{\mu_1}_{\Sigma_{11}}(z_1^i) \tag{34}$$

Given that the integrand (33) can be very 'sharp' peaked function on $[-T, T]$ e.g. in a case where the variances of the elements are very small, we use adaptive Lobatto quadrature [19] to compute (34). Adaptive quadrature rules automatically seeks to place the nodes z_1^i in an optimal manner so that as few as possible nodes are needed to obtain accurate approximation to the integral. Comparison of different quadrature rules is left for future work.

4 SIMULATIONS

We compare the performance of the proposed method as an outlier-robust filter and as a RAIM technique in simulated GNSS positioning scenarios in urban environment. The state is generated using model in which the velocity and clock-drift are a random walk processes. The state vector consists of three position coordinates $x_{1:3}$, three velocity elements $x_{4:6}$, clock bias x_7 and clock drift x_8. The motion model is

$$x_{k+1} = F_k x_k + w_k, \tag{35}$$

where

$$F_k = \begin{bmatrix} I_3 & I_3 & 0 & 0 \\ 0 & I_3 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{36}$$

and

$$Q_k = \begin{bmatrix} \sigma^2_{\text{vel}} \begin{bmatrix} \frac{1}{3}I_3 & \frac{1}{2}I_3 \\ \frac{1}{2}I_3 & I_3 \end{bmatrix} & 0 \\ 0 & \sigma^2_{\text{cd}} \begin{bmatrix} \frac{1}{3} & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix} \end{bmatrix}. \tag{37}$$

The variances σ^2_{vel} and σ^2_{cd} are in all simulations 1^2 and 0.001^2 respectively. These variances are used to generate the tracks but filtering methods use incorrect variance $\sigma^2_{\text{user}} = 4^2$ as the variance of the velocity process. Constellations of 6 visible satellites s_i, and pseudorange observations

$$h_{k,i}(x_k) = ||x_{1:3,k} - s_i|| + x_7 \tag{38}$$

with good and faulty noise distributions

$$\begin{aligned} p_G(y_{k,i}|x_k) &= N^{h_{k,i}(x_k)}_{\sigma_1^2}(y_{k,i}) \\ p_B(y_{k,i}|x_k) &= N^{h_{k,i}(x_k)+\mu}_{\sigma_2^2}(y_{k,i}), \end{aligned} \tag{39}$$

are generated. We use $\mu = 50$ and $\sigma_2 = 25$ so that the true noise is not zero mean. We are simulating 200 epochs long scenarios during which the satellites

Interval	k	Contaminated observations y_i
I	$[25, 50]$	$i = 3$
II	$[75, 100]$	$i = 1, 2$
III	$[125, 150]$	$i = 4, 5, 6$
IV	$[175, 200]$	$i = 1, 2, 3, 4$

Table 1: Faulty observations occur in batches defined by time intervals I – IV.

are taken to be stationary for simplicity. In simulations we generate faulty observations in batches during fixed time intervals as given by Table 1.

We test the following methods:
GMF1: The Gaussian mixture filter in which we merge all the components into a single one using moment matching after the decision about the system integrity is made.
GMF2: The Gaussian mixture filter in which we merge all the components into a single one using moment matching. The decision about the integrity is based on error probability computed after the components have been merged.
GMF3: The Gaussian mixture filter in which we sequentially process observations and merge the resulting components using moment matching into a single one, as described in Section 2. The decision about the integrity is made based on the approximative error probability.
GMF4: The Gaussian mixture filter in which we select the most probable model, which corresponds to the mixture component with largest weight. The integrity monitoring is based on the posterior odds of the most probable model against the second most probable model [9].
BFH: The robust Bayesian filter introduced in [20] and shown to work well in GNSS based positioning in [5, 3].
RAIM/FDE: The snapshot RAIM method introduced by [21] is one of the standard integrity monitoring/ fault detection and exclusion-techniques. It is based on a global least squares residuals test to detect errors and a local w-test to identify them [21, 22].

4.1 Example

We consider one GNSS positioning example in urban environment. The GMF1 is compared against the BFH as a robust filtering method with respect to the horizontal position error (HPE). The proposed method as a RAIM technique is compared against the RAIM/FDE. GMF1 uses parametes values $K_0 = 1$, $K_1 = 100$, $T = 25$ and RAIM/FDE uses parameter values $\alpha = 0.1$ and $\beta = 0.1$ [22]. BFH is tuned to perform as well as possible in this scenario. The coordinates of the satellites in this example are listed in Table 2 and the simulated track is within 1000 from the origin of the used coordinate system. The position dilution of precision (PDOP) [1] for this particular scenario is 2.5.

i	$s_i \cdot 10^{-7}$
1	$[9.4, 0.6, 1.4]^T$
2	$[5.2, 2.7, 1.4]^T$
3	$[-1.7, -3.9, 1.9]^T$
4	$[5.6, 3.9, 2.0]^T$
5	$[-1.0, 0.2, 1.5]^T$
6	$[1.4, 2.5, 1.4]^T$

Table 2: Satellite coordinates of the Example

The results of HPE and RAIM performance are given in Figure 1. Given the observation noise defined by (39) occurring in batches given by Table 1 GMF1 performs much more robustly than BHF. This is clear as BHF is designed to work robustly under observation noise distributions that have symmetrical zero mean densities.

As a RAIM method GMF1 performs much better than RAIM/FDE. RAIM/FDE works well in intervals I and II when there is only one and two faulty observations present. When three and four observations are faulty, RAIM/FDE can often detect the presence large errors but often very large errors go undetected, especially in time interval IV. From the Figure it can be seen that GMF1 performs very robustly in intervals I, II, and in intervals III and IV, when HPE becomes large, most of the time this can be detected by the proposed RAIM.

4.2 Summary of the simulations

We run the simulations 100 times and investigate the horizontal position root mean square error before (RMSE) and after (RMSE_{IM}) we have removed position estimates that were computed when sufficient integrity was not achieved. We also report the frequencies P_{00}, P_{10} and P_{01}, which correspond to *justified alarm*, *false alarm* and *missed detection* of the RAIM performed by the filters and RAIM/FDE. We use parameter values $K_0 = 1, K_1 = 100$ and $T = 25$. GMF4 uses threshold $T_{\text{GMF4}} = 10$ to decide whether the system is working correctly [9]. Finally, we report the maximum horizontal position error in the whole set of errors $\max e$ and in the set where we

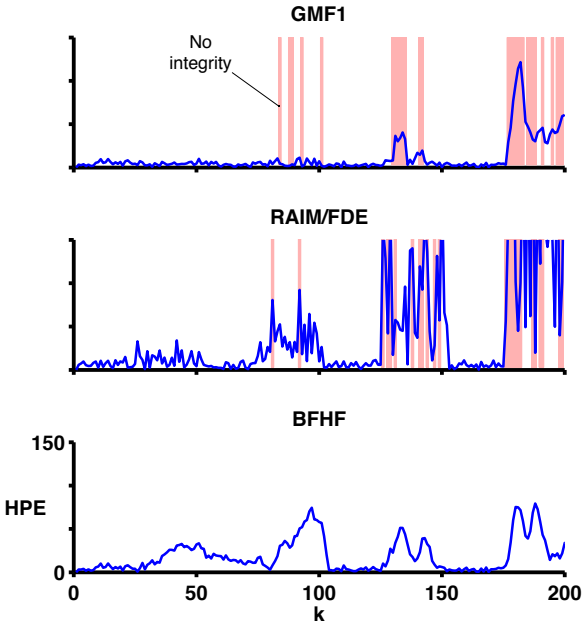

Figure 1: Example: GMF1 is more robust filter than BFH and performs better as RAIM method than RAIM/FDE

have removed errors given when sufficient integrity was not achieved $\max e_{\mathrm{IM}}$ which corresponds to the largest unremoved position error.

The results are summarized in Table 3. All the introduced methods GMF1, GMF2 and GMF3 perform well in the difficult test scenarios when compared to BFH and RAIM/FDE. Although differences between the new methods are small, GMF1 generally seem to perform better than GMF2 and GMF3. This is because of the approximations performed by GMF2 and GMF3. However, because GMF1 has to compute numerically 2^6 integrals at each epoch whereas GMF2 and GMF3 compute only one, GMF1 is computationally much more demanding. The performance of sequential GMF3 was almost as good as the performance of GMF1 and GMF2 but it is computationally much less demanding.

	P_{00}	P_{10}	P_{01}	RMSE	$\mathrm{RMSE}_{\mathrm{IM}}$	max e	max e_{IM}
GMF1	0.109	0.091	0.050	13	8	229	128
GMF2	0.094	0.072	0.066	13	9	229	138
GMF3	0.080	0.062	0.079	14	9	238	127
GMF4	0.127	0.391	0.052	17	12	258	258
BFH	–	–	–	19	–	169	–
RAIM/ FDE	0.079	0.019	0.262	32	28	$7{\cdot}10^3$	$7{\cdot}10^3$

Table 3: Simulation results

5 CONCLUSIONS

Outlier-robust Bayesian filtering based on Gaussian mixture distributions were discussed and a fast sequential method for computing approximation of the posterior distribution was introduced. In addition, a RAIM technique based on Bayesian decision theory was introduced. It was shown through simulations that the proposed techniques perform well compared to a snapshot RAIM/FDE and a robust filtering method in difficult urban positioning scenarios. The proposed techniques can be used as a positioning methods that are not only robust against outliers but monitor the integrity and often correctly warns that the estimate is not reliable.

The drawback of the proposed methods is that they need models for faulty observations and the proposed RAIM method depends on a user-defined loss function to model the relative costs of false alarms and missed detection. It is very difficult to construct these so that the methods work well in different situations. The study on the influence of models and loss functions is left as future work.

ACKNOWLEDGEMENTS

The author would like to thank Robert Piché for his helpful comments. This work was carried out under the funding of Tampere University of Technology Graduate School.

REFERENCES

[1] E. D. Kaplan, ed., *Understanding GPS : principles and applications*. Artech House, Inc., 1996.

[2] X.-W. Chang and Y. Guo, "Huber's M-estimation in GPS positioning: Computational aspects," in *ION NTM 2004, 26-28 January 2004, San Diego, CA*, 2004.

[3] H. Pesonen, "Robust estimation techniques for GNSS positioning," in *Proceedings of NAV07-The Navigation Conference and Exhibition, 31.10.-1.11.2007, London, England*, 2007.

[4] Y. Yang, H. He, and G. Xu, "Adaptively robust filtering for kinematic geodetic positioning," *Journal of Geodesy*, vol. 75, pp. 109–116, 2001.

[5] T. Perälä and R. Piché, "Robust Extended Kalman filtering in hybrid positioning applications," in *Proceedings of the 4th Workshop on Positioning, Navigation and Communication (WPNC'07), March 22*, pp. 55–64, 2007.

[6] Y. Oshman and M. Koifman, "Robust navigation using the Global Positioning System in the presence of spoofing," *Journal of Guidance, Control and Dynamics*, vol. 29, no. 1, pp. 95–104, 2006.

[7] P. J. G. Teunissen, "Quality control in integrated navigation systems," *IEEE Transactions on Aerosp. Electr. System*, vol. 5, pp. 35–41, July 1990.

[8] P. B. Ober, *Integrity Prediction and Monitoring of Navigation Systems*. PhD thesis, Teknische Universiteit Delft, 2003.

[9] H. Pesonen and R. Piché, "Bayesian receiver autonomous integrity monitoring technique," in *Proceedings of Institute of Navigation International Technical Meeting 2009, Anaheim, CA*, 2009.

[10] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. AC-17, pp. 439–448, August 1972.

[11] H. Sorenson and D. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, pp. 465–479, 1971.

[12] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2007*, 2007.

[13] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME-Journal of Basic Engineering*, vol. 82, 1960.

[14] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[15] G. Box and G. Tiao, "A Bayesian approach to some outlier problems," *Biometrika*, vol. 55, no. 1, pp. 119–129, 1968.

[16] A. R. Runnalls, "Kullback-Leibler approach to Gaussian mixture reduction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, pp. 989–999, July 2007.

[17] P. Ober, "Integrity according to Bayes," in *Position Location and Navigation Symposium, IEEE 2000, San Diego, CA, USA*, pp. 325–332, 2000.

[18] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag New York, Inc., 2006.

[19] W. Gander and W. Gautchi, "Adaptive quadrature – revisited," *BIT Numerical Mathematics*, vol. 40, pp. 84–101, March 2000.

[20] C. J. Masreliez and R. D. Martin, "Robust Bayesian estimation for the linear model and robustifying the Kalman filter," *IEEE Transactions on Automatic Control*, vol. 22, no. 3, pp. 361–271, 1977.

[21] W. Baarda, *A Testing Procedure for Use in Geodetic Networks*. Netherlands Geodetic Commission, Publication on Geodesy, New Series 2, No. 5, Delft, Netherlands, 1968.

[22] H. Kuusniemi, *User-Level Reliability and Quality Monitoring in Satellite-Based Personal Navigation*. PhD thesis, Tampere University of Technology, 2005.

PUBLICATION 4

Henri Pesonen and Robert Piché: Bayesian positioning using Gaussian mixture models with time-varying component weights. In *Proceedings of the 2011 Joint Statistical Meetings*, July 30– August 4, 2011, pages 4516 – 4524

Bayesian Positioning Using Gaussian Mixture Models with Time-varying Component Weights

Henri Pesonen Robert Piché *

Abstract
Gaussian mixture models are often used in target tracking applications to take into account maneuvers in state dynamics or changing levels of observation noise. In this study it is assumed that the measurement or the state transition model can have two plausible candidates, as for example in positioning with line-of-sight or non-line-sight-signals. The plausibility described by the mixture component weight is modeled as a time-dependent random variable and is formulated as a Markov process with a heuristic model based on the Beta distribution. The proposed system can be used to approximate some well-known multiple model systems by tuning the parameter of the state transition distribution for the component weight. The posterior distribution of the state can be solved approximately using a Rao-Blackwellized particle filter. Simulations of GPS pedestrian tracking are used to test the proposed method. The results indicate that the new system is able to find the true models and its root mean square error-performance is comparable to filters that know the true models.

Key Words: Bayesian filtering, multiple model filtering, model uncertainty, Rao-Blackwellization

1. Introduction

Positioning and tracking are often carried out by modeling the problem as discrete time stochastic systems [2]. By modeling the motion of the mobile station (MS) and the relationship of the state (position, velocity, acceleration, etc, ...) and some observable quantities stochastically, we can solve the posterior distribution of the state using Bayesian framework optimally. Furthermore, under certain assumptions the computations can be carried out recursively. However, in positioning systems the chosen models may not be valid at every time step. For example, in satellite positioning system such as global positioning system (GPS) signals obtained from directly visible and non-visible satellites have different statistical qualities [8]. As another example, a MS has different motion models when carried by a pedestrian vs. a car passenger.

Two common solutions to the problem are to use mixture distributions to describe the systems, or to describe the system using switching models. In the case of Gaussian linear systems, the posterior distribution can be evaluated analytically and recursively with the celebrated Kalman filter [7]. When using Gaussian mixture distributions to describe the stochastic components, Gaussian mixture filter (GMF) evaluates the posterior distribution analytically, at least in theory [10]. GMF gives generally only a theoretical solution to the problem, as it requires an exponentially growing number of mixture components to describe the posterior distribution. In practice, approximations such as multiple model filters (MMFs) are used [2].

A problem when using mixture distributions to describe the models, is that the weights, or the probabilities of the models, are static. However, in general case the probabilities will change with time. For example, often a two-component mixture distribution is used to describe the observation error, with one component describing a 'good' observation and the other describing a 'bad' observation [3, 9]. In the case of GPS positioning, obtaining

*Tampere University of Technology, Department of Mathematics, Korkeakoulunkatu 1, 33720, Tampere, Finland

a bad observation in an urban canyon is more likely than in a forest and much more likely than on a highway. We propose a method where the mixture weight of a two-component mixture distribution is modeled as a stochastic process, and is solved jointly with the state parameters using a Bayesian filter.

The paper is organized as follows. In Section 2 we describe the considered linear discrete time stochastic system, and in Section 3 we present a heuristic motion model for the time-varying mixture component weight based on the Beta-distribution. In Section 4 we formulate the Rao-Blackwellized particle filter (RBPF) for the described system. In Section 5 we test the performance of the filters based on the discussed models and conclude the study in Section 6.

2. Problem formulation

We consider a linear discrete time stochastic system

$$x_{k+1} = F_k x_k + w_k \tag{1}$$
$$y_k = H_k x_k + v_k. \tag{2}$$
$$x_0 \sim \mathsf{N}\left(\bar{x}_{0|0}, P_{0|0}\right), \tag{3}$$

where $\mathsf{N}(\mu, \Sigma)$ is a Gaussian distribution with mean μ and covariance Σ. It is well-known that in the case of white Gaussian noises w_k and v_k the Kalman filter (KF) can be used to compute recursively the Gaussian posterior filtering distribution parameters. However, as well-known is the non-robustness of the KF.

A common approach for making the system more robust against modeling errors is to assume a set of models from which a member generated the state or the observations. These are often referred to multiple model (MM) systems. We consider the case where there are two possible models, although any number of models could be considered. The model is

$$\begin{aligned} p(x_{k+1}|x_k) &= \mathsf{N}\left(x_{k+1} \mid F_k x_k, Q_k\right) \\ p(y_k|x_k, \lambda_k) &= \mathsf{N}\left(y_k \mid H_k x_k, (1-\lambda_k)R_k^{(1)} + \lambda_k R_k^{(2)}\right), \end{aligned} \tag{4}$$

where the state model is Gaussian but observation error is modeled a Gaussian mixture (GM) distributed random variable. This could model for example the presence of line-of-sight/non-line-of-sight (LOS/NLOS) signals between MS and a satellite.

The parameter $\lambda_k \in \{0, 1\}$ is the model parameter, and it remains to be defined. In the simplest case it would be a Bernoulli distributed random variable independent of previous values

$$\begin{aligned} p(\lambda_k = 0 \mid \lambda_{k-1}) &= p(\lambda_k = 0) = 1 - \epsilon \\ p(\lambda_k = 1 \mid \lambda_{k-1}) &= p(\lambda_k = 1) = \epsilon. \end{aligned} \tag{5}$$

The probability ϵ represents the uncertainty of the model. Often the probability of the model depends on the time. One approach to take this into account is consider the multiple model approach for switching models. Now $\lambda_k \in \{0, 1\}$ is a Markov chain with the transition probabilities

$$\begin{aligned} p(\lambda_k = 0 \mid \lambda_{k-1} = 0) &= 1 - p(\lambda_k = 1 \mid \lambda_{k-1} = 0) = 1 - \epsilon^{(1)} \\ p(\lambda_k = 0 \mid \lambda_{k-1} = 1) &= 1 - p(\lambda_k = 1 \mid \lambda_{k-1} = 1) = 1 - \epsilon^{(2)}. \end{aligned} \tag{6}$$

This adaptive formulation of the multiple model state-space problem is sometimes called the jump Markov linear system (JMLS) [5]. Notice that (5) is a special case of (6) where

$\epsilon^{(1)} = \epsilon^{(2)} = \epsilon$. In theory, the posterior distribution can be found analytically for (4) with the Gaussian mixture filter (GMF) [10, 1]. GMF evaluates the posterior distribution using a bank of Kalman filters, increasing the number of filters at each time step. Computationally the complexity of GMF increases exponentially with time, and complexity reduction methods are needed [9]. Two main methods are pruning and merging, in which some mixture components are removed or combined, respectively. Monte Carlo simulation-based Rao-Blackwellized Particle Filters (RBPFs) are an example of the pruning approach. RBPFs automatically remove improbable components of the mixture posterior [5]. Different multiple model filters such as the interacting multiple model filter and the generalized pseudo-Bayesian approaches are popular examples of merging algorithms [2].

