TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 813 • Publication 813

Simo Ali-Löytty

Gaussian Mixture Filters in Hybrid Positioning

Tampere 2009

Tampereen teknillinen yliopisto. Julkaisu 813
Tampere University of Technology. Publication 813

Simo Ali-Löytty

Gaussian Mixture Filters in Hybrid Positioning

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 7th of August 2009, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2009

ISBN 978-952-15-2166-9 (printed)
ISBN 978-952-15-2171-3 (PDF)
ISSN 1459-2045

Abstract

Bayesian filtering is a framework for the computation of an optimal state estimate fusing different types of measurements, both current and past. Positioning, especially in urban and indoor environments, is one example of an application where the powerful mathematical framework is needed to compute as a good position estimate as possible from all kinds of measurements and information.

In this work, we consider the Gaussian mixture filter, which is an approximation of the Bayesian filter. Especially, we consider filtering with just a few components, which can be computed in real-time on a mobile device. We have developed and compared different Gaussian mixture filters in different scenarios. One filter uses static solutions, which are possibly ambiguous, another extends the Unscented transformation to the Gaussian mixture framework, and some filters are based on partitioning the state space. It is also possible to use restrictive, inequality constraints, efficiently in Gaussian mixture filters.

We show that a new filter called the Efficient Gaussian mixture filter outperforms other known filters, such as Kalman-type filters or particle filter, in a positioning application. We also show that another new filter, the Box Gaussian mixture filter, converges weakly to the correct posterior. All in all we see that the Gaussian mixture is a competitive framework for real-time filtering implementations, especially in positioning applications.

Preface

The work presented in this thesis was carried out in the Department of Mathematics at Tampere University of Technology during the years 2005–2009.

I want to express my sincere gratitude to my supervisor Prof Robert Piché for his skilful guidance and encouragement during my doctoral studies. I am grateful to my colleagues in the Personal Positioning Algorithms Research group and in the department, especially Dr Niilo Sirola, Dr Kari Heine, Tommi Perälä and Henri Pesonen, for enlightening discussions and constructive feedback. I would like to thank Dr Jari Syrjärinne from Nokia Corporation for different perspectives and encouraging comments, the pre-examiners Dr Simon Julier and Dr Oleg Stepanov for their comments and suggestions. The financial support of Tampere Graduate School in Information Science and Engineering, Nokia Corporation and Nokia Foundation are gratefully acknowledged.

I am very grateful to my family, especially my dear wife Anna for hers love and support all these years and my wonderful son Onni.

I would like to dedicate this thesis in memory of my sons Joonatan and Oiva.

To God be the glory for all the things He has done in my life.

Tampere, May 2009,

Simo Ali-Löytty

Contents

List of publications

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

P1. "Consistency of three Kalman filter extensions in hybrid navigation" (with Niilo Sirola and Robert Piché). *European Journal of Navigation*, 4(1), 2006.

P2. "A modified Kalman filter for hybrid positioning" (with Niilo Sirola) In *Proceedings of the 19th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2006)*, Fort Worth TX, September 2006.

P3. "Gaussian mixture filter in hybrid navigation" (with Niilo Sirola) In *Proceedings of the European Navigation Conference 2007 (ENC 2007)*, Geneva, May 2007.

P4. "Gaussian mixture filter and hybrid positioning" (with Niilo Sirola) In *Proceedings of the 20th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2007)*, Fort Worth TX, September 2007.

P5. "Efficient Gaussian mixture filter for hybrid positioning" In *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS 2008)*, Monterey CA, May 2008.

P6. "On the convergence of the Gaussian mixture filter" Research report 89, *Tampere University of Technology, Institute of Mathematics*, 2008.

The author's role in the shared publications:

P1: The author developed the general inconsistency test, wrote the filters' code, ran the tests and wrote the manuscript.

P2: The author developed a method for using restrictive information with Kalman-type filters, wrote the code as well as the manuscript.

P3: The author developed the algorithms and part of the code, and wrote the manuscript.

P4: The author developed the algorithms and part of the code, and wrote the manuscript.