In real life applications where the environment has a major impact on the models, the assumption that the switching probability is known is not always very realistic. For example in a target tracking application with GPS data, the environment has a major impact on the data quality because objects such as trees and buildings block visibility to satellites and degrade the data quality significantly. In applications of this nature, the uncertainty about which model is generating the data is changing. In these kind of situations, a more realistic model could be one where the model uncertainty in consecutive time instants tend to be similar. In the next section we describe a state model for the model uncertainty with a continuous probability density.

3. Time-varying model uncertainty variable

We consider a hierarchical model for the MM filter where the switching probability ϵ_k in (5) and (6) is a time-dependent variable. The probability mass function of λ_k is now defined by ϵ_k

$$p(\lambda_k = 0 \mid \epsilon_k) = 1 - p(\lambda_k = 1 \mid \epsilon_k) = 1 - \epsilon_k \tag{7}$$

Constructing a model for the evolution of the model uncertainty parameter we have to take into account that the uncertainty $\epsilon_k \in [0, 1]$. We model the parameters as a Markov process and take the density $p(\epsilon_{k+1}|\epsilon_k)$ to be unimodal, with the mode near to the value of ϵ_k. A probability density fulfilling these criteria would be a Beta density

$$\mathsf{Beta}(\xi \mid \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \xi^{\alpha-1}(1 - \xi)^{\beta-1}. \tag{8}$$

The mode and variance of a Beta distributed random variable are

$$\mathsf{mode}\,(\xi) = \frac{\alpha - 1}{\alpha + \beta - 2}, \quad \mathsf{V}\,(\xi) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta - 1)}. \tag{9}$$

The Beta density function is unimodal when $\alpha, \beta > 1$. The variance of a Beta distributed random variable depends on α and β, with variance $\to 0$ as $\alpha, \beta \to \infty$.

We use a state-transition density

$$p(\epsilon_{k+1} \mid \epsilon_k, S) = \mathsf{Beta}(\epsilon_{k+1} \mid \epsilon_k(S - 2) + 1, (1 - \epsilon_k)S + 2\epsilon_k - 1), \tag{10}$$

where S is a tuning parameter. The mode and variance of (10) are

$$\mathsf{mode}\,(\epsilon_{k+1} \mid \epsilon_k, S) = \epsilon_k, \quad \mathsf{V}\,(\epsilon_{k+1} \mid \epsilon_k, S) = \frac{(1 - \epsilon_k)\epsilon_k}{S - 1}. \tag{11}$$

A larger tuning parameter S reduces the variance of $\epsilon_{k+1} \mid \epsilon_k$, and the expected model uncertainty is the previous model uncertainty. Different state transition densities are drawn in Figure 1. The hierarchical state-space model is illustrated by the directed acyclic graph (DAG) in Figure 2.

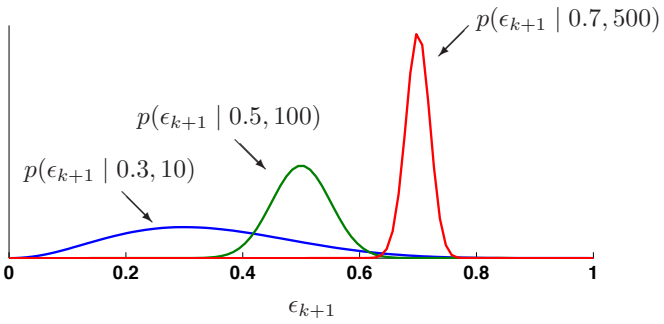

Figure 1: State transition densities with various parameter values.

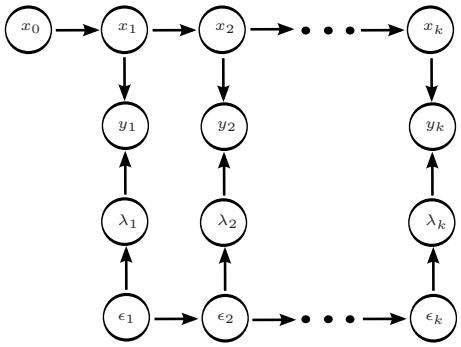

Figure 2: DAG of the hierarchical state-space model

4. Bayesian estimation

Bayesian filtering framework can be used to solve the state-space model with the time-varying model uncertainty. The posterior distribution $p(x_k|y_{1:k})$ can in theory be solved recursively [6]. In practice it can be approximated empirically using sequential Monte Carlo methods (SMC) [4]. The empirical representation of the posterior is a sum of N support points $\{(x_{0:k}^{(i)}, \lambda_{0:k}^{(i)}, \epsilon_{0:k}^{(i)}) : i = 1, \ldots, N\}$ with the corresponding weights $\{\omega_{1:k}^{(i)} : i = 1, \ldots, N\}$

$$\widehat{p_N}(x_{0:k}, \lambda_{0:k}, \epsilon_{0:k} \mid y_{1:k}) \approx \sum_{i=1}^{N} \omega_{0:k}^{(i)} \delta \left((x_{0:k}^{(i)}, \lambda_{0:k}^{(i)}, \epsilon_{0:k}^{(i)}) - (x_{0:k}, \lambda_{0:k}, \epsilon_{0:k}) \right). \tag{12}$$

The posterior filtering distribution can be obtained as a marginal distribution of (12). The performance of the SMC method can be enhanced in our system by using the fact that conditioned on $\lambda_{0:k}$, problem (4) can be solved optimally with the Kalman filter algorithm. This enables us to approximate the posterior using RBPF, where we need only to approximate empirically $p(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})$. If we write

$$\begin{aligned} p(x_k \mid y_{1:k}) &= \int_{[0,1]^{k+1}} \sum_{j=1}^{2^{k+1}} p(x_k \mid y_{1:k}, \epsilon_{0:k}, \lambda_{0:k}^{(j)}) p(\epsilon_{0:k}, \lambda_{0:k}^{(j)} \mid y_{1:k}) \mathrm{d}\epsilon_{0:k} \\ &= \int_{[0,1]^{k+1}} \sum_{j=1}^{2^{k}} p(x_k \mid y_{1:k}, \epsilon_{0:k}, \lambda_{0:k}^{(j)}) \frac{p(\epsilon_{0:k}, \lambda_{0:k}^{(j)} \mid y_{1:k})}{\pi(\epsilon_{0:k}, \lambda_{0:k}^{(j)} \mid y_{1:k})} \pi(\epsilon_{0:k}, \lambda_{0:k}^{(j)} \mid y_{1:k}) \mathrm{d}\epsilon_{0:k}, \end{aligned} \tag{13}$$

then the posterior can be approximated based on the strong law of large numbers as

$$\begin{aligned} & p(x_k \mid y_{1:k}) \\ & \approx \sum_{i=1}^{N} p(x_k \mid y_{1:k}, \epsilon_{0:k}, \lambda_{0:k}) \frac{p(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})}{\pi(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})} \delta\left((\epsilon_{0:k}^{(i)}, \lambda_{0:k}^{(i)}) - (\epsilon_{0:k}, \lambda_{0:k}) \right) \\ & = \sum_{i=1}^{N} p(x_k \mid y_{1:k}, \lambda_{0:k}^{(i)}) \frac{p(\epsilon_{0:k}^{(i)}, \lambda_{0:k}^{(i)} \mid y_{1:k})}{\pi(\epsilon_{0:k}^{(i)}, \lambda_{0:k}^{(i)} \mid y_{1:k})} \end{aligned} \tag{14}$$

where $\{\epsilon_{0:k}^{(i)}, \lambda_{0:k}^{(i)} : i = 1, \ldots, N\}$ is a sample drawn from the importance sampling distribution $\pi(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})$. In sequential importance sampling (SIS), potential importance distributions are restricted to be of the form

$$\pi(\lambda_{0:k}, \epsilon_{0:k} \mid y_{1:k}) = \pi(\lambda_0, \epsilon_0) \prod_{i=1}^{k} \pi(\lambda_i, \epsilon_i \mid y_{1:i}, \lambda_{0:i-1}, \epsilon_{0:1:i-1}), \tag{15}$$

where $\pi(\lambda_{0:k-1}, \epsilon_{0:k-1} \mid y_{1:k-1})$ is a marginal distribution at time $k-1$. This enables a sample set from $\pi(\lambda_k, \epsilon_k \mid y_{1:k})$ to be estimated by replacing the samples from $\pi(\lambda_{k-1}, \epsilon_{k-1} \mid y_{1:k-1})$ with a sample from $\pi(\lambda_k, \epsilon_k \mid y_{1:k}, \lambda_{0:k-1}^{(i)}, \epsilon_{0:1:k-1}^{(i)})$. Because we can write

$$\begin{aligned} p(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k}) &= \frac{p(y_k \mid \epsilon_{0:k}, \lambda_{0:k}, y_{1:k-1}) p(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\ &\propto p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\epsilon_k, \lambda_k \mid \epsilon_{0:k-1}, \lambda_{1:k-1}, y_{1:k-1}) p(\epsilon_{0:k-1}, \lambda_{0:k-1} \mid y_{1:k-1}) \\ &= p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\lambda_k \mid \epsilon_k, \epsilon_{0:k-1}, \lambda_{0:k-1}, y_{1:k-1}) \times \\ &\times p(\epsilon_k \mid \epsilon_{0:k-1}, \lambda_{0:k-1}, y_{1:k-1}) p(\epsilon_{0:k-1}, \lambda_{0:k-1} \mid y_{1:k-1}) \\ &= p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\lambda_k \mid \epsilon_k) p(\epsilon_k \mid \epsilon_{k-1}) p(\epsilon_{0:k-1}, \lambda_{0:k-1} \mid y_{1:k-1}), \end{aligned} \tag{16}$$

we have a recursive update formula for the empirical distribution weights

$$\begin{aligned} \omega_{0:k} &\triangleq \frac{p(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})}{\pi(\epsilon_{0:k}, \lambda_{0:k} \mid y_{1:k})} \\ &\propto \frac{p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\lambda_k \mid \epsilon_k) p(\epsilon_k \mid \epsilon_{k-1}) p(\epsilon_{0:k-1}, \lambda_{0:k-1} \mid y_{1:k-1})}{\pi(\lambda_0, \epsilon_0) \prod_{i=1}^{k} \pi(\lambda_i, \epsilon_i \mid y_{1:i}, \lambda_{0:i-1}, \epsilon_{0:i-1})} \\ &= \frac{p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\lambda_k \mid \epsilon_k) p(\epsilon_k \mid \epsilon_{k-1})}{\pi(\lambda_k, \epsilon_k \mid y_{1:k}, \lambda_{0:k-1}, \epsilon_{0:k-1})} \\ &\times \frac{p(\epsilon_{0:k-1}, \lambda_{0:k-1} \mid y_{1:k-1})}{\pi(\lambda_0, \epsilon_0) \prod_{i=1}^{k-1} \pi(\lambda_i, \epsilon_i \mid y_{1:i}, \lambda_{0:i-1}, \epsilon_{0:i-1})} \\ &= \frac{p(y_k \mid \lambda_{0:k}, y_{1:k-1}) p(\lambda_k \mid \epsilon_k) p(\epsilon_k \mid \epsilon_{k-1})}{\pi(\lambda_k, \epsilon_k \mid y_{1:k}, \lambda_{0:k-1}, \epsilon_{0:k-1})} \times \omega_{0:k-1}. \end{aligned} \tag{17}$$

In practice, after a few time steps, all but one particle will have nonzero weight. A procedure called resampling is required [5].

5. Tests

We test the introduced method in an idealised target tracking application, where we are able to observe horizontal coordinates of a MS at each time step, for example from a GPS receiver. The state consists of two-dimensional position x_k^{pos} and velocity vectors x_k^{vel}

$$x_k = \begin{bmatrix} x_k^{\text{pos}} \\ x_k^{\text{vel}} \end{bmatrix}.$$

The problem is modeled using a state-space model (1) – (3), with

$$F_k = \begin{bmatrix} I_2 & I_2 \\ \mathbf{0}_2 & I_2 \end{bmatrix}, \quad H_k = \begin{bmatrix} I_2 & \mathbf{0}_2 \end{bmatrix}. \tag{18}$$

I_n and $\mathbf{0}_n$ denote $n \times n$ identity and null matrices respectively. We use the constant velocity model

$$\mathsf{E}(w_k) = 0, \quad \mathsf{V}(w_k) = Q_k = 0.1^2 \cdot \begin{bmatrix} \frac{1}{3}I_2 & \frac{1}{2}I_2 \\ \frac{1}{2}I_2 & I_2 \end{bmatrix} \tag{19}$$

to describe the target motion [2]. As the observation model, we use the mixture distribution

$$p(y_k \mid x_k, \lambda_k) = \mathsf{N}\left(y_k \mid H_k x_k, (1-\lambda_k)5^2 I_2 + \lambda_k 25^2 I_2\right) \tag{20}$$

We use 100 runs of three different tests to test the performance of the introduced model. The evolution of the uncertainty parameter with time is changed with each test. This is illustrated in the Figure 3. In each test the averaged uncertainty is $\frac{1}{k_{\max}} \sum_{k=1}^{k_{\max}} \epsilon_k = 0.5$.

The RBPF algorithms with 50 particles using different models are compared. GMFs with model (5) and $\epsilon = 0.5$ (GMF_1) and model (6) with $\epsilon^{(1)} = 1 - \epsilon^{(2)} = 0.02$ (GMF_2) are compared to GMF employing the introduced hierarchical model for time-evolution of the model uncertainty pararameter ϵ_k (GMF_3). Tuning parameter $S = 100$ is fixed in all the tests.

The performance of the algorithms is compared through the average ability to estimate the parameter ϵ_k and root mean square error (RMSE) of the position coordinate estimates.

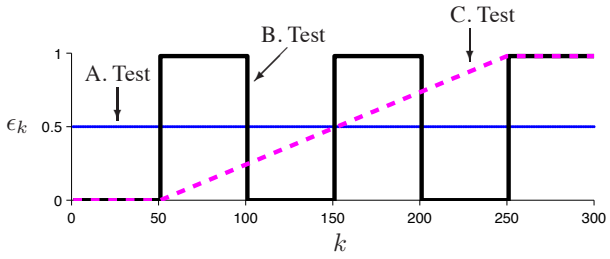

Figure 3: The evolution of the model uncertainty parameter with time in different tests.

A. Test

Observations are simulated from (20) with constant probability $p(\lambda_k = 0 \mid \epsilon_k) = 0.5$. GMF_1 is the filter based on the correct model. The results of the tests are reported in Figure 4. The RMSE performances of GMF_1 and GMF_3 are virtually identical, as is the estimations of ϵ_k. GMF_2 has the worst performance due to the small switching probabilities of λ_k.

B. Test

Observations are simulated from (20) with three changes of $\lambda_k = 0$ to $\lambda_{k+1} = 1$ and three changes of $\lambda_k = 1$ to $\lambda_{k+1} = 0$. GMF_2 is the filter based on model closest to the simulation model. The results of the tests are reported in Figure 5. The RMSE performance and the estimation accuracy of ϵ_k given by GMF_2 is the best. GMF_3 has an improved performance compared to GMF_1. The effect of the transition model for ϵ_k in GMF_3 algorithm can be seen in the estimation of ϵ_k as well as in the decrease of RMSE in time steps closer to the change points $k_c \in \{100, 200, 300\}$.

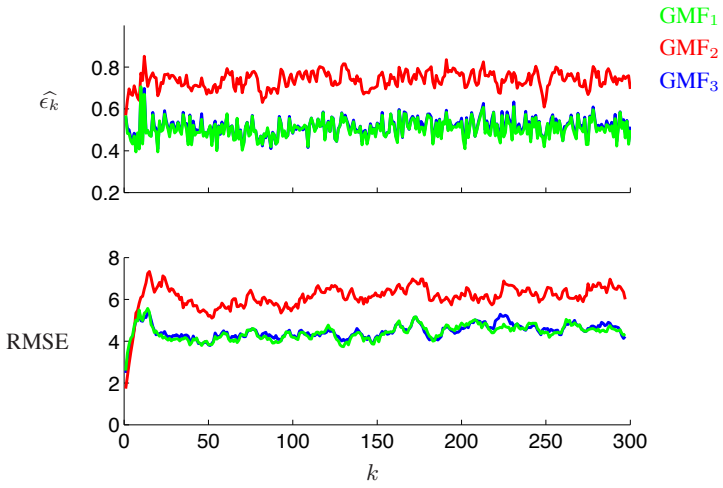

Figure 4: Results of A. Test.

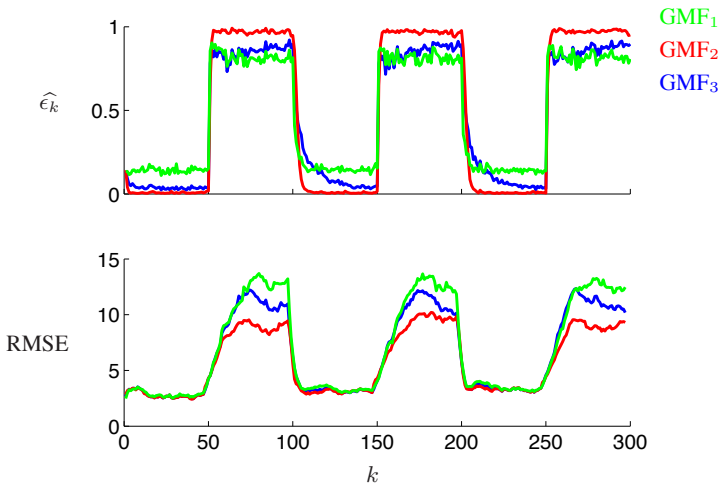

Figure 5: Results of B. Test.

C. Test

Observations are simulated from (20) such that there is a linear increase in the probability of $\lambda_k = 1$ in the interval $k \in [50, 250]$. The overall RMSE performance of GMF_3 is the best as is the estimation capability of ϵ_k. GMF_2 perform well when ϵ_k is close to 0 or 1 but has degraded performance otherwise, this is consistent with its performance in the previous tests.

6. Conclusions

A heuristic hierarchical model for the time-evolution of the model uncertainty parameter has been constructed and a RBPF-based algorithm for solving the resulting problem has been provided. Through simulations we showed that the new proposed method is able to approximate well the uncertainty parameter and it has RMSE performance comparable to

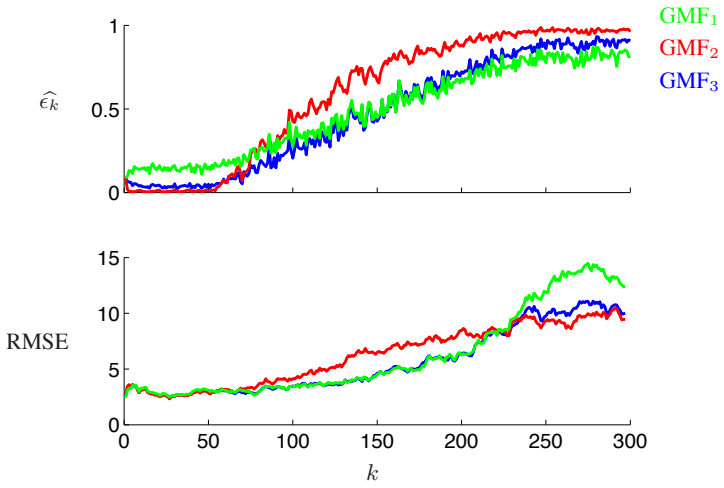

Figure 6: Results of C. Test.

the situations where an optimal model would be used to approximate the state.

It is expected that employing this additional level of hierarchy will be beneficial for applications where there the system behavior is governed by one of a set of models, with switching occurring only occasionally. Further study is required to expand the algorithm to handle more than two competing models.

References

[1] D. L. Alspach and H. W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, AC-17(4):439–448, August 1972.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[3] G.E.P. Box and G.C. Tiao. A Bayesian approach to some outlier problems. *Biometrika*, 55(1):119–129, 1968.

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

[5] A. Doucet, N. J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):2001, March 2001.

[6] A. H. Jazwinski. *Stochastic Processes And Filtering Theory*. Academic Press, Inc., 1970.

[7] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82, 1960.

[8] J. Marais and B. Godefroy. Analysis and optimal use of GNSS pseudo-range delays in urban canyons. In *Proceedings of IMACS Multiconference on Computational Engineering in System Applications (CESA), October 4-6, 2006, Beijing, China*, 2006.

[9] D. Peña and I. Guttman. Optimal collapsing of mixture distributions in robust recursive estimation. *Communications in Statistics: Theory and Methods*, 18(3):817–833, 1989.

[10] H.W. Sorenson and D.L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7:465–479, 1971.

PUBLICATION 5

Henri Pesonen: A framework for Bayesian receiver autonomous integrity monitoring in urban navigation. *NAVIGATION*, 58(3):229–240, 2011.

A Framework for Bayesian Receiver Autonomous Integrity Monitoring in Urban Navigation

HENRI PESONEN
Tampere University of Technology, Tampere, Finland

Received March 2010; Revised March 2011

ABSTRACT: *A Bayesian framework for RAIM designed to meet the requirements of today's urban navigation is introduced. The framework is general in the sense that when contaminated observations can be modeled, integrity information can be computed. We consider the case of positioning in the presence of multipath signals. The proposed method consists of a Bayesian filter that computes the posterior distribution of the history of the presence of multipath signals and the sizes of multipath biases, along with the navigation parameters. From the posterior distribution, optimal integrity information can be theoretically derived. For practical implementation, the optimal Bayesian filter is approximated with various techniques, and suboptimal integrity information is achieved. Simulations show that the approximations of the Bayesian RAIM framework have good performance compared to the traditional least squares RAIM, the drawback being the computational complexity of the methods.*

INTRODUCTION

The problem of determining and enhancing the reliability of the positioning service in different situations is one of the fundamental problems in the art of positioning. In the case of Global Navigation Satellite Systems (GNSS), the quality monitoring procedures are called integrity monitors [1, 2]. *The integrity is defined as a measure of trust which can be placed in the correctness of the information supplied by the total system. Integrity includes the ability of a system to provide timely warnings to the user when the system should not be used for intended operation* [3].

Historically, integrity monitoring procedures were mainly used in aviation, but now the positioning service provided by mobile phones, car navigators, etc., has become an everyday function for consumers and the quality of the positioning service needs to be monitored in more challenging scenarios. Traditional integrity monitoring was designed to detect a failure of a satellite, which is a very rare event, and the intended operation was, e.g., landing of an airplane, in which satellite failures and long intervals of lack of positioning service could cause severe problems. Nowadays as navigation in urban environments is much more common, integrity monitoring must be extended to accommodate poor satellite geometries and multipath signals in which the observation reaches the receiver along two or more paths causing signal quality to degrade and to be biased. Multipath is the most important source of error, generating errors up to 100 m without the receiver architecture being able to detect any problem in the signals [4].

The most common method of assessing the integrity is receiver autonomous integrity monitoring (RAIM). This is commonly based on statistical testing of the least squares residuals [1]. Using, e.g., a parity test, the presence of a defective observation within the set of observations is inspected. Given that the hypothesis of good quality observations is rejected, the erroneous observation is identified using the w-test [5] to test each observation separately. The observation that fails the test with greatest power of the test is discarded. Traditional RAIM is not suitable for urban positioning as it is based on the assumption of at most one defective observation, so at least five observations are needed to detect erroneous observation and six to identify it. Different, more or less heuristic methods have been developed to handle multiple failures [6, 7] based on repeating the process of identifying a single outlier multiple times. Another approach for monitoring the integrity of the positioning service is the quality control technique, which detects, identifies, and adapts itself to contaminated observations using hypothesis tests on the Kalman filter (KF) innovation process [8]. Techniques that use traditional RAIM for the extended Kalman filter (EKF) innovation process are also employed [9, 10].

NAVIGATION: *Journal of The Institute of Navigation*
Vol. 58, No. 3, Fall 2011
Printed in the U.S.A.

Another problem with RAIM methods based on frequent hypothesis tests is that they monitor integrity indirectly through the observations and minimal biases that can be detected in the observations. The Bayesian framework allows integrity of the positioning service to be monitored directly in the position domain [3, 11, 12].

The aim of this article is to describe a general Bayesian framework for integrity monitoring for personal navigation. The requirement for the framework is that it must work in an urban environment, but the performance requirements are not as strict as for aviation applications. The framework is general in the sense that after possible error sources in observations are modeled, Bayesian filtering is applied and the posterior probability distribution of the state is obtained. From this, we can in theory obtain optimal integrity information. As an example, we apply the Bayesian RAIM to the case in which multipath signals are the main cause of difficulty in positioning. We require from the algorithm that more than one multipath signal can be present and that there are no constraints on the satellite geometry or the number of visible satellites. The approach taken here is to model the multipath signals and their evolution and solve the joint posterior distribution of the quality of the signals and navigation parameters. From this posterior distribution containing all information about the navigation parameters and the quality of the signals, we can in theory infer the quality of the positioning performance optimally.

This paper is organized as follows. First, we give an overview of multipath mitigation strategies that are important in obtaining good quality positioning performance in the system. Next, we formulate the problem of joint detection/estimation of the multipath component as a state-space problem and present the optimal Bayesian framework for solving the filtering problem, including three methods to approximate the optimal posterior distribution. In the following sections we present the required navigation performance parameters and one of the traditional RAIM methods that is used to monitor that performance along with the Bayesian RAIM method. We compare the quality performance of the presented Bayesian RAIM method to that of traditional RAIM and also study the performance of the three presented approximating filters regarding the RAIM performance. Finally, we summarize the results and discuss some open questions.

MULTIPATH MITIGATION STRATEGIES

Reflections, attenuations, and blocked signals in urban environments have severe consequences on the positioning system performance. Reflected and attenuated signals have lower carrier-to-noise (C/N), causing the observations to be noisier than line-of-sight (LOS) signals and possibly have a time delay that can bias the observations [13]. The resulting combinations of reflected, attenuated, and possibly LOS signals are called multipath signals [2]. As the main task of a position system is to provide accurate and reliable positioning service, a variety of methods are used to enhance the system performance when multipath signals are present.

Hardware Multipath Mitigation

The most straightforward hardware multipath mitigation techniques use an array of antennas or a choke ring antenna designed to mitigate the multipath effects. Receiver correlators that are used to achieve code synchronization are another much researched multipath mitigation approach [4]. Narrowing the correlator reduces the bias, but these receivers can perform poorly in quickly changing environments. Also, a drawback with hardware based mitigation techniques is that they are bound to the chosen receiver architecture.

Statistical Hypothesis Testing

Traditional hardware-independent multipath mitigation techniques such as the fault detection and exclusion (FDE) that is often part of RAIM can be based on statistical hypothesis testing [1, 2, 10]. Techniques can be divided into *snapshot* and *filtering* techniques based on whether observations from only the current epoch are used, or past observations are also taken into account. Snapshot schemes are often based on the self-consistency check of the observations, using, e.g., least squares residuals, such as fault detection in RAIM [1], or on a check of the consistency of the position estimates given by two separate positioning systems [14]. In filtering systems, the same hypothesis tests used in snapshot methods are employed but instead of residuals, Kalman innovations are tested [8, 10, 15].

Generally, statistical hypothesis tests for multipath mitigation are based on the notion of the must frequent inference. In addition to these, Bayesian methods, which allow more versatile inspections of multipath signals and their effect on the position estimation, have also recently been investigated [3, 12, 16].

Robust Estimation Techniques

Robust estimation techniques are methods that automatically aim to desensitize the estimator against modeling errors. Most often modeling errors refer to outlying observations. In the snapshot situation, M-estimators [17, 18] have been used, and in time series problems, robust filters have been

employed [19, 20]. Other methods such as the residual weighting algorithm are also being developed [21]. Similar problems plague robust estimation techniques as those inherent in statistical hypothesis testing techniques. Snapshot techniques require at least five visible satellites and filtering techniques perform poorly with slowly growing errors. This is because small errors that are very difficult to detect can still bias the estimate to be consistent with the erroneous observation, causing the larger error in the next time step to be difficult to detect.

Joint Detection/Estimation of Multipath Biases

Recently, there has been increasing interest in filtering methods that detect and estimate the multipath biases simultaneously with the navigation parameters [22–24]. The methods are based on state-space models that include the effect of multipath biases to the pseudorange observations and the dynamic evolution of the biases in time. These enhanced, nonlinear state-models are used to estimate the navigation parameters and multipath biases using particle filters. We adopt this multipath mitigation strategy and describe it in more detail in the next section.

PROBLEM FORMULATION

We use the joint detection/estimation of the multipath biases with a Bayesian filter as our strategy for multipath mitigation [23, 24]. To approach the problem of Bayesian multipath mitigation filtering, we need a state-space model for the evolution of receiver position in time, and a model for the multipath pseudorange signals. Our model for these corrupted range measurements has an additive noise component and a bias component. If a LOS signal is present, the bias is close to zero [15]. The model is

$$y = ||p - s|| + b + \lambda(m + c) + v_{\text{atmo}} + v_{\text{svclock}} + v \quad (1)$$

where p and s are the position coordinates of the receiver and the satellite respectively, b is the receiver clock error in meters, boolean variable λ indicates the presence of the multipath signal, m is the mean of the NLOS multipath signal, c is the increased noise component involved with NLOS multipath signal, v_{atmo} is the error caused by ionospheric and atmospheric delays, v_{svclock} is the satellite clock error in meters and v is the receiver noise.

For this study we are mainly interested in the parameters related to multipath, as this is the error that varies greatly and quickly due to the environment of the receiver. Therefore, we assume that standard modeling of v_{atmo} and the correction of v_{svclock} based on broadcast GPS data is sufficient, and we ignore the nonwhite errors introduced by these terms.

The state-space model that we use for the GNSS pedestrian positioning problem is

$$\begin{aligned} x_{k+1} &= F_k x_k + w_k \\ y_k &= h_k(x_k, \lambda_k) + v_k \end{aligned} \quad (2)$$

which describes the linear evolution of the state and the nonlinear dependence of observations on the state. We model the observations as in Equation (1), so the state variable is

$$x_k = [p_k^T, \dot{p}_k^T, b_k, \dot{b}_k, m_k^T]^T \quad (3)$$

where $\dot{p}_k^T$ and $\dot{b}_k$ are the velocity and clock drift, respectively. The discrete state transition model can be derived by modeling the velocity and clock drift as Gauss Markov processes and integrating them over a time difference, $\Delta t_k = t_{k+1} - t_k$ [25, 26]. The model for the evolution of the clock bias is reasonable for short-term applications, as it does not take into account the discontinuities caused by the occasional resetting of the clock. The multipath bias is modeled as a random walk. The state transition matrix F_k is a block diagonal matrix as the evolution of different components are modeled as evolving independently of each other

$$F_k = \begin{bmatrix} I & \Delta t_k I & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t_k & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \quad (4)$$

The white noise processes for the state kinematics, w_k, nominal measurement noise, v_k, and increased noise due to multipath, c_k, are modeled as mutually independent Gaussian random variables with joint density

$$[w_k^T, v_k^T, c_k^T]^T \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q_k & 0 & 0 \\ 0 & \sigma^2 I_{n_k} & 0 \\ 0 & 0 & \sigma_c^2 I_n \end{bmatrix} \right) \quad (5)$$

where

$$Q_k = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & Q_2 & 0 \\ 0 & 0 & Q_3 \end{bmatrix} \quad (6)$$

and

$$\begin{aligned} Q_1 &= \begin{bmatrix} \frac{\Delta t_k^3}{3}\sigma_a^2 I_3 & \frac{\Delta t_k^2}{2}\sigma_a^2 I_3 \\ \frac{\Delta t_k^2}{2}\sigma_a^2 I_3 & \Delta t_k^2 \sigma_a^2 I_3 \end{bmatrix} \\ Q_2 &= \begin{bmatrix} \sigma_b^2 \Delta t_k + \frac{\Delta t_k^3}{3}\sigma_d^2 & \frac{\Delta t_k^2}{2}\sigma_d^2 \\ \frac{\Delta t_k^2}{2}\sigma_d^2 & \Delta t_k^2 \sigma_d^2 \end{bmatrix} \\ Q_3 &= \sigma_m^2 I \end{aligned} \quad (7)$$

The clock bias and drift noise variances, σ_b^2 and σ_d^2, depend on the quality of the receiver. The acceleration and multipath bias evolution variances, σ_a^2 and σ_m^2, depend on the environment and the motion of the receiver.

The observation noise variance, σ^2, represents the standard pseudorange error and σ_c^2 is the additional variance of noise caused by reflected and attenuated signals. As c_k and v_k are independent zero mean Gaussian random variables, the noise of the i^{th} observation is

$$\lambda_{k,i}c_{k,i} + v_{k,i} \sim \mathcal{N}(0, \lambda_{k,i}\sigma_c^2 + \sigma^2) \tag{8}$$

and coincides with the model in [15]. The boolean multipath indicator variable, $\lambda_{k,i}$, is modeled as a jump Markov process [24], with transition densities

$$\begin{aligned}
&\mathbb{P}(\lambda_{k+1,i} = 0|\lambda_{k,i} = 0) = p_{00},\\
&\mathbb{P}(\lambda_{k+1,i} = 1|\lambda_{k,i} = 0) = p_{10} = 1 - p_{00},\\
&\mathbb{P}(\lambda_{k+1,i} = 1|\lambda_{k,i} = 1) = p_{11},\\
&\mathbb{P}(\lambda_{k+1,i} = 0|\lambda_{k,i} = 1) = p_{01} = 1 - p_{11}
\end{aligned} \tag{9}$$

The transition densities depend on the positioning environment and on the user motion. After the problem is modeled, we are left with the task of estimating the state. In the next section we discuss the Bayesian filtering framework, which is in theory optimal for solving nonlinear filtering problems.

BAYESIAN FRAMEWORK

In the Bayesian framework, the state-space estimation problem (2) is solved by finding the posterior distribution, $p(x_k \mid y_{1:k})$, of the state x_k given all the observations, $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$, up until time k. In the problem of joint estimation of multipath biases and the state, the total probability theorem is used to find the posterior

$$\begin{aligned}
p(x_k|y_{1:k}) &= \sum_{i=1}^{2^{\Lambda_k}} p\left(x_k, \lambda_{1:k}^{(i)}|y_{1:k}\right)\\
&= \sum_{i=1}^{2^{\Lambda_k}} \mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right) p\left(x_k|y_{1:k}, \lambda_{1:k}^{(i)}\right)
\end{aligned} \tag{10}$$

which can be done in theory as the multipath indicator process histories, $\lambda_{1:k}^{(i)} \in \Lambda_k$, are part of a finite set

$$\Lambda_k = \{\lambda_{1:k}|\lambda_{i,j} \in \{0,1\}, i = 1, \ldots, n, j = 1, \ldots, k\} \tag{11}$$

However, the number of discrete sequences 2^{Λ_k} in the set Λ_k grows exponentially with time.

The posterior (10) is a mixture of distributions weighted with the probabilities of multipath indicator processes $\lambda_{1:k}^{(i)}$. These can be computed as

$$\begin{aligned}
&\mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right) \propto p\left(y_k|\lambda_{1:k}^{(i)}, y_{1:k-1}\right)\mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k-1}\right)\\
&= p\left(y_k|\lambda_{1:k}^{(i)}, y_{1:k-1}\right)\mathbb{P}\left(\lambda_k^{(i)}|\lambda_{1:k-1}^{(i)}, y_{1:k-1}\right)\\
&\quad \times \mathbb{P}\left(\lambda_{1:k-1}^{(i)}|y_{1:k-1}\right) = p\left(y_k|\lambda_{1:k}^{(i)}, y_{1:k-1}\right)\\
&\quad \times \mathbb{P}\left(\lambda_k^{(i)}|\lambda_{k-1}^{(i)}\right)\mathbb{P}\left(\lambda_{1:k-1}^{(i)}|y_{1:k-1}\right)
\end{aligned} \tag{12}$$

where $\propto$ denotes proportionality. Because of the term $p(y_k \mid \lambda_{1:k}^{(i)}, y_{1:k-1})$ in (12), conditioning with all the possible histories from the beginning is necessary to compute (10), even if λ_k is modeled as a Markov process.

Conditioned upon the multipath indicator history, we can compute the posterior of the state in two steps. The first step is the *prediction*:

$$\begin{aligned}
p\left(x_{k+1}|y_{1:k}, \lambda_{1:k+1}^{(i)}\right) &= p\left(x_{k+1}|y_{1:k}, \lambda_{1:k}^{(i)}\right)\\
&= \int p(x_{k+1}|x_k) p\left(x_k|y_{1:k}, \lambda_{1:k}^{(i)}\right) \mathrm{d}x_k
\end{aligned} \tag{13}$$

where we use the fact that x_{k+1} is a Markov process and it depends on $\lambda_{k+1}^{(i)}$ only with observation y_{k+1}. The second step is the *update*:

$$\begin{aligned}
&p\left(x_{k+1}|y_{1:k+1}, \lambda_{1:k+1}^{(i)}\right)\\
&\propto p\left(y_{k+1}|x_{k+1}, \lambda_{1:k+1}^{(i)}\right) p\left(x_{k+1}|y_{1:k}, \lambda_{1:k+1}^{(i)}\right)\\
&= p\left(y_{k+1}|x_{k+1}, \lambda_{k+1}^{(i)}\right) p\left(x_{k+1}|y_{1:k}, \lambda_{1:k+1}^{(i)}\right)
\end{aligned} \tag{14}$$

where the prediction distribution (13) is multiplied with the likelihood. The form of likelihood is obtained by using the fact that the observation depends only on the current state, x_{k+1}, and multipath indicator, $\lambda_{k+1}^{(i)}$. The Bayesian framework provides a conceptually optimal solution to the filtering problem. Considering the exponentially growing number of possible multipath indicator histories, for practical implementation we have to use approximate methods for the GNSS tracking/multipath estimation problem.