The main contributions of the publications are:

- **P1:** The general inconsistency test without any assumptions of the shape of the distribution. The publication also shows that nonlinearity is not significant in satellite measurement, but it might be significant in base station range measurement.
- **P2:** A method for taking into account restrictive information (inequality constraints) in Kalman-type filters. This method is also applied to hybrid positioning using base station sector and maximum range information (cell-ID) with Kalman-type filters.
- **P3:** A method for applying static solutions (in our case likelihood peaks) in the filtering framework, that is approximate likelihood as Gaussian mixture. Publication also presents how we can use static solutions to robustify Kalman-type filters.
- **P4:** Sigma Point Gaussian Mixture Approximation (SPGMA), which extends sigma-points to the Gaussian mixture filter framework. The publication also presents how to measure nonlinearity, when EKF possibly fails.
- **P5:** Box and Efficient Gaussian Mixture Filters (BGMF and EGMF), that split the state space into sets using parallel planes. BGMF and EGMF give better performance in hybrid positioning simulations than the particle filter.
- **P6:** The convergence results of BGMF. BGMF increases number of components only to the "dimensions" where it is necessary and BGMF allows the reduction of the number of components.

Related publications

The following publications relate to the subject, but are either redundant or contain only minor contribution from the author:

R1. Simo Ali-Löytty, Niilo Sirola, and Robert Piché. Consistency of three Kalman filter extensions in hybrid navigation. In *Proceedings of the European Navigation Conference GNSS 2005 (ENC 2005)*, Munich, July 2005.

R2. Niilo Sirola and Simo Ali-Löytty. Local positioning with parallelepiped moving grid. In *Proceedings of the 3rd Workshop on Positioning, Navigation and Communications (WPNC'06)*, March 16, 2006.

R3. Niilo Sirola and Simo Ali-Löytty. Moving grid filter in hybrid local positioning. In *Proceedings of the European Navigation Conference (ENC 2006)*, Manchester, May 7-10, 2006.

R4. Niilo Sirola, Simo Ali-Löytty, and Robert Piché. Benchmarking nonlinear filters. In *Nonlinear Statistical Signal Processing Workshop (NSSPW 2006)*, Cambridge, September 2006.

R5. Matti Raitoharju, Niilo Sirola, Simo Ali-Löytty, and Robert Piché. PNaFF: a modular software platform for testing hybrid positioning estimation algorithms. In *Proceedings of the 5th Workshop on Positioning, Navigation and Communication (WPNC'08)*, Hannover, March 28, 2008.

R6. Tommi Perälä and Simo Ali-Löytty. Kalman-type positioning filters with floor plan information. In *Third International Workshop on Broadband and Wireless Computing, Communication and Applications (BWCCA 2008)*, Linz, November 24-26, 2008.

R7. Ville Honkavirta, Tommi Perälä, Simo Ali-Löytty, and Robert Piché. A comparative survey of WLAN location fingerprinting methods. In *Proceedings of the 6th Workshop on Positioning, Navigation and Communication (WPNC'09)*, Hannover, March 19, 2009.

R8. Simo Ali-Löytty. On the convergence of box mixture filters. In *IEEE Transactions on Automatic Control* (submitted).

Nomenclature

Abbreviations

B	Byte
BGMA	Box Gaussian Mixture Approximation
BGMF	Box Gaussian Mixture Filter
BLUE	Best Linear Unbiased Estimator
cdf	cumulative density function
CPU	Central Processing Unit
EGMF	Efficient Gaussian Mixture Filter
EKF	Extended Kalman Filter
EKF2	Second Order Extended Kalman Filter
GM	Gaussian Mixture
GMF	Gaussian Mixture Filter
GML	Gaussian Mixture Likelihood
GPS	Global Positioning System
ID	Identity
IEKF	Iterated Extended Kalman Filter
KF	Kalman Filter
LKF	Linearized Kalman Filter
PF	Particle Filter
PNaFF	Personal Navigation Filter Framework
RMS	Root Mean Squares
SPGMA	Sigma Point Gaussian Mixture Approximation
SPGMF	Sigma Point Gaussian Mixture Filter
UKF	Unscented Kalman Filter
UT	Unscented Transformation
WLAN	Wireless Local Area Network
2D	two dimensional
3D	three dimensional
6D	six dimensional