For approximating the posterior, we take into account that the pseudorange observation model can be well approximated as a linear model. Furthermore, conditioned on $\lambda_{k+1}^{(i)}$, the bias and noise components enter Eq. (1) linearly. Therefore the distributions (13) and (14) can be approximated as Gaussians obtained by the extended Kalman filter (EKF). Assume that at the k^{th} step, the distribution is

$$p\left(x_k|y_{1:k}, \lambda_{1:k}^{(i)}\right) = \mathcal{N}\left(\bar{x}_{k|k}^{(i)}, P_{k|k}^{(i)}\right) \tag{15}$$

where $\mathcal{N}\left(\bar{x}_{k|k}^{(i)}, P_{k|k}^{(i)}\right)$ is a normal distribution with mean $\bar{x}_{k|k}^{(i)}$ and covariance $P_{k|k}^{(i)}$. Then the predictive distribution is

$$p\left(x_{k+1}|y_{1:k}, \lambda_{1:k+1}^{(i)}\right) = \mathcal{N}\left(\bar{x}_{k+1|k}^{(i)}, P_{k+1|k}^{(i)}\right) \tag{16}$$

where

$$\begin{aligned} \bar{x}_{k+1|k}^{(i)} &= F_k \bar{x}_{k|k}^{(i)} \\ P_{k+1|k}^{(i)} &= F_k P_{k|k}^{(i)} F_k^T + Q_k \end{aligned} \tag{17}$$

The posterior distribution is approximated as

$$p\left(x_{k+1}|y_{1:k+1}, \lambda_{1:k+1}^{(i)}\right) = \mathcal{N}\left(\bar{x}_{k+1|k+1}^{(i)}, P_{k+1|k+1}^{(i)}\right) \tag{18}$$

where

$$\begin{aligned} \bar{x}_{k+1|k+1}^{(i)} &= \bar{x}_{k+1|k}^{(i)} + K_{k+1}^{(i)}\left(y_{k+1} - h_k\left(\bar{x}_{k+1|k}^{(i)}, \lambda_{k+1}^{(i)}\right)\right) \\ P_{k+1|k+1}^{(i)} &= P_{k+1|k}^{(i)} - K_{k+1}^{(i)} H_{k+1}^{(i)} P_{k+1|k}^{(i)} \end{aligned} \tag{19}$$

The matrix $H_{k+1}^{(i)}$ is the Jacobian of $h_k(x_k, \lambda_k^{(i)})$ around $\bar{x}_{k+1|k}^{(i)}$ and the Kalman gain is

$$K_{k+1}^{(i)} = P_{k+1|k}^{(i)} H_{k+1}^{(i)\,T}\left(H_{k+1}^{(i)} P_{k+1|k}^{(i)} H_{k+1}^{(i)\,T} + R_k^{(i)}\right)^{-1} \tag{20}$$

The observation covariance matrix $R_k^{(i)}$ is diagonal with j^{th} element, $\lambda_{k_j}^{(i)}\,\sigma_c^2 + \sigma^2$. The complete posterior distribution is

$$\begin{aligned} p(x_{k+1}|y_{1:k+1}) = \sum_{i=1}^{2^{\Lambda_{k+1}}} &\mathbb{P}\left(\lambda_{1:k+1}^{(i)}|y_{1:k+1}\right) \\ &\times \mathcal{N}\left(\bar{x}_{k+1|k+1}^{(i)}, P_{k+1|k+1}^{(i)}\right) \end{aligned} \tag{21}$$

which in theory can be computed using a bank of EKFs. The exponentially growing number ($2^{\Lambda_{k+1}}$) of required filters makes this impractical, and approximation techniques are required. In the following section we discuss a few methods used to approximate the posterior.

Multiple Model Filters

The problem of joint detection/estimation of multipath biases can be interpreted as a switching model problem, where $\lambda_k^{(i)}$, $i = 1,\ldots,2^n$ define the models at each epoch. Multiple model filters for switching models are among the most used methods for approximating the posterior (10) [25]. Multiple model filters, such as the interacting multiple model (IMM) algorithm are based on different strategies of merging the posterior components so that the number of filters required to approximate the posterior remains bounded, or even fixed. In IMM the number of filters is fixed to be 2^n, so that

$$\begin{aligned} p(x_k|y_{1:k}) &= \sum_{i=1}^{2^n} \mathbb{P}\left(\lambda_k^{(i)}|y_{1:k}\right) p\left(x_k|y_{1:k}, \lambda_k^{(i)}\right) \\ &= \sum_{i=1}^{2^n} \mathbb{P}\left(\lambda_k^{(i)}|y_{1:k}\right) \frac{p\left(y_k|\lambda_k^{(i)}, x_k\right)}{p\left(y_k|\lambda_k^{(i)}, y_{1:k-1}\right)} \cdot \\ &\qquad \times p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}\right) \end{aligned} \tag{22}$$

IMM is based on approximating the Gaussian sum distribution

$$\begin{aligned} &p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}\right) \\ &= \sum_{j=1}^{2^n} p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}, \lambda_{k-1}^{(j)}\right) \mathbb{P}\left(\lambda_{k-1}^{(j)}|\lambda_k^{(i)}, y_{1:k-1}\right) \\ &\propto \sum_{j=1}^{2^n} p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}, \lambda_{k-1}^{(j)}\right) \mathbb{P}\left(\lambda_k^{(i)}|\lambda_{k-1}^{(j)}, y_{1:k-1}\right) \\ &\qquad \times \mathbb{P}\left(\lambda_{k-1}^{(j)}|y_{1:k-1}\right) \\ &= \sum_{j=1}^{2^n} p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}, \lambda_{k-1}^{(j)}\right) \mathbb{P}\left(\lambda_k^{(i)}|\lambda_{k-1}^{(j)}\right) \mathbb{P}\left(\lambda_{k-1}^{(j)}|y_{1:k-1}\right) \end{aligned} \tag{23}$$

with a single Gaussian using moment matching. Assuming that

$$p\left(x_k|y_{1:k-1}, \lambda_k^{(i)}, \lambda_{k-1}^{(j)}\right) = \mathcal{N}\left(\bar{x}_{k|k-1}^{(i,j)}, P_{k|k-1}^{(i,j)}\right)$$

and $w^{(i,j)} = \mathbb{P}\left(\lambda_k^{(i)}|\lambda_{k-1}^{(j)}, y_{1:k-1}\right)\mathbb{P}\left(\lambda_{k-1}^{(j)}|y_{1:k-1}\right)$, the mean and variance of (23) are

$$\begin{aligned} \bar{x}_{k|k-1}^{(i)} &= \sum_{j=1}^{2^n} w^{(i,j)} \bar{x}_{k|k-1}^{(i,j)} \\ P_{k|k-1}^{(i)} &= \sum_{j=1}^{2^n} w^{(i,j)}\left(P_{k|k-1}^{(i,j)} + e_k^{(i,j)} e_k^{(i,j)^T}\right) \end{aligned} \tag{24}$$

where $e_k^{(i,j)} = \bar{x}_{k|k-1}^{(i)} - \bar{x}_{k|k-1}^{(i,j)}$. More details on the derivation of the IMM algorithm can be found in [25].

Particle Filters

A general method for solving nonlinear filtering problems such as the GNSS tracking problem with detection/estimation would be to use particle filters (PFs) [27, 28] where the posterior distribution is approximated empirically with a set of N support points, $\{(x_{0:k}^{(i)}, \lambda_{1:k}^{(i)}): i = 1,\ldots, N\}$, and associated normalized weights, $\{\omega_{1:k}^{(i)}: i = 1,\ldots, N\}$. The empirical estimate of the distribution is

$$\widehat{p_N}(x_{0:k}, \lambda_{1:k}|y_{1:k}) = \sum_{i=1}^{N} \omega_{1:k}^{(i)} \delta((x_{0:k}, \lambda_{1:k}) - (x_{0:k}^{(i)}, \lambda_{1:k}^{(i)})) \tag{25}$$

In practice, sequential importance sampling is used to generate the empirical distribution. Weights, $\omega_{1:k}^{(i)}$, are assigned to samples drawn from an importance distribution, $\pi(x_{0:k}, \lambda_{1:k}|y_{1:k})$, from which it is convenient to draw samples, and are given by

$$\omega_{1:k}^{(i)} \propto \frac{p(x_{0:k}^{(i)}, \lambda_{1:k}^{(i)}|y_{1:k})}{\pi(x_{0:k}^{(i)}, \lambda_{1:k}^{(i)}|y_{1:k})}$$

Given the linear Gaussian state model and the transition probabilities for the multipath indicators, it is easy to sample from $p(x_k, \lambda_k | x_{k-1}^{(i)}, \lambda_{k-1}^{(i)})$. Because of this, we choose importance distribution that can be factored as

$$\begin{aligned}\pi(x_{0:k}, \lambda_{1:k}|y_{1:k}) &= p(x_{0:k}, \lambda_{1:k}|y_{1:k-1}) \\ &= p(x_k, \lambda_k|x_{k-1}, \lambda_{k-1}) p(x_{0:k-1}, \lambda_{1:k-1}|y_{1:k-1})\end{aligned}$$

Now, new samples $(x_k^{(i)}, \lambda_k^{(i)})$ are obtained by propagating the old samples $(x_{k-1}^{(i)}, \lambda_{k-1}^{(i)})$ according to the distribution $p(x_k, \lambda_k | x_{k-1}^{(i)}, \lambda_{k-1}^{(i)})$. The associated weights are obtained by updating them as

$$\omega_k^{(i)} \propto \omega_{k-1}^{(i)} p\left(y_k | x_k^{(i)}, \lambda_k^{(i)}\right) \tag{26}$$

Sequential importance sampling suffers from a condition called degeneracy, whereby eventually all but one particle will have negligible weight. Resampling is used to reduce the effect of degeneracy of the samples. In resampling, we compare the approximate effective sample size

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} \left(\omega_k^{(i)}\right)^2} \tag{27}$$

to some threshold N_T. If $N_{\text{eff}} < N_T$, then we generate a new set of samples $(x_k^{(i)*}, \lambda_k^{(i)*})$ from the old ones $(x_k^{(i)}, \lambda_k^{(i)})$, by sampling from a distribution

$$\mathbb{P}\left(\left(x_k^{(i)*}, \lambda_k^{(i)*}\right) = \left(x_k^{(i)}, \lambda_k^{(i)}\right)\right) = \omega_k^{(i)} \tag{28}$$

and setting equal weights for the new samples.

Rao-Blackwellized Particle Filter

As previously mentioned, the GNSS tracking problem (2) is a nearly linear problem if the multipath indicators would be known. Therefore a variance reduction method, known as Rao-Blackwellization, could be used to improve the performance of the PF [24, 29]. The resulting filter is referred to as the marginalized, or Rao-Blackwellized particle filter (RBPF).

The RBPF takes advantage of the fact that $p(x_k | y_{1:k}, \lambda_{1:k}^{(i)})$ is well approximated as a Gaussian distribution with mean and covariance given by the EKF. Therefore using RBPF we only need to draw a sample from $\mathbb{P}(\lambda_{1:k}|y_{1:k})$ to estimate the posterior. Because it is awkward to sample directly from $\mathbb{P}(\lambda_{1:k}|y_{1:k})$, we use sequential importance sampling. As it is easy to draw samples from $\mathbb{P}\left(\lambda_k|\lambda_{k-1}^{(i)}\right)$, we use the prior distribution as the importance distribution

$$\Pi(\lambda_{1:k}|y_{1:k}) = \mathbb{P}(\lambda_k|\lambda_{k-1})\mathbb{P}(\lambda_{1:k-1}|y_{1:k-1}) \tag{29}$$

The weights, $\omega_k^{(i)}$, can be recursively computed as

$$\omega_k^{(i)} \propto \omega_{k-1}^{(i)} p\left(y_k|y_{1:k-1}, \lambda_{1:k}^{(i)}\right) \tag{30}$$

Resampling is applied analogously to the ordinary particle filter.

The resulting RBPF is very similar to the multiple model filters such as IMM but instead of merging the Gaussian distributions given by the bank of EKFs, the most probable EKFs defined by the indicator processes are selected automatically by the propagation of the samples.

BAYESIAN RAIM

The positioning performance is often monitored using RAIM, which is a hardware independent method that monitors the positioning performance based on the received observations. RAIM is based on statistical hypothesis testing. There are several different RAIM methods, such as least squares, parity, and range comparison, which have been shown to be essentially the same methods. However, the way RAIM is traditionally carried out imposes some restrictions on when it can be applied:

- The assumption of at most one biased observation is too strict for urban navigation where multipath signals are common.
- The assumption that biased signals do not contain any useful information about the position is also too strict.
- The predicted availability of RAIM is traditionally based only on the geometry and number of visible satellites, and not on the received observations. The geometry and number of visible satellites can be very poor in urban environments where large parts of the sky are blocked from view, and generally it is difficult to predict future visibility of satellites in urban environments.

- Separate algorithms for positioning, error detection, and error identification complicate the receiver architecture [3].

In the previous section we formulated a filter that solves the posterior distribution of the navigation parameters, x_k, jointly with the parameters describing the multipath propagation given all the current and previous observations. From this probability distribution we can in theory infer the full position performance in an optimal way.

Required Navigation Performance

The main difference between traditional and Bayesian integrity monitoring is that in the Bayesian framework the position is a random variable with a probability distribution $p(x_k \mid y_{1:k})$ and we can infer information directly about the position instead of doing it indirectly from the measurements and test statistics derived from them. The performance of RAIM is traditionally monitored through the required navigation performance (RNP) parameters of accuracy, availability, continuity and integrity, which are defined for aviation purposes [30]. Also, traditionally, availability and continuity parameters are predictive parameters which use information about the orbits of the satellites. For urban navigation, we should not assume future visibility of any of the satellites. Therefore we use different definitions for the availability and continuity parameters.

Accuracy

The accuracy parameter can be found from the probability distribution of the position. Defining

$$\Phi_k(T) = \mathbb{P}(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) \tag{31}$$

where

$$\Omega_T(\hat{x}_k) = \{x_{k,1:2} | \|x_{k,1:2} - \hat{x}_{k,1:2}\| \leq T\} \tag{32}$$

and $\hat{x}_k$ is the position estimate, e.g., the posterior mean. The accuracy of the positioning system is

$$T = \Phi_k^{-1}(\alpha) \tag{33}$$

where α a prespecified number that is the required probability to be contained in the disk of radius T. The precise evaluation of accuracy can be computationally complex, but a value that is good enough for practical purposes can be obtained by approximating the posterior distribution with a simpler one. Often the actual value of the accuracy parameter is not of interest, but rather whether the estimate is accurate enough. If required accuracy of, e.g., T meters is defined on α-level, then the system is accurate enough if

$$\mathbb{P}(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) \geq \alpha \tag{34}$$

Integrity

The integrity is the RNP parameter most directly linked to the safety of the operation. In the traditional RAIM it is defined as P_{MI} and is based solely on the test statistic constructed from the observations, not on the actual positioning error. In the Bayesian framework, we can in theory compute exactly the probability that the position error is larger than a specified alarm limit (AL). Therefore the definition of integrity is directly applicable, and more general than in the least squares RAIM (LS RAIM). The probability that the position error is larger than AL is

$$\begin{aligned} P_{\mathrm{AL}} &= \mathbb{P}(x_k \notin \Omega_{\mathrm{AL}}(\hat{x}_k)|y_{1:k}) = 1 - \int_{\Omega_{\mathrm{AL}}(\hat{x}_k)} p(x_k|y_{1:k})\mathrm{d}x_k \\ &= 1 - \sum_{i=1}^{2^{\Lambda_k}} \mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right) \int_{\Omega_{\mathrm{AL}}(\hat{x}_k)} p\left(x_k|y_{1:k}, \lambda_{1:k}^{(i)}\right)\mathrm{d}x_k \\ &= \sum_{i=1}^{2^{\Lambda_k}} \mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right) \mathbb{P}\left(x_k \notin \Omega_{\mathrm{AL}}(\hat{x}_k)|y_{1:k}, \lambda_{1:k}^{(i)}\right) \end{aligned} \tag{35}$$

where

$$\Omega_{\mathrm{AL}}(\hat{x}_k) = \{x_{k,1:2} | \|x_{k,1:2} - \hat{x}_{k,1:2}\| \leq \mathrm{AL}\} \tag{36}$$

One has to define a maximum integrity risk, P_0, for the probability, P_{AL}, that one is willing to accept. If $P_{\mathrm{AL}} \leq P_0$, we make a decision to not warn about a possibly too large position error. Now the integrity risk, or probability of misleading information, P_{MI}, is equal to, P_{AL}, and the probability of the system operating within tolerable limits is $1 - P_{\mathrm{AL}}$. On the other hand, if $P_{\mathrm{AL}} > P_0$ we make a decision to warn the user about the possibility of the error being too large. Given this decision, the probability of the error actually being within tolerable limits, or the probability of false alarm, P_{FA}, is equal to $1 - P_{\mathrm{AL}}$. Now the probability of true alert is P_{AL}.

Availability

Availability in aviation RAIM requirements is the probability that we are able to compute our position with the required accuracy and integrity at any instant in time and at any location in the coverage area. It is a predictive parameter, meaning that it is computed without using actual observations.

However, for personal navigation purposes we are more interested in whether the positioning service is available now in our location. The posterior distribution of the position can be obtained at each epoch, even when *no* observations are available, or when the geometry of the satellites is extremely poor. When no observations are available, $y_{k+1} = \emptyset$, the state is still propagated with the motion model at each epoch. If the posterior distribution is at k^{th} epoch of the form

$$p(x_k|y_{1:k}) = \sum_{i=1}^{m} \alpha_i \mathcal{N}\left(\bar{x}_{k|k}^{(i)}, P_{k|k}^{(i)}\right) \tag{37}$$

then at the $(k+1)^{\text{th}}$ epoch, the posterior is

$$\begin{aligned} p(x_{k+1}|y_{1:k+1}) &= p(x_{k+1}|y_{1:k}) \\ &= \sum_{i=1}^{m} \alpha_i \mathcal{N}\left(F_k \bar{x}_{k|k}^{(i)}, F_k P_{k|k}^{(i)} F_k^T + Q_k\right) \end{aligned} \tag{38}$$

and so on. We are thus able to monitor accuracy and integrity 100% of the time and consequently, the availability coincides with the probability that we are able to determine the position with the required accuracy and integrity. Now the positioning system is available with probability 1 if

$$\begin{aligned} &\mathbb{P}(x_k \in \Omega_T(\hat{x}_k)|y_{1:k}) > \alpha \quad \text{and} \\ &\mathbb{P}(x_k \in \Omega_{\text{AL}}(\hat{x}_k)|y_{1:k}) < P_0 \end{aligned} \tag{39}$$

Continuity

Operation is interrupted when insufficient accuracy or integrity is detected. So the system has continuity of operation for Δ future epochs if

$$\begin{aligned} &\mathbb{P}(x_{k+\delta} \in \Omega_T(\hat{x}_{k+\delta})|y_{1:k}) > \alpha \quad \text{and} \\ &\mathbb{P}(x_{k+\delta} \in \Omega_{\text{AL}}(\hat{x}_{k+\delta})|y_{1:k}) < P_0 \end{aligned} \tag{40}$$

for $\delta = 1,\ldots,\Delta$.