Symbols

∞	infinity
$\cdot_k$	lower index k represents time t_k
α_i	weight of ith GM component
$\dot{A}$	novel polyhedron approximation of restrictive area
$\sqrt{\mathrm{A}}$	Square root of symmetric positive semidefinite matrix A, such as $\mathrm{A} = \sqrt{\mathrm{A}}\sqrt{\mathrm{A}}^T$
$\mathrm{A} < \mathrm{B}$	$x^T(\mathrm{B}-\mathrm{A})x > 0$ for all $x \neq 0$
c_g	parameter of conventional GM approximation
d_{ij}	distance between mixture components [P3, Page 834]
e_i	vector whose ith component is one and others are zero
$\mathrm{E}(x)$	expectation value of random variable x
$\mathrm{Err}_\mathrm{app}$	error, $\sup_\xi \lvert F_x(\xi) - F_{x_\mathrm{app}}(\xi)\rvert$
F	state transition matrix
F_x	cumulative density function (cdf) of random variable x
h	measurement function
H	linear measurement function or Jacobian of h
$\ddot{\mathrm{H}}_{k_i}$	Hessian of ith component of function h_k
I	identity matrix
K	Kalman gain
l	parameter of BGMA [P5]
n_eff	approximation of effective sample size
n_gm	number of GM components
n_pf	number of particles
n_thr	threshold value
n_x	dimension of variable x
$\mathbb{N}$	natural numbers, $\{0,1,2,3,\ldots\}$
N	parameter of BGMA [P6]
$\mathrm{N}_n(\hat{x},\mathrm{P})$	Gaussian distribution with parameters $\hat{x} \in \mathbb{R}^n$ and symmetric positive semidefinite matrix $\mathrm{P} \in \mathbb{R}^{n\times n}$

$\mathrm{N}_{\mathrm{p}}^{\hat{x}}(x)$	density function of non-singular $\mathrm{N}_n(\hat{x}, \mathrm{P})$
$p_{\mathrm{c}}(x)$	continuous density function
$p_{\mathrm{gm}}(x)$	density function of GM
$p_x(x), p(x)$	density function of random variable x
$p(x_k \mid x_{k-1})$	transitional density
$p(x_k \mid y_{1:k-1})$	prior (density function)
$p(x_k \mid y_{1:k})$	posterior (density function)
$p(y_k \mid x_k)$	likelihood
P	covariance of posterior
P^-	covariance of prior
P_{xy}	$\mathrm{E}\left((x-\mathrm{E}(x))\left(y-\mathrm{E}(y)\right)^T\right)$
$\mathrm{P}(A)$	probability of event A
Φ	cdf of $\mathrm{N}_1(0,1)$
Q	covariance of state model noise
$\mathbb{R}$	real numbers
R	covariance of measurement noise
τ	parameter of SPGMA
$\mathrm{tr}(\mathrm{A})$	trace of matrix $\mathrm{A} \in \mathbb{R}^{n \times n}$, $\mathrm{tr}(\mathrm{A}) = \sum_{i=1}^{n} A_{i,i}$
$\mathrm{U}(a,b]$	uniform distribution with support $(a,b]$
$u_{\mathrm{prior}\to\mathrm{bs}}$	unit vector from prior mean to base station
v	measurement noise
$\mathrm{V}(x)$	covariance of random variable x
ω_i	weight of ith sigma point
w	state model noise
$w^{(i)}$	weight of ith particle of PF
x^-	prior state
$\hat{x}$	estimate of x, usual mean of x
$\|x\|_{\mathrm{A}}^2$	weighted squared norm $\|x\|_{\mathrm{A}}^2 = x^T \mathrm{A} x$
χ	sigma-point
x	usually state of the system
$x^{(i)}$	ith particle
$x_{\mathrm{g}}^{(i)}$	ith grid point on conventional GM approximation
$x \mid y$	conditional probability distribution of x when condition y is fixed
y	usually measurement
$y_{1:k}$	set of measurements $y_1, \ldots, y_k$

CHAPTER 1

Introduction

This thesis consists of an introduction and six published articles. The purpose of this introductory chapter is to give a short unified background of the problem and summarise the results of publications [P1–P6].

1 Background

Personal positioning has become very popular in recent years. In good signal environments, such as open-sky environments, a satellite based positioning system, e.g. Global Positioning System (GPS, which is also a popular abbreviation for any positioning device), provides accurate and reliable positioning. However, in poor signal environments, such as urban canyons or indoors, satellite based positioning systems do not work satisfactorily or at all.

One possible solution for this problem is to also use other possible measurements and information in positioning. There are plenty of possible measurements sources, e.g. local wireless networks such as cellular networks [26, 53, 79, P1] or WLAN [3, 46, 65, R7] and map or floor plan information [18, 20, 40, R6], [51, 76, 83]. We call

positioning that uses measurements from many different sources *hybrid positioning*. Many of these possible measurements are so called "signals of opportunity", which are not originally intended for positioning, furthermore many of these measurements are highly nonlinear. For these reasons, conventional positioning algorithms such as the extended Kalman filter [P1] or the minimum least squares estimator [R4] do not work satisfactorily in hybrid positioning. More sophisticated measurement fusion methods for hybrid positioning are needed.

One possible fusion method is the Bayesian method, which is a probabilistic method [43, 64]. The Bayesian method enable different kinds of measurement and information to be fused. Futhermore, under some conditions it is possible to compute an optimal estimator recursively, which is an important property for real-time implementation. Unfortunately, it is not possible to compute this Bayesian solution analytically. Therefore, plenty of different approximate solutions, called filters, exist for this problem. For the rest of this thesis, we study these filters.

2 Problem statement

The problem of estimating the state of a stochastic system from noisy measurement data is considered. We consider the discrete-time system with a nonlinear measurement model:

$$\text{Initial state:} \quad x_0 \tag{1a}$$

$$\text{State model:} \quad x_k = \mathrm{F}_{k-1} x_{k-1} + w_{k-1} \tag{1b}$$

$$\text{Measurement model:} \quad y_k = h_k(x_k) + v_k \tag{1c}$$

where $k \in \mathbb{N}\backslash\{0\}$, vectors $x_{k-1} \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_{y_k}}$ represent the state of the system and the measurement, respectively. Matrix $\mathrm{F}_{k-1} \in \mathbb{R}^{n_x \times n_x}$ is the known state transition matrix, and $w_{k-1} \in \mathbb{R}^{n_x}$ is the state model noise with known statistics. Function $h_k : \mathbb{R}^{n_x} \to \mathbb{R}^{n_{y_k}}$ is a known measurement function, and $v_k \in \mathbb{R}^{n_{y_k}}$ is the measurement noise with known statistics. Subscript k represents time t_k. The aim of Bayesian filtering is to find the conditional

probability distribution given all current and past measurements, called the posterior and denoted $x_k|y_{1:k}$, where $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$. We consider the Bayesian filtering problem because the knowledge of the posterior enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error estimate is the conditional mean of x_k [8, 63]. We assume that noises w_k and v_k are white, mutually independent and independent of the initial state x_0; these assumptions allow posteriors to be computed recursively. We use the abbreviation $\mathrm{V}(w_k) = \mathrm{Q}_k$ and $\mathrm{V}(v_k) = \mathrm{R}_k$ for the covariance of the state model noise and the covariance of the measurement noise, respectively. We assume that initial state x_0 has density function $p_{x_0} \triangleq p_{x_0|y_{1:0}}$. If noises w_{k-1} and v_k have density functions $p_{w_{k-1}}$ and p_{v_k}, respectively, then the posterior can be determined according to the following relations [17, 63].

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

$$\text{Update:} \quad p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})\mathrm{d}x_k} \tag{2b}$$

where the transitional density (also referred to as transition equation) $p(x_k|x_{k-1}) = p_{w_{k-1}}(x_k - \mathrm{F}_{k-1}x_{k-1})$ and the likelihood $p(y_k|x_k) = p_{v_k}(y_k - h_k(x_k))$. We call the distribution $x_k|y_{1:k-1}$ the prior.

Usually assumptions that noises w_{k-1} and v_k have density functions are too strict. In some cases, the state model noise or the measurement noise does not have density function; this happens when we know that some components of the state are constant in time or our measurements contain restrictive information [P2], respectively. Throughout the remainder of this work, it is assumed that:

- Initial state x_0 has density function.
- Either state noise w_{k-1} has density function $p_{w_{k-1}}$, or state transition matrix F_{k-1} is nonsingular.
- Measurement noise v_k has stricly positive density p_{v_k} or $v_k = 0$ and $\int_{y_k = h_k(x_k)} p(x_k|y_{1:k-1})\mathrm{d}x_k > 0$.