Bayesian RAIM Algorithm

The probability (35) contains all the integrity information that can possibly be obtained by the system defined by the models. The probability of any imaginable combination of faulty observations occurring in time causing an unacceptable error can be found in $\mathbb{P}\left(x_k \notin \Omega_{\text{AL}}(\hat{x}_k)|y_{1:k}, \lambda_{1:k}^{(i)}\right), i = 1,\ldots,2^{\Lambda_k}$. The influence that this probability has on the overall integrity is contained in the probability of a unique history of faulty observations, $\mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right)$, $i = 1,\ldots,2^{\Lambda_k}$. The probabilities $\mathbb{P}\left(\lambda_{1:k}^{(i)}|y_{1:k}\right)$ are a measure of how well a faulty observation history has been identified as the 'true' history. The marginal probability, $\mathbb{P}(\lambda_k|y_{1:k})$, can be used to identify the current corrupted observations, and the marginal $\mathbb{P}\left(\lambda_{k,j}|y_{1:k}\right)$ is the probability that the j^{th} observation is now corrupted.

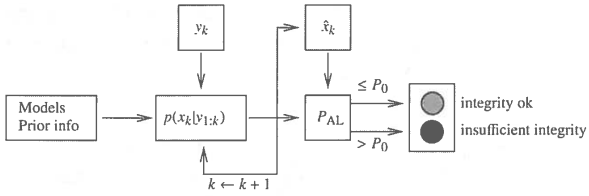

Fig. 1–Diagram of the bayesian RAIM algorithm

As the integrity is the most important safety parameter, in the proposed Bayesian RAIM the decision of whether or not the position estimate should be trusted is based on the value of the computed P_{AL}. The basic Bayesian RAIM architecture is depicted in Figure 1.

Bayesian RAIM has the following benefits compared to traditional RAIM. The process is conceptually much simpler, as all information about the positioning, fault detection, fault identification, and bias estimation is derived from the posterior distribution of the navigation parameters. Also, no restrictions on the geometry, number of visible satellites, or number of faulty observations are imposed. Bias values do not have to be fixed, or bounded, to obtain certain performance. And in theory, all the results are optimal, of course conditioned on the used models.

Unfortunately the exact computation of the posterior distribution, and therefore integrity, is impossible in reality in any but the most trivial cases. Therefore, the integrity computations have to be carried out using approximating Bayesian filters such as the ones discussed earlier.

Numerical Integration

The computations of the integrity and accuracy involve the integration of the posterior distribution over $\Omega_{\text{AL}}(\hat{x}_k)$ and $\Omega_{\text{T}}(\hat{x}_k)$, respectively. For the posterior approximated by PF we can compute integrity analytically as

$$\begin{aligned} 1 - P_{\text{MI}} &= \int_{\Omega_{\text{AL}}(\hat{x}_k)} \sum_{i=1}^{N} \omega_k^{(i)} \delta((x_k, \lambda_k) - (x_k^{(i)}, \lambda_k^{(i)}))\mathrm{d}x_k \\ &= \sum_{i \in J} \omega_k^{(i)} \end{aligned} \tag{41}$$

where $J = \{i : x_{k,1:2}^{(i)} \in \Omega_{\text{AL}}(\hat{x}_k)\}$, and analogously for the integral over $\Omega_{\text{T}}(\hat{x}_k)$. Posterior distributions

approximated by IMM and RBPF are Gaussian sum distributions (GS). These cannot be integrated analytically over $\Omega_{\mathrm{AL}}(\hat{x}_k)$, but instead have to rely on approximations. There are two approaches for approximating the integral. First, one can use numerical integration methods, i.e., quadrature or cubature rules that approximate the integral as a weighted sum of integrand evaluations at a set of nodes. Second, one can approximate the integration problem with a simpler one. We can approximate the integration region, or an integrand with something that we can analytically, or at least very accurately, evaluate. In this work, we compute numerical approximations.

Approximating P_{MI} we note from (35) that the integral can be expressed as a weighted integral of a Gaussian over a disk, $\Omega_{\mathrm{AL}}(\hat{x}_k)$. The integral of a Gaussian, $\mathcal{N}(\mu, \Sigma)$, over an origin-centered disk, $\Omega_{\mathrm{r}}(0)$, can be expressed as [31]

$$\int_{\Omega_r} N(\mu, \Sigma)\mathrm{d}x = \int_{-r}^{r} \int_{-\sqrt{r^2-x_1^2}}^{\sqrt{r^2-x_1^2}} \times \mathcal{N}(\mu_{2|1}, \Sigma_{22|1})\mathrm{d}x_2 \mathcal{N}(\mu_1, \Sigma_{11})\mathrm{d}x_1 \quad (42)$$

where

$$\begin{aligned} \mu_{2|1} &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_1 - \mu_1) \\ \Sigma_{22|1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{aligned} \quad (43)$$

Given that we can use efficient algorithms for the error function to compute

$$F(x_1) = \int_{-\sqrt{r^2-x_1^2}}^{\sqrt{r^2-x_1^2}} \mathcal{N}(\mu_{2|1}, \Sigma_{22|1})\mathrm{d}x_2$$

we are left with approximating the integral

$$\int_{-r}^{r} F(x_1)\mathcal{N}(\mu_1, \Sigma_{11})\mathrm{d}x_1 \quad (44)$$

The integral (44) can be approximated using quadrature rules, such as Romberg integration [32], or adaptive Lobatto quadrature [33]. Even if an efficient quadrature rule is employed, the computational requirements can turn out to be significant if all of the components of the GS posterior are integrated over $\Omega_{\mathrm{AL}}(\hat{x}_k)$. Fortunately, often most of the probability of the GS posterior is contained in a few of the components and this allows us to use a recursive evaluation of P_{AL} which can be stopped when a sufficient amount of probability is contained inside or outside of $\Omega_{\mathrm{AL}}(\hat{x}_k)$. The recursive process is described in Algorithm, and is proved in the Appendix.

Algorithm 1: Evaluation of sufficient P_{AL} for GS posteriors of the form $p(x_k|y_{1:k}) = \sum_{i=1}^{m} \alpha_i \mathcal{N}\left(\bar{x}_{k|k}^{(i)}, P_{k|k}^{(i)}\right)$

Find indexing J s.t. $\alpha_{j_i} \geq \alpha_{J_{i+1}}$
for $i = 1,2,\ldots$ **do**
  $I_{J_i} = \int_{\Omega_{\mathrm{AL}}(\hat{x}_k)} \mathcal{N}\left(\bar{x}_{k|k}^{(J_i)}, P_{k|k}^{(J_i)}\right)\mathrm{d}x_k$
  $P_{\mathrm{AL},1:i} = P_{\mathrm{AL},1:i-1} + \alpha_{J_i} I_{J_i}$
  $\alpha_{1:i} = \alpha_{1:i-1} + \alpha_{J_i}$
  if $P_0 \geq 1 - P_{\mathrm{AL},1:i}$ **then**
    Integrity ok
    break
  else if $P_0 < \alpha_{1:i} - P_{\mathrm{AL},1:i}$ **then**
    Insufficient integrity
    break
  end
end

SIMULATIONS

We study the performance of Bayesian RAIM implemented with PFs, RBPFs and IMM in satellite positioning simulations. We denote the filters with N particles as PF_N and RBPF_N. We simulate 100 different 600 s long pedestrian walks simulated with the motion model decribed in the previous section. Constellations of six satellites that provide pseudorange observations are generated. We include a multipath bias to two low-elevation satellites as illustrated by Figure 2. The parameters

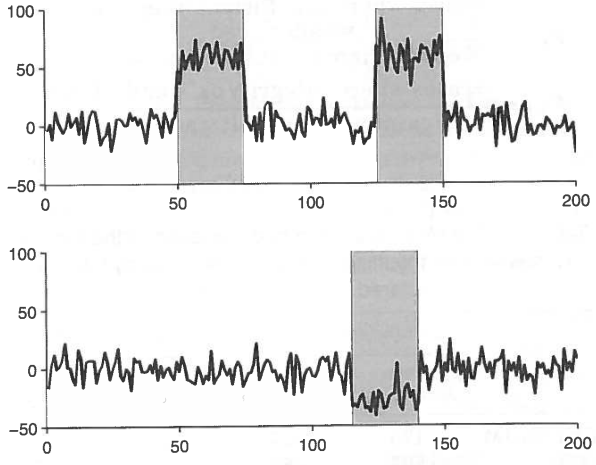

Fig. 2–Simulated observation noise contaminated with multipath is $v_{k,i}^{\lambda_{k,i}} \sim \mathcal{N}(\lambda_{k,i} m_{k,i}, \sigma^2 + \lambda_{k,i} 5^2)$. Multipath bias is generated randomly, $m_{k,i} \sim Uni[-50, 150]$, and is kept constant for the duration of one test track.

Table 1—Simulation Parameters

$\sigma^2_{a,true} = 1$	$p_{00} = 0.8$
$\sigma^2_b = .01$	$p_{11} = 0.8$
$\sigma^2_d = .01$	
$\sigma^2 = 100$	
$\sigma^2_c = 25$	

Table 2—Simulation Results of the First Test

	$\hat{P}_{FA}$	$\hat{P}_{MI}$	Time
EKF/RAIM	0.17	0.83	1
IMM	0.03	0.56	66
$RBPF_{64}$	0.10	0.31	53
$RBPF_{256}$	0.12	0.26	204
PF_{10^5}	0.08	0.56	447

used to simulate the data are given in Table 1. In addition to Bayesian RAIM methods, we implement EKF that performs traditional least squares RAIM fault detection and identification/exclusion at each epoch. All the filters use $\sigma^2_a = 25$ and $\sigma^2_m = 10$, as we assume that true values of these given in Table 1 are unknown. As the performance requirements, we use E911 requirements. The current version of wireless E911 requires that 67% and 95% of emergency callers are positioned within 50 m and 150 m, respectively [34].

We monitor the performance in two cases. In the first case we have the performance requirement of 67% probability being within 50 m and in the second case 95% probability being within 150 m. The performance of the simulations are given in Tables 2 and 4, where we report the performance of $\hat{P}_{FA}$ and $\hat{P}_{MI}$, which are defined as

$$\hat{P}_{FA} = \frac{\#\text{cases where 'insufficient integrity' was declared}}{\#\text{cases where no fault was present}}$$
$$\hat{P}_{MI} = \frac{\#\text{cases where 'integrity ok' was declared}}{\#\text{cases where a fault was present}}. \quad (45)$$

Table 3—The Mean and Standard Deviation of the Error in Cases Where 'Insufficient Integrity' or 'Integrity Ok' was Declared in the First Testcase

	Insufficient Integrity		Integrity Ok	
	Mean (error)(m)	SD (error)(m)	Mean (error)(m)	SD (error)(m)
EKF/RAIM	173	327	177	345
IMM	595	687	68	95
$RBPF_{64}$	177	251	30	30
$RBPF_{256}$	155	180	28	32
PF_{10^5}	241	267	74	137

Table 4—Simulation Results of the Second Test

	$\hat{P}_{FA}$	$\hat{P}_{MI}$	Time
EKF/RAIM	0.10	0.90	1
IMM	0.01	0.68	192
$RBPF_{64}$	0.09	0.46	292
$RBPF_{256}$	0.09	0.39	1139
PF_{10^5}	0.04	0.83	851

The presence of a fault is interpreted as a case where the error of the estimate is larger than the acceptable error, which is 50 m or 150 m in the first and second tests, respectively. In addition we report the relative computation time in the 'time' column. This value is only intended as a rough indication, as computing time can vary widely depending on the particular implementation of the algorithm.

In Tables 2 and 3 are the results of the first test. The false alarm rates reported in Table 2 are better in all Bayesian methods than with EKF/RAIM by more than 0.05; in addition, misleading information rates are much better than with the traditional method applied to this problem. However, required computation times are much longer with Bayesian methods. In Table 3 we report the statistics of the estimation errors when the system is in 'insufficient integrity' or 'integrity ok' states, which respectively refer to the states when a failure is declared or not. When in 'insufficient integrity' state, the traditional method and Bayesian methods have mostly similar performance. However, when in the 'integrity ok' state, Bayesian methods clearly have better performance.

Tables 4 and 5 present the results of the second test. The false alarm rates reported in Table 4 show the Bayesian methods similar to EKF/RAIM, but again misleading information rates are much better than with the traditional method applied to this problem. However, required computation times with Bayesian methods are even longer than in the first test. This is due to the numerical integration

Table 5—The Mean and Standard Deviation of the Error in Cases Where 'Insufficient Integrity' or 'Integrity Ok' was Declared in the Second Test

	Insufficient Integrity		Integrity Ok	
	Mean (error)(m)	SD (error)(m)	Mean (error)(m)	SD (error)(m)
EKF/RAIM	181	377	196	416
IMM	1264	1369	127	219
$RBPF_{64}$	215	269	48	74
$RBPF_{256}$	263	533	45	94
PF_{10^5}	445	1802	192	1280

procedure. As the probability masses are contained in smaller regions inside the integration region, this causes problems for the numerical procedures. In Table 5 we report the statistics of the estimation errors when the system is in 'insufficient integrity' or 'integrity ok' states. The larger allowed error size can be seen in the mean errors compared to the first test.

CONCLUSIONS

In this paper we presented a basic framework for Bayesian integrity monitoring based on Bayesian filtering and posterior distributions of the position error. Bayesian RAIM is general in that the posterior distribution can always be computed when we can construct a model for the positioning problem and from it, optimal integrity information can be computed in theory. In the current work, an integrity monitor was constructed for urban navigation and it was based on the joint estimation of multipath signals and the navigation parameters. Estimation of multipath signals was based on the presented simple dynamic model.

A drawback of Bayesian RAIM is that without simple models or heavy approximations to the posterior distribution, Bayesian RAIM is much more computationally demanding than traditional RAIM, and so it currently may not be feasible for consumer-grade GPS receivers. We presented a few approximation methods which were compared to the traditional RAIM method in simulated test cases. The Bayesian method performed well but computationally all the approximations were much more demanding than the traditional method. Therefore, more research on approximate methods or simpler models is required. Currently Bayesian RAIM could be used as a baseline for RAIM algorithms as there exists approximation methods that provide nearly optimal integrity information when considerable computation resources are available.

Other issues related to the Bayesian RAIM are the requirement for the prior distributions and state transition models, although snapshot Bayesian RAIM could also be used. The modeling of prior information is not by any means impossible, as in the case of little knowledge of the parameters, prior distributions should reflect this, e.g., priors with large variances would be employed. However, development of better models is still an important research topic, as are better approximations to the posterior distribution.

ACKNOWLEDGMENTS

The comments by anonymous reviewers and professor Robert Piché are gratefully acknowledged.

REFERENCES

1. Brown, R. G., "A Baseline GPS RAIM Scheme and a Note on the Equivalence of Three RAIM Methods," *NAVIGATION*, Vol. 39, No. 3, Fall 1992, pp. 301–316.
2. Kaplan, E. D., Ed., *Understanding GPS: Principles and Applications*, Artech House, Inc., 1996.
3. Ober, P. B., *Integrity Prediction and Monitoring of Navigation Systems*, Ph.D. Thesis, Teknische Universiteit Delft, 2003.
4. Soubielle, J., Fijalkow, I., Duvaut, P., Bibaut, A., "GPS Positioning in a Multipath Environment," *IEEE Transactions on Signal Processing*, Vol. 50, No. 1, January 2002, pp. 141–150.
5. Baarda, W., *A Testing Procedure for Use in Geodetic Networks*, Netherlands Geodetic Commission, Publication on Geodesy, New Series 2, No. 5, Delft, Netherlands, 1968.
6. Hewitson, S. and Wang, J., "GNSS Receiver Autonomous Integrity Monitoring (RAIM) for Multiple Outliers," *The European Journal of Navigation*, Vol. 4, No. 4, 2006, pp. 47–57.
7. Kuusniemi, H., *User-Level Reliability and Quality Monitoring in Satellite-Based Personal Navigation*, Ph.D. Thesis, Tampere University of Technology, 2005.
8. Teunissen, P. J. G., "Quality Control in Integrated Navigation Systems," *IEEE Transactions on Aerospace Electronic Systems*, Vol. 5, 2006, pp. 35–41.
9. Wang, J., Xu, C., and Wang, J., "Applications of Robust Kalman Filtering Schemes in GNSS Navigation," *Proceedings of the International Symposium on GPS/GNSS*, Yokohama, Japan, November 2008, pp. 308–316.
10. Hewitson, S. and Wang, J., "Impact of Dynamic Information on GNSS Receiver Integrity Monitoring," *Proceedings of the International Symposium on GPS/GNSS*, Sydney, Australia, December 2004.
11. Ober, P. B., "Integrity According to Bayes," *Position Location and Navigation Symposium*, IEEE 2000, San Diego, CA, 2000.
12. Pesonen, H., "Outlier-Robust Bayesian Filter with Integrity Monitoring for GNSS Positioning," *Proceedings of ENC-GNSS 2009*, Naples, Italy, 2009.
13. Marais, J. and Godefroy, B., "Analysis and Optimal Use of GNSS Pseudo-Range Delays in Urban Canyons," *Proceedings of IMACS Multiconference on Computational Engineering in System Applications (CESA)*, Beijing, China, October 4–6, 2006.
14. Hwang, P. Y. and Brown, R. G., "From RAIM to NIORAIM - A New Integrity Approach to Integrated Multi-GNSS Systems," *Inside GNSS*, May/June, 2008.
15. Spangenberg, M., Tourneret, J.-Y., Calmettes, V., and Duchâteau, G., "Detection of Variance Changes and Mean Value Jumps in Measurement Noise for Multipath Mitigation in Urban Canyons," *Proceedings of Asilomar*, 2008.
16. Pesonen, H. and Piche, R., "Bayesian Receiver Autonomous Integrity Monitoring Technique," *Proceedings of the 2009 International Technical Meeting of The Institute of Navigation*, Anaheim, CA, January 2009, pp. 420–425.

17. Huber, P. J., *Robust Statistics*, John Wiley & Sons, Inc., 1981.
18. Chang, X.-W. and Guo, Y., "Huber's M-Estimation in GPS Positioning: Computational Aspects," *Proceedings of the 2004 National Technical Meeting of The Institute of Navigation*, San Diego, CA, January 2004, pp. 829–839.
19. Masreliez, C. J. and Martin, R. D., "Robust Bayesian Estimation for the Linear Model and Robustifying the Kalman Filter," *IEEE Transactions on Automatic Control*, Vol. 22, No. 3, 1977, pp. 361–371.
20. Pesonen, H., "Robust Estimation Techniques for GNSS Positioning," *Proceedings of NAV07-The Navigation Conference and Exhibition*, London, England, October 31–November 1, 2007.
21. Chen, P., "A Non-Line-of-Sight Error Mitigation Algorithm in Location Estimation," *Proceedings of IEEE Wireless Communications and Networking Conference*, Vol. 1, 1999, pp. 316–320.
22. Huerta, J. M., Vidal, J., Giremus, A., and Tourneret, J.-Y., "Joint Particle Filter and UKF Position Tracking in Severe Non-Line-of-Sight Situations," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 3, No. 5, October 2009.
23. Giremus, A. and Tourneret, J.-Y., "Joint Detection/Estimation of Multipath Effects for the Global Positioning System," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, PA, March 2005, pp. 17–20.
24. Giremus, A., Tourneret, J., and Calmettes, V., "A Particle Filtering Approach for Joint Detection/Estimation of Multipath Effects on GPS Measurements," *IEEE Transactions on Signal Processing*, Vol. 55, No. 4, 2007, pp. 1275–1285.
25. Bar-Shalom, Y., Li, X. R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, Inc., 2001.
26. Farrell, J. and Barth, M., *The Global Positioning System & Inertial Navigation*, McGraw-Hill, Inc., 1998.
27. Doucet, A., Godsill, S., and Andrieu, C., "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, Vol. 10, 2000, pp. 197–208.
28. Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, February 2002, pp. 174–188.
29. Doucet, A., Gordon, N. J., and Krishnamurthy, V., "Particle Filters for State Estimation of Jump Markov Linear Systems," *IEEE Transactions on Signal Processing*, Vol. 50, No. 3, March 2001.
30. Tiemeyer, B., *Performance Evaluation of Satellite Navigation and Safety Case Development*, Ph.D. Thesis, Universität der Bundeswehr Munchen, 2002.
31. Mardia, K. V., Kent, J. T., and Bibby, J. M., *Multivariate Analysis*, Academic Press, Inc., (London) Ltd., 1979.
32. Kythe, P. K. and Schäferkotter, M. R., *Handbook of Computational Methods for Integration*, Chapman & Hall/CRC Press, 2005.
33. Gander, W. and Gautchi, W., "Adaptive Quadrature – Revisited," *BIT Numerical Mathematics*, Vol. 40, No. 1, March 2000, pp. 84–101.
34. Hatfield, D. N., "A Report on Technical and Operational Issues Impacting the Provision of Wireless Enhanced 911 Services," Technical report, Federal Communications Commission, 2002.

APPENDIX

PROOF OF ALGORITHM 1

Assume that the posterior distribution is of the form $p(x_k|y_{1:k}) = \sum_{i=1}^{m} \alpha_i \mathcal{N}\left(\bar{x}_{k|k}^{(i)}, P_{k|k}^{(i)}\right)$ and the threshold for P_{AL} is P_0. For notational simplicity we assume that the components of GS are ordered based on the weights α_i, so that $\alpha_i \geq \alpha_{i+1}$.

At the j^{th} step of the algorithm, we have computed the integrals $I_1,\ldots,I_j$, so that $\sum_{i=1}^{j} \alpha_j I_j$ is the probability mass inside $\Omega_{\mathrm{AL}}(\hat{x}_k)$. We do not know the values of $I_{j+1},\ldots,I_m$, but we know the weights, $\alpha_{j+1},\ldots,\alpha_n$. The best, or the worst case scenario is that $\alpha_{j+1} + \cdots + \alpha_n$ amount of probability is inside, or outside of $\Omega_{\mathrm{AL}}(\hat{x}_k)$. Combining the information, we compute that, if in the best possible case the probability of being inside $\Omega_{\mathrm{AL}}(\hat{x}_k)$ is less than $1 - P_0$,

$$\sum_{i=1}^{j} \alpha_j I_j + \sum_{k=j+1}^{n} \alpha_k = \sum_{i=1}^{j} \alpha_j I_j + 1 - \sum_{k=1}^{j} \alpha_k$$

$$= 1 - \sum_{i=1}^{j} \alpha_i (I_i - 1) < 1 - P_0$$

$$\Leftrightarrow P_0 < \sum_{i=1}^{j} \alpha_i (1 - I_1) = \sum_{i=1}^{j} \alpha_i - \sum_{i=1}^{j} \alpha_i I_i$$

we can stop the recursive computation as sufficient integrity cannot be obtained. Similarly, if in the worst possible case the probability of being outside $\Omega_{\mathrm{AL}}(\hat{x}_k)$ is less than P_0, then

$$\sum_{i=1}^{j} \alpha_i (1 - I_i) + \sum_{k=j+1}^{n} \alpha_k = \sum_{i=1}^{j} \alpha_i (1 - I_i) + 1 - \sum_{k=1}^{j} \alpha_k$$

$$= 1 - \sum_{i=1}^{j} \alpha_i I_i < P_0,$$

we can stop the recursive computation as sufficient integrity has already been obtained.

PUBLICATION 6

Henri Pesonen and Robert Piché: Bayesian fault diagnosis method for linear systems with additive sensor errors. In *2nd International Conference and Exhibition on Ubiquitous Positioning, Indoor Navigation, and Location Based Service UPINLBS 2012*, Helsinki, October 3–4, 2012.

Bayesian Fault Detection Method for Linear Systems with Outliers

Henri Pesonen and Robert Piché
Tampere University of Technology
Korkeakoulunkatu 1, 33720 Tampere

***Abstract*—A novel approach for monitoring the accuracy of the Bayesian estimate of linear Gaussian state-space model is introduced, based on the monitoring of the propagation of the errors in the Kalman filter algorithm. The effect of the sensor errors on the Kalman filter estimate is explicitly computed and compensated. A marginalized particle filter is used to compute the posterior distribution of the sensor errors. Using a target tracking simulation it is shown that the proposed method has improved performance over the standard detection-identification-adaptation (DIA) method**

***Index Terms*—Bayesian filtering, marginalized particle filtering, fault diagnosis, jump detection, change detection, fault monitoring, Kalman filter, DIA**

I. Introduction

Abrupt changes in linear dynamic systems are often of significant interest as they can provide essential information about the processes, or possibly cause major degeneracy of the state estimator if the changes of the system go undetected. For example, in clinical trials, changes in the system can be caused by biological events which are of paramount importance to analyze [1]. In positioning and tracking systems, the changes in the environment or maneuvers cause the system to provide biased position estimates which can lead to hazardous situations [2].

In positioning systems, fault detection methods are usually referred to as receiver autonomous integrity monitoring (RAIM) methods [3]. The traditional RAIM methods first perform fault detection based on a statistical test for the consistency of the observation vector. If the test fails, statistical tests are performed on each of the observations in order to identify and remove the faulty observation [4]. The diagnosis is usually carried out at each time step by testing the bias of each of the observations separately, but there have been studies of integrity and quality monitoring methods in time series data [5], [6], [7].

The first author thanks TUT graduate school for financial support.

We model the abrupt changes, or errors, in the system as suddenly appearing or disappearing additive components in the sensor model. In positioning systems, such errors could be caused by multipath or non-line-of-sight-signals, or sensor malfunctions [8]. The detection of these kind of changes has traditionally been performed with generalized likelihood ratio (GLR), or almost analogous detection-identification-adaptation (DIA) method, and CUSUM algorithms [9], [5], [10], [11]. These methods are based on monitoring the innovation process of a Kalman filter (KF) that does not take into account the abrupt changes. Another approach is to approximate the joint posterior distribution of the state and the abrupt changes using multiple model filtering [12], or sequential Monte Carlo methods [13], [2]. From the resulting posterior distribution one can solve for any quality measure of the chosen estimator. However, quality measures based is the posterior distribution can be quite sensitive to the probabilities on the tails of the posterior which can be poorly estimated by sampling based methods.

We propose a change detection method which combines the two approaches. Instead of solving the joint distribution of the state and the changes, we compute the joint distribution of the changes and the KF estimate error caused by the additive changes. One benefit of this approach is that the detection procedure is separate from the state estimator and can be applied as a separate module to any system estimated by KF.

The paper is organized as follows. In Section II we describe the state-space model for the system and for the additive sensor errors. In Section III we present a Bayesian approximative solution for the problem which employes marginalized particle filter, which can be used due to the special property of the proposed model. In Section IV we describe our novel approach for fault diagnosis of the nominal Kalman filter and in Section V we compare the presented approaches in a simple positioning problem. In Section VI we conclude our study.

II. Problem formulation

We consider a discrete time stochastic system with additive unknown changes

$$x_{k+1} = F_k x_k + w_k \tag{1}$$
$$y_k = H_k x_k + v_k + s_k. \tag{2}$$
$$x_0 \sim \mathsf{N}\left(x_{0|0}, P_{0|0}\right), \tag{3}$$

where $x_k \in \mathsf{R}^{n_x}$ is the state vector, $y_k \in \mathsf{R}^{n_y}$ is the observation, $w_k \sim \mathsf{N}(0, Q_k)$ and $v_k \sim \mathsf{N}(0, R_k)$ are mutually independent, zero mean Gaussian noise processes. s_k is the additional error process in the sensor model. $\mathsf{N}(\mu, \Sigma)$ is a Gaussian distribution with mean μ and covariance Σ.

The Bayesian filtering framework can be used to compute the posterior distribution $p(x_k \mid y_{1:k}, s_{1:k})$, where $y_{1:k} \triangleq [y_1, \ldots, y_k]$, with the *prediction* step

$$\begin{aligned} &p(x_k \mid y_{1:k-1}, s_{1:k-1}) \\ &= \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid y_{1:k-1}, s_{1:k-1}) \mathrm{d}x_{k-1} \end{aligned} \tag{4}$$

and the *update* step

$$\begin{aligned} &p(x_k \mid y_{1:k}, s_{1:k}) \\ &\propto p(y_k \mid x_k, s_k) p(x_k \mid y_{1:k-1}, s_{1:k-1}). \end{aligned} \tag{5}$$

It is well known that with the given model (1) – (3) and known $s_{1:k}$ the posterior distribution is a Gaussian distribution $p(x_k | y_{1:k}, s_{1:k}) = \mathsf{N}\left(x_{k|k}, P_{k|k}\right)$ with mean and covariance computed recursively by the KF algorithm

$$x_{k+1|k} = F_k x_{k|k} \tag{6}$$
$$P_{k+1|k} = F_k P_{k|k} F_k^T + Q_k \tag{7}$$
$$z_k = y_k - H_k x_{k|k-1} - s_k \tag{8}$$
$$S_k = H_k P_{k|k-1} H_k^T + R_k \tag{9}$$
$$K_k = P_{k|k-1} H_k^T S_k^{-1} \tag{10}$$
$$x_{k|k} = x_{k|k-1} + K_k z_k \tag{11}$$
$$P_{k|k} = (I_{n_x} - K_k H_k) P_{k|k-1}, \tag{12}$$

where I_n is a $n \times n$ identity matrix, z_k is the innovation and S_k is the innovation covariance. The additive errors s_k are however usually unknown and, as they have a linear influence on the state estimates, they can significantly degrade the performance of KF algorithm.

We model the additive errors s_k as a Gaussian Markov process depending on a Markov chain $\lambda_k \in \{0, 1\}$ with

$$\mathsf{P}(\lambda_{k+1} = j \mid \lambda_k = i) = p_{ij} \tag{13}$$

which is the switch probability between ith and jth models at time $k+1$. Modeling of p_{ij} can be often an extremely complicated task, as it can be dependent on the time k, and also on the value of x_k [14].

Also the size of the errors is a difficult to model as the cause of the errors is often unknown, and assuming the value of the sensor error to be e.g. constant in time, is quite a strong assumption. Therefore we consider additive errors as jump Markov linear system

$$s_{k,i} = \lambda_{k,i} \epsilon_{k,i}, \tag{14}$$

where $s_{k,i}$ is the ith element of s_k at kth timestep and $\epsilon_k \sim \mathsf{N}(0, R_k^\epsilon)$ is Gaussian white noise process independent of the stochastic processes in (1)–(3). Using this model we estimate the value of $e_{k,i}$ independent of the estimated $e_{k-1,i}$. Note that

$$v_k + s_k \sim \mathsf{N}(0, R_k + \Lambda_k R_k^\epsilon \Lambda_k) = \mathsf{N}(0, R_k(\Lambda_k)), \tag{15}$$

where $\Lambda_k \triangleq \mathrm{diag}(\lambda_{k,1}, \ldots, \lambda_{k,n_y})$. This model is often used to describe outliers in the observations [15] and it is, in a sense, a conservative model for the sensor errors. Often errors are modeled as a constant, or slowly evolving, bias for consecutive time steps [2]. However, if nothing is known about the dynamic nature of the error, the assumption of constant bias may degrade the performance of the estimator significantly. On the other hand, if we assume that the size of the bias may change freely from a time step to the next, we may be able to estimate it better in the case when it is not constant or slowly evolving.

III. Marginalized particle filtering

The posterior distribution for the model introduced in the previous section is

$$p(x_k, \Lambda_{1:k} \,|\, y_{1:k}) = p(\Lambda_{1:k} \mid y_{1:k}) p(x_k \mid y_{1:k}, \Lambda_{1:k}). \tag{16}$$

The indicator variable history $\Lambda_{1:k}$ is a discrete random variable with a finite number (n^{y_k}) of possible values and probability mass function

$$p(\Lambda_{1:k} \mid y_{1:k}) = \sum_{i=1}^{n_y^k} \mathsf{P}\left(\Lambda_{1:k}^{(i)} \mid y_{1:k}\right) \delta\left(\Lambda_{1:k} - \Lambda_{1:k}^{(i)}\right),$$

where $\mathsf{P}\left(\Lambda_{1:k}^{(i)} \mid y_{1:k}\right)$ is a shorthand notation for the probability $\mathsf{P}\left(\Lambda_{1:k} = \Lambda_{1:k}^{(i)} \mid y_{1:k}\right)$. The marginal distribution of the state is

$$p(x_k \mid y_{1:k}) = \sum_{i=1}^{n_y^k} \mathsf{P}\left(\Lambda_{1:k}^{(i)} \mid y_{1:k}\right) p\left(x_k \mid y_{1:k}, \Lambda_{1:k}^{(i)}\right),$$

which can be computed with a bank of KFs [16], [17]. The sum goes over all possible $\Lambda_{1:k}$ and thus the exact

solution is computationally intractable for even small k. Several approximative techniques have been applied for this problem, e.g. pruning or merging the Gaussian components [12], [18].

We apply sequential Monte Carlo estimation for approximating the posterior [19]. Because the part $p(x_k|y_{1:k}, \Lambda_{1:k})$ of the posterior can be solved analytically, it is possible to approximate only the marginal distribution $p(\Lambda_{1:k} \mid y_{1:k})$ and thus decrease the variance of the empirical approximative posterior [13]. The resulting estimation method is the marginalized particle filter (MPF). Here are the details.

We approximate $p(\Lambda_{1:k}|y_{1:k})$ empirically with N samples

$$p(\Lambda_{1:k} \mid y_{1:k}) \approx \sum_{j=1}^{N} \omega_{1:k}^{(j)} \delta\left(\Lambda_{1:k} - \Lambda_{1:k}^{(j)}\right), \tag{17}$$

where

$$\omega_{1:k}^{(j)} \propto \frac{\mathsf{P}\left(\Lambda_{1:k}^{(j)} \mid y_{1:k}\right)}{\pi\left(\Lambda_{1:k}^{(j)} \mid y_{1:k}\right)}. \tag{18}$$

The importance sampling distribution $\pi\left(\Lambda_{1:k} \mid y_{1:k}\right)$ can be chosen freely within certain requirements but the choice

$$\pi(\Lambda_{1:k} \mid y_{1:k}) = \pi\left(\Lambda_1\right) \prod_{j=1}^{k} \pi\left(\Lambda_j \mid \Lambda_{j-1}\right), \tag{19}$$

enables recursive updating of weights $\omega_{1:k}$ according to

$$\omega_{1:k}^{(j)} \propto \omega_{1:k-1}^{(j)} p\left(y_k \mid y_{1:k-1}, \Lambda_{1:k}^{(j)}\right). \tag{20}$$

The latter term is evaluated by sampling $\Lambda_k^{(j)}$ from (19), adding it to $\Lambda_{1:k-1}^{(j)}$ and evaluating

$$\begin{aligned} & p\left(y_k \mid y_{1:k-1}, \Lambda_{1:k}^{(j)}\right) \\ &= \int p\left(y_k \mid x_k, \Lambda_k^{(j)}\right) p\left(x_k \mid y_{1:k-1}, \Lambda_{1:k}^{(j)}\right) \mathrm{d}x_k \\ &= \frac{1}{\sqrt{\det(2\pi S_k(\Lambda_{1:k}^{(j)}))}} e^{-\frac{1}{2} z_k(\Lambda_{1:k}^{(j)})^T S_k(\Lambda_{1:k}^{(j)}) z_k(\Lambda_{1:k}^{(j)})}, \end{aligned} \tag{21}$$

where $z_k(\Lambda_{1:k}^{(j)})$ and $S_k(\Lambda_{1:k}^{(j)})$ are the innovation and innovation covariance given the history $\Lambda_{1:k}^{(j)}$ and are computed in (8) and (9).

The approximative posterior distribution is now

$$\hat{p}(x_k \mid y_{1:k}) = \sum_{j=1}^{N} \omega_{1:k}^{(j)} \mathsf{N}\left(x_{k|k}(\Lambda_{1:k}^{(j)}), P_{k|k}(\Lambda_{1:k}^{(j)})\right), \tag{22}$$

where (11) and (12) are computed given $\Lambda_{1:k}^{(j)}$. In practice we need to occasionally *resample* the Gaussian mixture components to prevent degeneracy of the weights of the approximative distribution. In the resampling procedure, the components with large weights are duplicated and used to replace components with small weights if the effective sample size N_{eff} drops lower than some threshold value [19]. The effective sample size can be approximated as

$$N_{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N} \left(\omega_{1:k}^{(i)}\right)^2}. \tag{23}$$

IV. Nominal system fault diagnosis

Many of the classic change detection and quality monitoring algorithms, such as GLR method [9] and detection-identification-adaptation (DIA) method [5], are based on the innovation of the nominal KF. The nominal KF is run with the assumption that $\Lambda_i = 0$ for all $i \geq 1$. Due to the recursive nature of the KF algorithm, the error propagates according to Lemma 1.

Lemma 1. *Let the state space model be described by* (1)–(3)*. The influence of the realized additive error sequence $s_{1:k}$ on the Kalman innovation* (8) *and the posterior mean* (11) *can be expressed explicitly as*

$$z_k = z_k(0_{1:k}) + \Delta z_k \tag{24}$$

and

$$x_{k|k} = x_{k|k}(0_{1:k}) + \Delta x_{k|k} \tag{25}$$

where

$$z_k^0 = y_k - H_k x_{k|k-1}(0_{1:k}) \tag{26}$$

The sequences Δz_k and $\Delta x_{k|k}$ can be expressed recursively as

$$\Delta z_k = s_k - H_k F_{k-1} \Delta x_{k-1|k-1} \tag{27}$$

and

$$\Delta x_{k|k} = K_k s_k + C_k \Delta x_{k-1|k-1}, \tag{28}$$

where $C_k = (I_{n_x} - K_k H_k) F_{k-1}$.