These assumptions guarantee that the posterior has a density function $p(x_k|y_{1:k})$. Because the state transition function is linear, then the prior (prediction step), or at least the mean and covariance of the prior, is usually easy to compute. For example, results for Gaussian mixture case are given in [P6, Theorem 6]. So the biggest challenge in solving this Bayesian filtering problem is how to update the prior to obtain the posterior (update step). An illustration of our problem is in Figure 1, which presents the posterior when the prior[1] is Gaussian and we have two measurements, one linear[2] and one range[3] measurement, with (zero mean independent) Gaussian noise.

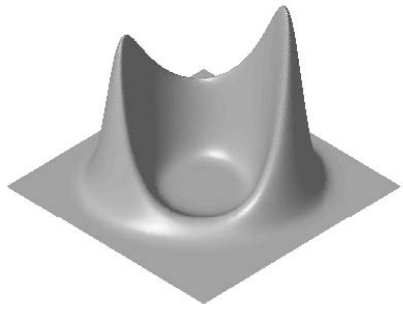

Figure 1: Posterior density with Gaussian prior, one linear measurement, one range measurement, and Gaussian noise.

We see that a crucial question in developing filters is how to store the distribution for the next time step. As we will see in Section 3 this is also one criterion for classifying filters. Note that the illustration in Figure 1, as well as other illustrations in Figures 2-9, are in 2D for practical reasons.[4] The real positioning applications are usually in 6D or higher dimensional state space. However, these illustrations, Figures 1-9, give a quite good idea of our problem and solutions.

[1] $x^- \sim \mathrm{N}_2\left(\begin{bmatrix} 50 \\ 40 \end{bmatrix}, \begin{bmatrix} 90^2 & 0 \\ 0 & 90^2 \end{bmatrix}\right)$.

[2] $0 = \begin{bmatrix} 1 & 1 \end{bmatrix} x + v_{\text{linear}}, \quad v_{\text{linear}} \sim \mathrm{N}_1\left(0, 90^2\right)$.

[3] $\sqrt{2} \cdot 90 = \|x\|_{\mathrm{I}} + v_{\text{range}}, \quad v_{\text{range}} \sim \mathrm{N}_1\left(0, 20^2\right)$.

[4] These illustrations are computed using MATLAB-software and 200^2 grid points.

The goal of this work is to develop an efficient filter with respect to computational and memory requirements. However, more important property than efficiency is that the filter is consistent [P1, R1], [50] and informative. Roughly speaking, a filter is consistent if the true estimation error is smaller than the estimated error, and informative if the difference of these errors is small.

3 Literature review

In this section, we give a brief literature review of filtering. We consider the problem that is presented in Section 2 and the filters that we have used in publications [P1–P6]. For further information see e.g. [5, 8, 13, 22, 29, 56, 57, 63, 71, 80]. This section contains different interpretations of the celebrated and well known Kalman Filter (KF) [39] (Section 3.1), Kalman-type filters (Section 3.2) and nonlinear filters (Section 3.3) especially Gaussian mixture filter, which we study also in Section 4.

3.1 Kalman filter

If the measurement function h_k (1c) is linear and all the noises and the initial state are Gaussian, then posterior $x_k|y_{1:k}$ is Gaussian, and its mean and covariance can be computed exactly using KF [2, 27]. KF considers the system:

$$\begin{aligned}
&\text{Initial state:} && x_0, && \mathrm{V}(x_0) = \mathrm{P}_0 && \text{(3a)}\\
&\text{State model:} && x_k = \mathrm{F}_{k-1} x_{k-1} + w_{k-1}, && \mathrm{V}(w_{k-1}) = \mathrm{Q}_{k-1} && \text{(3b)}\\
&\text{Meas. model:} && y_k = \mathrm{H}_k x_k + v_k, && \mathrm{V}(v_k) = \mathrm{R}_k && \text{(3c)}
\end{aligned}$$

where initial estimate (mean of initial state) is $\hat{x}_0$, covariance P_0 is non-singular, noises w_{k-1} and v_k have zero mean and covariances R_k are non-singular. KF algorithm is given as Algorithm 1. However, this Bayesian interpretation is not the only interpretation of KF. We present other interpretations of KF: the Best Linear Unbiased Estimator (BLUE) interpretation and a deterministic interpretation.