Proof: Analogous to the proof of Lemma 5 in [20]. ∎

Instead of solving the marginal distribution of the state, we compute the posterior distribution of the additive errors and the error of the nominal KF estimator. Lemma 1 describes the evolution of the influence of the additive errors on the state, the innovation, and the KF estimator. Using the lemma, the quality monitoring

procedure is formulated as a linear system with white noise processes as process uncertainty:

$$\begin{bmatrix} s_{k+1} \\ \Delta x_{k|k} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ K_k & C_k \end{bmatrix} \begin{bmatrix} s_k \\ \Delta x_{k-1|k-1} \end{bmatrix} + \begin{bmatrix} \Lambda_{k+1}\epsilon_{k+1} \\ 0 \end{bmatrix} \tag{29}$$

$$z_k = \begin{bmatrix} I_{n_y} & -H_k F_{k-1} \end{bmatrix} \begin{bmatrix} s_k \\ \Delta x_{k-1|k-1} \end{bmatrix} + z_k^0, \tag{30}$$

$$\begin{bmatrix} s_0 \\ \Delta x_{0|0} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{31}$$

where $z_k(0_{1:k}) \sim N(0, S_k(0_{1:k}))$ is a white noise process independent of s_k and $\Delta x_{k|k}$. The system is observed through (24). We can filter the system (29) – (31) in the Bayesian framework and obtain the posterior

$$\begin{aligned} &p(s_k, \Delta x_{k|k} \mid z_{1:k}) \\ &= \sum_{j=1}^{n_y^k} \mathsf{P}\left(\Lambda_{1:k}^{(j)} \mid z_{1:k}\right) p(s_k, \Delta x_{k|k} \mid z_{1:k}, \Lambda_{1:k}). \end{aligned} \tag{32}$$

The approximative distribution is computed using MPF analogously to the previous section. The resulting distribution is

$$\begin{aligned} &\hat{p}(s_k, \Delta x_{k|k} \mid z_{1:k}) \\ &= \sum_{j=1}^{N} \omega_{1:k}^{(j)} \mathsf{N}\left(\begin{bmatrix} s_{k|k}(\Lambda_{1:k}^{(j)}) \\ \Delta x_{k|k}(\Lambda_{1:k}^{(j)}) \end{bmatrix}, \Sigma_{k|k}(\Lambda_{1:k}^{(j)})\right), \end{aligned} \tag{33}$$

where the mean and the covariance are computed applying the KF algorithm to the linear Gaussian system (29) – (31) given $\Lambda_{1:k}^{(j)}$. It can be shown that given $\Lambda_{1:k}$ the weights for the approximative distributions of the previous section and of the introduced model are the same, i.e.

$$\mathsf{P}\left(\Lambda_{1:k} \mid y_{1:k}\right) = \mathsf{P}\left(\Lambda_{1:k} \mid z_{1:k}\right). \tag{34}$$

The posterior filtering distribution $\mathsf{P}\left(\Lambda_k \mid z_{1:k}\right)$ is the probability of an error being present in the observation, and can be used to determine the quality of the observation vector. If $\mathsf{P}\left(\lambda_{k,i} = 1 \mid z_{1:k}\right) > 0.5$, then it is more probable that the error is present than not, given the whole observation history. In addition, to determining whether an error is present, we are able to monitor the size of the cumulative effect $\Delta x_{k|k}$ of the sensor errors $s_{1:k}$. We use the mean of $\Delta x_{k|k}$ as the estimate of the sensor error size, and using this estimate we can compute a corrected estimate $\tilde{x}_{k|k}$ using the filter estimate $x_{k|k}$ with the estimated error $\Delta x_{k|k}$

$$\tilde{x}_{k|k} = x_{k|k} - \Delta x_{k|k}. \tag{35}$$

The quality monitoring method is illustrated in Figure 1.

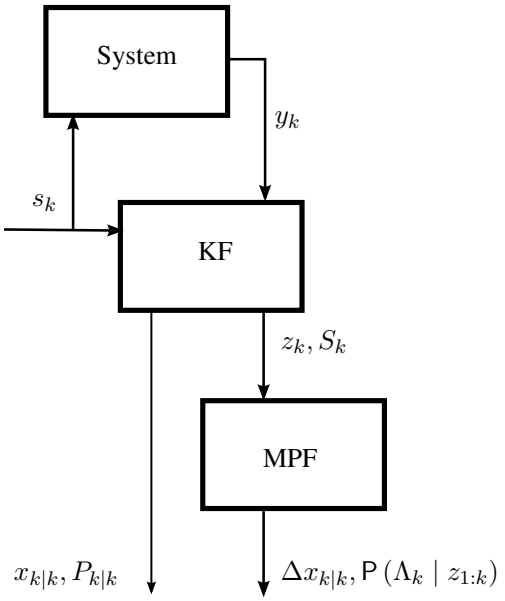

Fig. 1. Quality monitoring of the nominal KF

V. Simulations

We test the discussed estimation method, referred to as nominal system fault detection (NSFD), in a simulation of a simple two-dimensional positioning tracking problem in urban environment. At each time step we receive the position coordinates of the receiver as the measurement. The state consists of two position and two velocity coordinates, and the motion of the target (1) is modeled with the constant velocity model [12]

$$F_k = \begin{bmatrix} I_2 & I_2 \\ 0_2 & I_2 \end{bmatrix}, \quad Q_k = 0.1^2 \begin{bmatrix} \frac{1}{3}I_2 & \frac{1}{2}I_2 \\ \frac{1}{2}I_2 & I_2 \end{bmatrix}. \tag{36}$$

The nominal measurement model is (2) with

$$H_k = \begin{bmatrix} I_2 & 0_2 \end{bmatrix}, \quad R_k = \begin{bmatrix} 7^2 & 3^2 \\ 3^2 & 8^2 \end{bmatrix}. \tag{37}$$

We simulate 1000 tracks with 300 time steps. In time interval $k \in [101, 200]$ we simulate outliers in the additive sensor error with model (14), using $p_{00} = p_{11} = 0.9$ and $\epsilon_k \sim \mathsf{N}\left(0, 30^2 I_2\right)$. We run the MPFs with $N = 25$ particles, and use effective sample size threshold $0.6 \cdot N$ in the resampling procedure. The mean value of the approximative posterior distribution is used as the estimate. A DIA method that tests the presence of bias at each time step with test statistic threshold $T_{\mathrm{DIA}} = 5$ is used as a comparison.

The simulation results are illustrated with Table I. The correlation coefficient of the true error of the nominal KF and the estimated error $\Delta x_{k|k}$ given by NSFD is 0.77. Using the estimated $\Delta x_{k|k}$ to evaluate a new state

estimate $\tilde{x}_{k|k}$ in (35), the RMSE of the KF 5.43 is lowered to 4.38. This is an improvement to the RMSE 5.11 of DIA method. The errors are computed in the two-dimensional position coordinates, i.e. the first two dimensions of the state vector.

The failure detection power of NSFD and DIA is compared by the ability to detect presence of the bias in observations. This is quantified with the frequency of false alarms (type I error), i.e. determining that there is an error present when there is not, and missed alarms (type II error), i.e. determining that there is no bias, when in reality there is. NSFD is better than DIA with respect to both type I and II errors. However, although NSFD has improved performance of DIA with respect to error detection power and RMSE performance, NSFD is computationally more demanding. The current implementation of NSFD takes roughly N times more time than DIA, where N is the number of particles in (33).

	NSFD	DIA	KF
Type I error	0.04	0.11	
Type II error	0.18	0.26	
RMSE	4.38	5.11	5.43

TABLE I
PERFORMANCE COMPARISON OF ERROR DETECTION POWER AND RMSE OF NSFD AND DIA METHODS.

VI. CONCLUSIONS

A new novel fault detection approach NSFD has been proposed for linear Gaussian state-space models, and marginalized particle filtering solution has been provided for solving the resulting problem. The method was tested against standard DIA method using positioning simulations, and NSFD has better performance with respect to the fault detection power and RMSE. However, the improved performance is achieved at the cost of more demanding computational requirements.

REFERENCES

[1] A. F. M. Smith and M. West, "Monitoring renal transplants: An application of the multiprocess Kalman filter," *Biometrics*, vol. 39, pp. 867–878, December 1983.

[2] A. Giremus, J. Tourneret, and V. Calmettes, "A particle filtering approach for joint detection/estimation of multipath effects on GPS measurements," *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1275–1285, April 2007.

[3] R. G. Brown, "A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods," *NAVIGATION : Journal of Institute of Navigation*, vol. 39, no. 3, pp. 101–116, 1992.

[4] W. Baarda, *A Testing Procedure for Use in Geodetic Networks*, Netherlands Geodetic Commission, Publication on Geodesy, New Series 2, No. 5, Delft, Netherlands, 1968.

[5] P. J. G. Teunissen, "Quality control in integrated navigation systems," *IEEE Transactions on Aerosp. Electr. System*, vol. 5, pp. 35–41, July 1990.

[6] M. Spangenberg, J.-Y. Tourneret, V. Calmettes, and G. Duchâteau, "Detection of variance changes and mean value jumps in measurement noise for multipath mitigation in urban canyons," in *Proceedings of Asilomar, 2008*, 2008.

[7] H. Pesonen, "A framework for Bayesian receiver autonomous integrity monitoring in urban navigation," *NAVIGATION : Journal of Institute of Navigation*, vol. 58, pp. 229–240, 2011.

[8] L. Cong and W. Zhuang, "Nonline-of-sight error mitigation in mobile location," *IEEE Transactions of wireless communications*, vol. 4, no. 2, pp. 560–573, March 2005.

[9] A. S. Willsky and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 1, pp. 108–121, 1976.

[10] M. Basseville and I. V. Nikiforov, *Detection of abrupt changes: theory and applications*, Information and system science series. Prentice Hall, NJ, 1993.

[11] F. Gustafsson, *Adaptive Filtering and Change Detection*, John Wiley & Sons Ltd., 2000.

[12] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, Inc., 2001.

[13] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 2001, March 2001.

[14] H. Pesonen and R. Piché, "Bayesian positioning using Gaussian mixture models with time-varying component weights.," in *Proceedings of 2011 Joint Statistical Meetings (JSM), Miami Beach, FL, July–August, 2011*, 2011.

[15] D. Peña and I. Guttman, "Optimal collapsing of mixture distributions in robust recursive estimation," *Communications in Statistics: Theory and Methods*, vol. 18, no. 3, pp. 817–833, 1989.

[16] H.W. Sorenson and D.L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, pp. 465–479, 1971.

[17] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. AC-17, no. 4, pp. 439–448, August 1972.

[18] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, November 2006.

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

[20] F. Gustafsson, "The marginalized likelihood ratio test for detection abrupt changes," *IEEE Transactions on Automatic Control*, vol. 41, no. 1, pp. 66–78, January 1996.

PUBLICATION 7

Henri Pesonen and Robert Piché: Variational Bayes algorithm for tracking linear systems with abrupt changes of the noise covariances. Research report 100, Tampere University of Technology, Department of Mathematics, 2012.

Variational Bayes algorithm for tracking linear systems with abrupt changes of the noise covariances

H. Pesonen, *Tampere University of Technology*
R. Piché, *Tampere University of Technology*

***Abstract*—The variational Bayes method is applied to the state-space estimation problem with manoeuvres or changes in the covariance of the observation noise. The resulting algorithm is an off-line batch method that can be used to provide a baseline performance estimation results for the recursive methods. In addition to batch methods we introduce a heuristic approach to make the algorithm on-line. Through simulations we show how the introduced method obtain the best accuracy out of all compared approximative estimation methods.**

***Index Terms*—Bayesian estimation, Kalman filter, Rauch-Tung-Striebel smoother, variational Bayes, change detection**

I. Introduction

The estimation of the state described by a linear state-space system requires that we define the parameters of the system, after which we can solve the system optimally with the Kalman filter (KF) if the system noise processes are normally distributed white processes [5]. However, it might be problematic to describe the processes with a Gaussian distribution. Gaussian mixture (GM) distributions are more general models that can take into account several plausible models for the system [1, 9]. For example, a navigation system with manoeuvres can be described with one model for the constant velocity motion and an another model for the manoeuvres [3]. Also, systems with outlying measurements can be described with one model for the good data and another for the bad [7]. Although the most straightforward approach would be to use the Gaussian model even if there are multiple models, it is possible that the performance may be degraded. In the present work we derive the variational Bayes (VB) algorithm for approximating the posterior distribution of the state within a time-window, given that the noise processes are described with a two-component GM distribution. The VB method can be applied as a batch method for computing the posterior distribution offline for the whole track, or as a online method using a moving window for the estimation.

This note is organized as follows. In Section II we describe the linear state-space model with GM-noise processes. In Section III we formulate the VB algorithm for the problem and in Section IV we test several methods in two sets of simulations. Finally in Section V we conclude the study.

II. Problem

We model the problem of the abruptly changing linear dynamic system as follows. The system is constructed as a linear state-space model

$$x_k = F_{k-1}x_{k-1} + w_{k-1}, \tag{1}$$
$$y_k = H_k x_k + v_k, \tag{2}$$
$$x_0 \sim \mathsf{N}\left(x_{0|0}, P_{0|0}\right), \tag{3}$$

where v_k and w_k are mutually independent white noise processes independent of the initial state x_k, and $\mathsf{N}(\mu, \Sigma)$ is the normal distribution with the mean μ and the covariance Σ. The noise processes have two plausible models and are defined as follows. The state noise is modeled as

$$w_k \sim \mathsf{N}(0, Q_k)^{1-\lambda_{k+1}} \mathsf{N}(0, M_k)^{\lambda_{k+1}}, \tag{4}$$

and the observation noise as

$$v_k \sim \mathsf{N}(0, R_k)^{1-\lambda_k} \mathsf{N}(0, W_k)^{\lambda_k}, \tag{5}$$

where $\lambda_k \in \{0, 1\}, k = 1, \ldots, N$ are mutually independent Bernoulli-distributed random variables

$$\lambda_k \sim \mathrm{Ber}(\theta). \tag{6}$$

In theory the problem can be solved using the Bayesian framework. The posterior distribution of the state is

$$p(x_{0:N} \mid y_{1:N}) = \sum_{\lambda_1=0}^{1} \cdots \sum_{\lambda_N=0}^{1} p(x_{0:N} \mid y_{1:N}, \lambda_{1:N}) p(\lambda_{1:N} \mid y_{1:N}), \tag{7}$$

where $a_{1:N} \triangleq [a_1, \ldots a_N]$ The posterior (7) is a GM distribution and evaluation of it or its marginals is an enormous task for even small N. Therefore, we are required to restrict ourself to solving either only parts of the problem, or to approximate the posterior distribution. For example, we could restrict ourselves to computing only the marginal distributions of (7). The marginal $p(x_k \mid y_{1:N}, \lambda_{1:N}) = \mathsf{N}\left(x_{k|N}(\lambda_{1:N}), P_{k|N}(\lambda_{1:N})\right)$ can be computed recursively using KF for $k = N$

$$[x_{k+1|k+1}(\lambda_{1:k+1}), P_{k+1|k+1}(\lambda_{1:k+1})] \leftarrow \mathrm{KalmanStep}(x_{k|k}(\lambda_{1:k}), P_{k|k}(\lambda_{1:k}), y_k, F_k, Q_k^{1-\lambda_{k+1}} M_k^{\lambda_{k+1}}, H_k, R_k^{1-\lambda_{k+1}} W_k^{\lambda_{k+1}}), \tag{8}$$

where $\mathrm{KalmanStep}$ is one step of the KF algorithm, as given in Algorithm 1.

Algorithm 1 $[x_{k|k}, P_{k|k}] \leftarrow \mathrm{KalmanStep}(x_{k-1|k-1}, P_{k-1|k-1}, y_k, F_{k-1}, \mathcal{Q}_{k-1}, H_k, \mathcal{R}_k)$

1: $x_{k|k-1} \leftarrow F_{k-1} x_{k-1|k-1}$
2: $P_{k|k-1} \leftarrow F_{k-1} P_{k-1|k-1} F_{k-1}^T + \mathcal{Q}_{k-1}$
3: $K_k \leftarrow P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + \mathcal{R}_k)^{-1}$
4: $x_{k|k} \leftarrow x_{k|k-1} + K_k(y_k - H_k x_{k|k-1})$
5: $P_{k|k} \leftarrow P_{k|k-1} + K_k H_k P_{k|k-1}$

Rauch-Tung-Striebel smoother (RTS) [8] is a recursive smoothing algorithm for the linear Gaussian state-space model to compute the marginals with $k < N$

$$\begin{aligned}&[x_{k|N}(\lambda_{1:N}), P_{k|N}(\lambda_{1:N}), C_{k+1|N}(\lambda_{1:N})] \\ &\leftarrow \mathrm{RTSStep}(x_{k+1|N}(\lambda_{1:N}), P_{k+1|N}(\lambda_{1:N}), x_{k|k}(\lambda_{1:N}), \\ &\qquad P_{k|k}(\lambda_{1:N}), F_k, Q_k^{1-\lambda_k} M_k^{\lambda_k}),\end{aligned} \tag{9}$$

where RTSStep is defined in Algorithm 2. In the algorithm, we compute also the cross-covariance

$$C_{k+1|N}(\lambda_{1:k}) = \mathrm{cov}(x_{k+1}, x_k \mid y_{1:N}, \lambda_{1:k}), \tag{10}$$

as it required in the VB approximation of the posterior (7) as discussed in Section III.

Algorithm 2 $[x_{k|N}, P_{k|N}, C_{k+1|N}] \leftarrow \mathrm{RTSStep}(x_{k+1|N}, P_{k+1|N}, x_{k|k}, P_{k|k}, F_k, \mathcal{Q}_k)$

1: $x_{k+1|k} \leftarrow F_k x_{k|k}$
2: $P_{k+1|k} \leftarrow F_k P_{k|k} F_k^T + \mathcal{Q}_k$
3: $G_k \leftarrow P_{k|k} F_k^T P_{k+1|k}^{-1}$
4: $C_{k+1|N} \leftarrow P_{k+1|N} G_k^T$
5: $x_{k|N} \leftarrow x_{k|k} + G_k(x_{k+1|N} - x_{k+1|k})$
6: $P_{k|N} \leftarrow P_{k|k} + G_k(P_{k+1|N} - P_{k+1|k}) G_k^T$

The weight in (7) is obtained by running the bank of 2^N KFs.

$$\begin{aligned}&p(\lambda_{1:N} \mid y_{1:N}) \propto p(y_{1:N} \mid \lambda_{1:N}) p(\lambda_{1:N}) \\ &= \prod_{k=1}^{N} p(y_k \mid y_{1:k-1}, \lambda_{1:k}) p(\lambda_k) \\ &= \prod_{k=1}^{N} \mathsf{N}\left(y_k \mid H_k x_{k|k-1}(\lambda_{1:k}), S_k(\lambda_{1:k})\right) (1-\theta_k)^{1-\lambda_k} \theta_k^{\lambda_k}.\end{aligned} \tag{11}$$

There exists several methods for making the evaluation of the distributions feasible. Most methods are based on cutting off or merging the branches of the mixture filtering distribution $p(x_k \mid y_{1:k})$ [1, 9, 7, 2].

III. Variational approximation

We use the VB approach to approximate the posterior (7). In the VB method we seek out the optimal approximative distributions $q_\cdot(\cdot)$ that can be factorized as

$$\begin{aligned}&p(x_{1:N}, \lambda_{1:N} | y_{1:N}) \\ &\approx q_{x_{1:N}, \lambda_{1:N}}(x_{1:N}, \lambda_{1:N}) = q_{x_{1:N}}(x_{1:N}) \prod_{k=1}^{N} q_{\lambda_k}(\lambda_k).\end{aligned} \tag{12}$$

This will be the only assumption about the form of $q_\cdot(\cdot)$. Distributions $q_{x_{1:N}}(x_{1:N}), q_{\lambda_1}(\lambda_1), \ldots, q_{\lambda_N}(\lambda_N)$ are found such that they minimize the Kullback-Leibler (KL) divergence between the posterior and the approximative distribution

$$\begin{aligned}&\mathrm{KL}(q(\lambda_{1:N}, x_{1:N}) || p(x_{1:N}, \lambda_{1:N} | y_{1:N})) \\ &= \int q(x_{1:N}) \prod_{k=1}^{N} q(\lambda_k) \log \frac{p(x_{1:N}, \lambda_{1:N} | y_{1:N})}{q(x_{1:N}) \prod_{k=1}^{N} q(\lambda_k)} \mathrm{d}x_{1:N} \mathrm{d}\lambda_{1:N}.\end{aligned} \tag{13}$$

In (13) and in the following we have left out the subscript from the approximative distributions $q(\cdot)$. The KL divergence can be minimized using calculus of variations by first fixing $q(x_{1:N})$ and $q(\lambda_i)$, $i \in 1:N \backslash k$, to minimize (13) with respect to $q(\lambda_k)$. As the result, we get

$$\begin{aligned}&\log q(\lambda_k) \\ &= \mathsf{E}_{x_{1:N}, \lambda_{1:N \backslash k}} (\log p(x_{1:N}, \lambda_{1:N} \mid y_{1:N})) + \text{const.}\end{aligned} \tag{14}$$

for $k = 1, \ldots, N$ The expectation $\mathsf{E}_{x_{1:N}, \lambda_{1:N \backslash k}}(\cdot)$ is evaluated for $q(x_{1:N}) \prod_{i=1, i \neq k}^{N} q(\lambda_i)$. After finding $q(\lambda_k), k = 1, \ldots, N$, we minimize (13) with respect to $q(x_{1:N})$ as

$$\begin{aligned}&\log q(x_{1:N}) \\ &= \mathsf{E}_{\lambda_{1:N}} (\log p(x_{1:N}, \lambda_{1:N} \mid y_{1:N})) + \text{const.},\end{aligned} \tag{15}$$

where the expectation $\mathsf{E}_{\lambda_{1:N}}(\cdot)$ is evaluated for $\prod_{k=1}^{N} q(\lambda_k)$.

To compute (14) and (15), we express the posterior distribution as

$$\begin{aligned}&p(x_{1:N}, \lambda_{1:N} \mid y_{1:N}) \\ &= p(y_{1:N} \mid x_{1:N}, \lambda_{1:N}) p(x_{1:N}, \lambda_{1:N}) \\ &= \prod_{k=1}^{N} p(y_k \mid x_k, \lambda_k) p(x_k \mid x_{k-1}, \lambda_k) p(\lambda_k)\end{aligned} \tag{16}$$

and find its logarithm

$$\begin{aligned}&\log p(x_{1:N}, \lambda_{1:N} \mid y_{1:N}) \\ &= \sum_{k=1}^{N} \log p(y_k \mid x_k, \lambda_k) + \log p(x_k \mid x_{k-1}, \lambda_k) + \log p(\lambda_k) \\ &= \sum_{k=1}^{N} (1-\lambda_k) \left[-\frac{1}{2} \log\det R_k - \frac{1}{2} ||y_k - H_k x_k||^2_{R_k^{-1}} \right. \\ &\left. -\frac{1}{2} \log\det Q_{k-1} - \frac{1}{2} ||x_k - F_{k-1} x_{k-1}||^2_{Q_{k-1}^{-1}} + \log(1-\theta) \right] \\ &+ \lambda_k \left[-\frac{1}{2} \log\det W_k - \frac{1}{2} ||y_k - H_k x_k||^2_{W_k^{-1}} \right. \\ &\left. -\frac{1}{2} \log\det M_{k-1} - \frac{1}{2} ||x_k - F_{k-1} x_{k-1}||^2_{M_{k-1}^{-1}} + \log\theta \right].\end{aligned} \tag{17}$$

We compute (14) as

$$\begin{aligned}
&\log q(\lambda_k) = \mathsf{E}_{x_{1:N},\lambda_{1:N\setminus k}}\left(\log p(x_{1:N},\lambda_{1:N} \mid y_{1:N})\right) + \text{const.}\\
&= (1-\lambda_k)\mathsf{E}_{x_{k-1:k}}\left(-\frac{1}{2}\log\det R_k - \frac{1}{2}||y_k - H_k x_k||^2_{R_k^{-1}}\right.\\
&\left.-\frac{1}{2}\log\det Q_{k-1} - \frac{1}{2}||x_k - F_{k-1}x_{k-1}||^2_{Q_{k-1}^{-1}} + \log(1-\theta)\right)\\
&+ \lambda_k\mathsf{E}_{x_{k-1:k}}\left(-\frac{1}{2}\log\det W_k - \frac{1}{2}||y_k - H_k x_k||^2_{W_k^{-1}}\right.\\
&\left.-\frac{1}{2}\log\det M_{k-1} - \frac{1}{2}||x_k - F_{k-1}x_{k-1}||^2_{M_{k-1}^{-1}}\log(\theta)\right)\\
&+ \text{const.} \qquad (18)
\end{aligned}$$

After some mechanical manipulation, we introduce the notation

$$\begin{aligned}
\log\rho_{k,1} &= -\frac{1}{2}\log\det R_k - \frac{1}{2}||y_k - H_k x_{k|N}||^2_{R_k^{-1}}\\
&-\frac{1}{2}\mathrm{tr}\left(H_k^T R_k^{-1} H_k P_{k|N}\right) - \frac{1}{2}\log\det Q_{k-1}\\
&-\frac{1}{2}||x_{k|N} - F_{k-1}x_{k-1|N}||^2_{Q_{k-1}^{-1}}\\
&-\frac{1}{2}\mathrm{tr}\left(\left(Q_{k-1}^{-1} + F_{k-1}Q_{k-1}^{-1}F_{k-1}^T\right)\right.\\
&\left.\times\left(P_{k|N} - 2C_{k|N}F_{k-1}^T + F_{k-1}P_{k-1|N}F_{k-1}^T\right)\right) + \log(1-\theta)\\
\log\rho_{k,2} &= -\frac{1}{2}\log\det W_k - \frac{1}{2}||y_k - H_k x_{k|N}||^2_{W_k^{-1}}\\
&-\frac{1}{2}\mathrm{tr}\left(H_k^T W_k^{-1} H_k P_{k|N}\right) - \frac{1}{2}\log\det M_{k-1}\\
&-\frac{1}{2}||x_{k|N} - F_{k-1}x_{k-1|N}||^2_{M_{k-1}^{-1}}\\
&-\frac{1}{2}\mathrm{tr}\left(\left(M_{k-1}^{-1} + F_{k-1}M_{k-1}^{-1}F_{k-1}^T\right)\right.\\
&\left.\times\left(P_{k|N} - 2C_{k|N}F_{k-1}^T + F_{k-1}P_{k-1|N}F_{k-1}^T\right)\right) + \log\theta,
\end{aligned}$$

where $\mathsf{E}_{x_k}(x_k) = x_{k|N}$, $\mathsf{V}(x_k) = P_{k|N}$ and $\mathrm{cov}(x_k, x_{k-1}) = C_{k|N}$. The marginal density of the model indicator variable λ_k can be shown to be

$$q(\lambda_k) = (1-\theta_{k|N})^{1-\lambda_k}\theta_{k|N}^{\lambda_k} = \mathrm{Ber}(\theta_{k|N}) \qquad (19)$$

$$\theta_{k|N} = \frac{\rho_{k,2}}{\rho_{k,1} + \rho_{k,2}}. \qquad (20)$$

As $q(\lambda_k)$ is a Bernoulli-distribution, it has the mean $\mathsf{E}(\lambda_k) = \theta_{k|N}$.

After finding each of the marginal distributions $q(\lambda_i)$, we evaluate the marginal distribution of the state $q(x_{1:k})$ as

$$\begin{aligned}
&\log q(x_{1:N}) = \mathsf{E}_{\lambda_{1:N}}\left(\log p(x_{1:N},\lambda_{1:N} \mid y_{1:N})\right) + \text{const.}\\
&= \sum_{k=1}^{N} -\frac{1}{2}\mathsf{E}_{\lambda_k}(1-\lambda_k)\,||y_k - H_k x_k||^2_{R_k^{-1}}\\
&-\frac{1}{2}\mathsf{E}_{\lambda_k}(\lambda_k)\,||y_k - H_k x_k||^2_{W_k^{-1}}\\
&-\frac{1}{2}\mathsf{E}_{\lambda_k}(1-\lambda_k)\,||x_k - F_{k-1}x_{k-1}||^2_{Q_{k-1}^{-1}}\\
&-\frac{1}{2}\mathsf{E}_{\lambda_k}(\lambda_k)\,||x_k - F_{k-1}x_{k-1}||^2_{M_{k-1}^{-1}} + \text{const.}\\
&= \sum_{k=1}^{N} -\frac{1}{2}||y_k - H_k x_k||^2_{\Xi_k^{-1}} - \frac{1}{2}||x_k - F_{k-1}x_{k-1}||^2_{\Sigma_{k-1}^{-1}} + \text{const.},
\end{aligned}$$

where

$$\begin{aligned}
\Xi_k^{-1} &= \mathsf{E}_{\lambda_k}(1-\lambda_k)\,R_k^{-1} + \mathsf{E}_{\lambda_k}(\lambda_k)\,W_k^{-1}\\
&= (1-\theta_{k|N})R_k^{-1} + \theta_{k|N}W_k^{-1} \qquad (21)\\
\Sigma_{k-1}^{-1} &= \mathsf{E}_{\lambda_k}(1-\lambda_k)\,Q_{k-1}^{-1} + \mathsf{E}_{\lambda_k}(\lambda_k)\,M_{k-1}^{-1}\\
&= (1-\theta_{k|N})Q_{k-1}^{-1} + \theta_{k|N}M_{k-1}^{-1}. \qquad (22)
\end{aligned}$$

We can see that the density $q(x_{1:N})$ is a normal distribution and we can compute the marginals

$$q(x_{k-1:k}) = \mathsf{N}\left(\begin{bmatrix} x_{k|N} \\ x_{k-1|N}\end{bmatrix}, \begin{bmatrix} P_{k|N} & C_{k|N} \\ C_{k|N}^T & P_{k-1|N}\end{bmatrix}\right) \qquad (23)$$

using KF and RTS-smoother. The set of equations (19) and (23) can be solved by a fixed-point iteration for which the convergence is guaranteed given certain convexity properties of the error in the approximative distribution [4]. This is the VB method that is summarized in Algorithm 3. Although convergence checks could be performed within the algorithm, we fix the number of iterations to M to control the computational costs. The resulting algorithm is very close to the EM-method for detecting change in the state transition model [3, 6].

Algorithm 3 $[x_{0:N|N}, P_{0:N|N}, \theta_{1:N|N}] \leftarrow$
$\mathrm{VB}(x_{0|0}, P_{0|0}, \theta_{1:N}, F_{1:N-1}, Q_{0:N-1}, M_{0:N-1}, H_{1:N}, R_{1:N}, W_{1:N})$

$\theta_{k|N} \leftarrow 0, \quad k = 1,\ldots,N$
$a^{(1)} \leftarrow -\frac{1}{2}\log\det R_k - \frac{1}{2}\log\det Q_{k-1} + \log(1-\theta_k)$
$a^{(2)} \leftarrow -\frac{1}{2}\log\det W_k - \frac{1}{2}\log\det M_{k-1} + \log\theta_k$
for $m = 1,\ldots,M$ **do**
  for $k = 0,\ldots,N-1$ **do**
    $\Xi_{k+1}^{-1} \leftarrow (1-\theta_{k+1|N})R_{k+1}^{-1} + \theta_{k+1|N}W_{k+1}^{-1}$
    $\Sigma_k^{-1} \leftarrow (1-\theta_{k+1|N})Q_k^{-1} + \theta_{k+1|N}M_k^{-1}$
    $[x_{k+1|k+1}, P_{k+1|k+1}] \leftarrow$
    $\mathrm{KalmanStep}(x_{k|k}, P_{k|k}, y_k, F_k, \Sigma_k, H_{k+1}, \Xi_{k+1})$
  end for
  for $j = N-1,\ldots,0$ **do**
    $[x_{j|N}, P_{j|N}, C_{j+1|N}] \leftarrow$
    $\mathrm{RTSStep}(x_{j+1|N}, P_{j+1|N}, x_{j|j}, P_{j|j}, F_j, \Sigma_j)$
  end for
  for $i = 1,\ldots N$ **do**
    $\log\rho_{k,1} \leftarrow a^{(1)} - \frac{1}{2}||y_k - H_k x_{k|N}||^2_{R_k^{-1}}$
    $-\frac{1}{2}\mathrm{tr}\left(H_k^T R_k^{-1} H_k P_{k|N}\right) - \frac{1}{2}||x_{k|N} - F_{k-1}x_{k-1|N}||^2_{Q_{k-1}^{-1}}$
    $-\frac{1}{2}\mathrm{tr}\left(\left(Q_{k-1}^{-1} + F_{k-1}Q_{k-1}^{-1}F_{k-1}^T\right)\right.$
    $\left.\times\left(P_{k|N} - 2C_{k|N}F_{k-1}^T + F_{k-1}P_{k-1|N}F_{k-1}^T\right)\right)$
    $\log\rho_{k,2} \leftarrow a^{(2)} - \frac{1}{2}||y_k - H_k x_{k|N}||^2_{W_k^{-1}}$
    $-\frac{1}{2}\mathrm{tr}\left(H_k^T W_k^{-1} H_k P_{k|N}\right) - \frac{1}{2}||x_{k|N} - F_{k-1}x_{k-1|N}||^2_{M_{k-1}^{-1}}$
    $-\frac{1}{2}\mathrm{tr}\left(\left(M_{k-1}^{-1} + F_{k-1}M_{k-1}^{-1}F_{k-1}^T\right)\right.$
    $\left.\times\left(P_{k|N} - 2C_{k|N}F_{k-1}^T + F_{k-1}P_{k-1|N}F_{k-1}^T\right)\right)$
    $\theta_{k|N} \leftarrow \frac{\rho_{k,2}}{\rho_{k,1}+\rho_{k,2}}$
  end for
end for

The Algorithm 3 is an offline method for approximating the posterior distribution but can be heuristically modified for online applications. First we choose a window size K, and then approximate $p(x_{1:K}, \lambda_{1:K} \mid y_{1:K})$ using the VB method. Then using $q(x_K)$ as the prior, we approximate

$p(x_{K+1:2K}, \lambda_{K+1:2K} \mid y_{1:2K})$ by applying the VB method for the data $y_{k+1:K}$. This is repeated at every Kth time step.

IV. Simulations

We simulate two cases of a GPS-positioning problem. In both problems the estimation methods are based solving the system (1)–(3), where

$$F_k = \begin{bmatrix} I_2 & I_2 \\ 0_2 & I_2 \end{bmatrix}, \quad H_k = \begin{bmatrix} I_2 & 0_2 \end{bmatrix} \tag{24}$$

$$Q_k = \sigma^2 \begin{bmatrix} \frac{1}{3}I_2 & \frac{1}{2}I_2 \\ \frac{1}{2}I_2 & I_2 \end{bmatrix}, \quad R_k = \begin{bmatrix} 10^2 & \frac{5}{2}^2 \\ \frac{5}{2}^2 & 5^2 \end{bmatrix}, \tag{25}$$

and if not otherwise mentioned, these parameters are used to generate the simulation data.

A. Manoeuvring target

In the example we consider only changes in the state transition model, or manoeuvres. We generated 100 tracks with velocities

$$\Delta x_j = \begin{cases} \begin{bmatrix} 5+\nu_j & 0 \end{bmatrix}^T, & j \in [0,19] \cup [51,70] \\ \begin{bmatrix} 0 & 5+\nu_j \end{bmatrix}^T, & j \in [21,49] \\ \begin{bmatrix} 5/\sqrt{2} & 5/\sqrt{2} \end{bmatrix}^T, & j \in \{20,50\}, \end{cases} \tag{26}$$

where $\nu_j \sim \mathsf{N}\left(0, 0.1^2\right)$ is a white noise process. All the estimation methods model the constant velocity motion with $\sigma = 0.1$ and the manouevres with $M_k = 100 \cdot Q_k$. The compared methods are the KFs and RTSs using only Q_k (KF1,RTS1) or M_k (KF2,RTS2), EM-algorithm (EM) [3], GM filter (GM) with component merging at the each time step [7], the VB algorithm (VB) and the moving window VB (with window size 15) (MWVB), both with 40 iterations. VB and EM methods use the prior $\theta_k = 0.1$. From the simulations we investigate the root mean square error (RMSE) performance and the error threshold containing 95% of the estimation errors. The numbers are reported in Table I. Amongst the online estimation methods, MWVB has the best performance and from the offline methods VB has the best performance. EM-algorithm seems to be more sensitive to the initial estimates of $\theta_k = 0$ than VB method, which is the reason for its performance being worse.

B. Change in the observation noise

In second problem, we generated 100 tracks of 70 time steps. The track is generated with the constant velocity model with $\sigma = 1$, and for the observations we simulated batches of observation noise with larger covariance. The observation noise is generated using $W_k = 25 \cdot R_k$ for time steps $k \in [20,30) \cup [50,60)$. The compared methods are the KFs and RTSs using only R_k (KF1, RTS1) or W_k (KF2, RTS2), EM-algorithm (EM) [3] modified for the problem, GM filter (GM) with component merging at each time step [7],the VB algorithm (VB) and the moving window VB (with window size 15) (WBVB), both with 40 iterations. VB and EM methods use the prior $\theta_k = 0.1$. RMSEs and 95% error threshold performances are reported in Table I. Again, MWVB has the best performance out of the online methods and from the offline methods VB has the best performance, although the performance is almost identical to the EM-algorithm.

	Test A		Test B	
	RMSE	95%-err	RMSE	95%-err
KF1	13.7	24.9	12.7	41.0
KF2	6.0	12.7	13.5	32.4
GM	7.2	15.1	11.6	37.2
MWVB	4.9	11.6	7.2	19.5
RTS1	9.6	21.5	7.5	21.3
RTS2	3.4	7.1	7.8	17.0
EM	3.6	7.6	5.7	14.1
VB	2.7	6.1	5.6	13.9

TABLE I
THE SIMULATIONS INDICATE THAT MWVB AND VB METHODS HAVE GOOD PERFORMANCE AMONG THE ONLINE AND OFFLINE METHODS.

V. Conclusions

A variational Bayes change detection method was described for linear state-space systems with noise processes defined by changing noise covariances. Through simulations it was shown that not only the method performs very well in offline mode, but a heuristic online modification of the technique provides good accuracy compared to other methods. Future study on the extension of the method for more general noise processes is required.

References

[1] D. L. Alspach and H. W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, AC-17(4):439–448, August 1972.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[3] N. Bergman and F. Gustafsson. Three statistical batch algorithms for tracking manoeuvring targets. Technical report, Department of Electrical Engineering, Linköping University, 1999.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 2006.

[5] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82, 1960.

[6] V. Krishnamurthy and J. B. Moore. On-line estimation of hidden Markov model parameters based on the Kullback-Leibler information measure. *IEEE Transactions on Signal Processing*, 41(8):2557–2573, August 1993.

[7] D. Peña and I. Guttman. Optimal collapsing of mixture distributions in robust recursive estimation. *Communications in Statistics: Theory and Methods*, 18(3):817–833, 1989.

[8] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, August 1965.

[9] H.W. Sorenson and D.L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7:465–479, 1971.

This thesis was typeset in LaTeX2e using the `memoir` document style. The body text is set with 12pt Adobe Utopia. All the figures were prepared with MATLAB and/or Xfig.

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-3008-1
ISSN 1459-2045