Algorithm 1: Kalman filter

1. Start with initial estimate $\hat{x}_0$ and estimate error covariance P_0. Set $k = 1$.
2. Prior estimate of state at t_k is $\hat{x}_k^- = \mathrm{F}_{k-1}\hat{x}_{k-1}$, and estimate error covariance is $\mathrm{P}_k^- = \mathrm{F}_{k-1}\mathrm{P}_{k-1}\mathrm{F}_{k-1}^T + \mathrm{Q}_{k-1}$.
3. Posterior estimate of state at t_k is $\hat{x}_k = \hat{x}_k^- + \mathrm{K}_k\left(y_k - \mathrm{H}_k\hat{x}_k^-\right)$, and estimate error covariance is $\mathrm{P}_k = (\mathrm{I} - \mathrm{K}_k\mathrm{H}_k)\mathrm{P}_k^-$, where Kalman gain $\mathrm{K}_k = \mathrm{P}_k^-\mathrm{H}_k^T\left(\mathrm{H}_k\mathrm{P}_k^-\mathrm{H}_k^T + \mathrm{R}_k\right)^{-1}$.
4. Increment k and repeat from 2.

BLUE interpretation of Kalman filter

BLUE interpretation drops the assumptions that initial state and noises are Gaussian and seeks unbiased and linear estimator[5] which minimizes the cost function $\mathrm{E}(\|x_k - \hat{x}_k\|^2)$. It can be shown that the posterior estimate of KF at t_k (Algorithm 1) minimizes that cost function, see for example [2, 5, 10, 39].

Deterministic interpretation of Kalman filter

Consider the deterministic least-squares problem

$$\hat{x}_k = \mathrm{argmin}_{x_k}\left(\|\hat{x}_0 - x_0\|^2_{\mathrm{P}_0^{-1}} + \sum_{n=1}^{k}\left(\|w_{n-1}\|^2_{\mathrm{Q}_{n-1}^{-1}} + \|v_n\|^2_{\mathrm{R}_n^{-1}}\right)\right),$$

where $w_{n-1} = x_n - \mathrm{F}_{n-1}x_{n-1}$ and $v_n = y_n - \mathrm{H}_n x_n$. Here we assume that all necessary matrices are non-singular. Now $\hat{x}_k$ is the same as the posterior estimate of KF at t_k (Algorithm 1) see example [38]. Note that this interpretation is truly deterministic: there are no assumptions of noise statistics or independence.

[5] Estimator $\hat{x}_k$ is unbiased if $\mathrm{E}(x_k - \hat{x}_k) = 0$ and linear if $\hat{x}_k = \mathrm{A}\begin{bmatrix} \hat{x}_{k-1} \\ y_k \end{bmatrix}$, where matrix A is arbitrary.

We conclude that KF is a good choice for the linear system even if we do not know the noise statistic exactly or what the interpretation of the problem is. Unfortunately, in the general, nonlinear measurement function case, these three problems usually produce different solutions. Therefore, in the nonlinear case, it is more important to think about the interpretation of the measurement and the state than in the linear case. In this thesis, as mentioned, we have chosen the Bayesian interpretation of the state and our goal is to compute the posterior, see Section 2. For an example of the frequentist interpretation see [91].

3.2 Kalman-type filters

Because KF works well for linear systems (3) there are many KF extensions to the nonlinear system (1). We call these extensions Kalman-type filters. Kalman-type filters approximate only the mean (estimate) and covariance (error estimate) of the posterior. If necessary, a Kalman-type filter approximates the posterior as Gaussian with those parameters.

Many Kalman-type filters are based on BLUE. Let

$$\mathrm{E}\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} \hat{x}^- \\ \hat{y} \end{bmatrix} \quad \text{and} \quad \mathrm{V}\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} \mathrm{P}_{xx} & \mathrm{P}_{xy} \\ \mathrm{P}_{yx} & \mathrm{P}_{yy} \end{bmatrix} \tag{4}$$

then the BLUE of the state x is [8]

$$\begin{aligned} \hat{x} &= \hat{x}^- + \mathrm{P}_{xy}\mathrm{P}_{yy}^{-1}\left(y - \hat{y}\right) \\ \mathrm{E}\left((x-\hat{x})(x-\hat{x})^T\right) &= \mathrm{P}_{xx} - \mathrm{P}_{xy}\mathrm{P}_{yy}^{-1}\mathrm{P}_{yx}. \end{aligned} \tag{5}$$

Kalman-type filters use BLUE recursively, so that at every time instant the Kalman-type filter approximates these expectations (4) based on previous state and error estimates and computes new state and error estimates using (5), e.g. Algorithms 2-4. Naturally, different Kalman-type filters do these approximations using different methods. Next, we present some of these approximations.

Maybe the best know KF extension is the Extended Kalman Filter (EKF) [5, 8, 22, 29, 57, 63, 71]. EKF is based on BLUE and it linearizes

Algorithm 2: Extended Kalman filter

1. Start with the mean $\hat{x}_0 = \mathrm{E}(x_0)$ and the covariance $\mathrm{P}_0 = \mathrm{V}(x_0)$ of the initial state. Set $k = 1$.
2. Prior mean of state at t_k is $\hat{x}_k^- = \mathrm{F}_{k-1}\hat{x}_{k-1}$, and prior covariance is $\mathrm{P}_k^- = \mathrm{F}_{k-1}\mathrm{P}_{k-1}\mathrm{F}_{k-1}^T + \mathrm{Q}_{k-1}$.
3. Posterior mean of state at t_k is $\hat{x}_k = \hat{x}_k^- + \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\left(y_k - \hat{y}_k\right)$ and posterior covariance is $\mathrm{P}_k = \mathrm{P}_k^- - \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\mathrm{P}_{xy_k}^T$, where

$$\begin{aligned} &\mathrm{P}_{xy_k} = \mathrm{P}_k^-\mathrm{H}_k^T, \quad \mathrm{H}_k = h_k'(\hat{x}_k^-), \\ &\mathrm{P}_{yy_k} = \mathrm{H}_k\mathrm{P}_k^-\mathrm{H}_k^T + \mathrm{R}_k \quad \text{and} \quad \hat{y}_k = h_k(\hat{x}_k^-). \end{aligned}$$

4. Increment k and repeat from 2.

the measurement function h_k (1c) around the prior mean and uses this linearization to compute the necessary expectations (4) [2, 8]. EKF algorithm for system (1) is given as Algorithm 2. This EKF Algorithm 2 is used in publications [P1–P5, R1–R6]. It is also possible to compute higher order EKF, which takes into account the higher order terms of the Taylor series of the measurement function. For example, Second Order Extended Kalman Filter (EKF2) takes into account the second order terms of the Taylor series [8]. EKF2 algorithm for system (1) is given as Algorithm 3. This EKF2 Algorithm 3 is used in publications [P1–P3, R1, R4–R6]. EKF2 uses the assumption that the posterior is Gaussian, contrary to EKF, which does not use this, usually incorrect, assumption. Modified Gaussian Second Order Filter [29, 57] is the same as EKF2.

One big drawback of EKF and the higher order EKF is that the approximation of the measurement function is computed in the prior mean estimate, so if it is incorrect then the approximation of the measurement function is incorrect and the next estimate of the prior mean is incorrect, and so on. One example of a situation where EKF estimate veers away from the true route and gets stuck in an incorrect solution branch is given in [P1]. One approach to avoiding this kind of situation is to use different linearization points that do

Algorithm 3: Second order extended Kalman filter

1. Start with the mean $\hat{x}_0 = \mathrm{E}(x_0)$ and the covariance $\mathrm{P}_0 = \mathrm{V}(x_0)$ of the initial state. Set $k = 1$.
2. Prior mean of state at t_k is $\hat{x}_k^- = \mathrm{F}_{k-1}\hat{x}_{k-1}$, and prior covariance is $\mathrm{P}_k^- = \mathrm{F}_{k-1}\mathrm{P}_{k-1}\mathrm{F}_{k-1}^T + \mathrm{Q}_{k-1}$.
3. Posterior mean of state at t_k is $\hat{x}_k = \hat{x}_k^- + \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\left(y_k - \hat{y}_k\right)$ and posterior covariance is $\mathrm{P}_k = \mathrm{P}_k^- - \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\mathrm{P}_{xy_k}^T$, where

$$\mathrm{P}_{xy_k} = \mathrm{P}_k^-\mathrm{H}_k^T, \quad \mathrm{H}_k = h_k'(\hat{x}_k^-),$$

$$\hat{y}_k = h_k(\hat{x}_k^-) + \frac{1}{2}\sum_{i=1}^{n_{y_k}} e_i \operatorname{tr}\left(\ddot{\mathrm{H}}_{k_i}\mathrm{P}_k^-\right), \quad \ddot{\mathrm{H}}_{k_i} = h_{k\,i}''(\hat{x}_k^-) \quad \text{and}$$

$$\mathrm{P}_{yy_k} = \mathrm{H}_k\mathrm{P}_k^-\mathrm{H}_k^T + \mathrm{R}_k + \frac{1}{2}\sum_{i=1}^{n_{y_k}}\sum_{j=1}^{n_{y_k}} e_i e_j^T \operatorname{tr}\left(\ddot{\mathrm{H}}_{k_i}\mathrm{P}_k^-\ddot{\mathrm{H}}_{k_j}\mathrm{P}_k^-\right).$$

4. Increment k and repeat from 2.

not depend on the prior, if possible. The name of that kind of KF extension is Linearized Kalman Filter (LKF) [22, 71]. One possible choice for the linearization point is likelihood peak(s), see [P3] or the mean of a truncated Gaussian, see [P5]. Of course, if we have more than one linearization point, we usually end up to Gaussian mixture filter, see Section 3.3. Another possibility is to generate linearization points using the state model; this idea is used in [47, 48]. It is also possible to compute EKF (or LKF), use the posterior mean estimate as a linearization point of LKF and iterate this procedure. The name of that kind of filter is Iterated Extended Kalman Filter (IEKF) [8, 29]. Note that IEKF computes the maximum a posterior estimate not the posterior mean estimate [8].

One drawback of EKF and the higher order EKF is that we have to compute the derivative of the measurement function analytically, which is not always possible or practical. Because of this, there are many derivative-free Kalman filter extensions. Note that in some cases, computing derivative analytically is not a problem at

all [R1]. Central difference filter [68], first-order divided difference filter [60] and Unscented Kalman Filter (UKF) [33, 35, 36, 37] are examples of derivative-free Kalman-type filters. These filters are also referred to as linear regression Kalman filters [49] and sigma-point Kalman filters [88]. These filters use different numerical integration methods to compute the expectations (4). For example, UKF uses so called Unscented Transformation (UT) [35] that approximates expectation values of $x \sim \mathrm{N}_n(\hat{x}, \mathrm{P})$

$$\mathrm{E}(h(x)) = \int h(\xi) p_x(\xi) \mathrm{d}\xi \approx \sum_{i=0}^{2n} \omega_i h(\chi_i), \tag{6}$$

where ω_i and χ_i are so called extended symmetric sigma-point set of distribution $\mathrm{N}_n(\hat{x}, \mathrm{P})$, which are given in Table 1.1 [36].

Table 1.1: Extended symmetric sigma-point set of $\mathrm{N}_n(\hat{x}, \mathrm{P})$

Index (i)	Weight (ω_i)	sigma-point (χ_i)
0	ω_0	$\hat{x}$
$1, \ldots, n$	$\frac{1-\omega_0}{2n}$	$\hat{x} + \sqrt{\frac{n}{1-\omega_0}\mathrm{P}}e_i$
$n+1, \ldots, 2n$	$\frac{1-\omega_0}{2n}$	$\hat{x} - \sqrt{\frac{n}{1-\omega_0}\mathrm{P}}e_{i-n}$

Approximation (6) is accurate if function h is third order polynomial [33]. In Table 1.1, the weight of the mean point $\omega_0 < 1$ is a freely selectable constant. The choice $\omega_0 = 1 - \frac{n}{3}$ is justified because it guarantees that approximation (6) is accurate for some fourth order polynomial [36]. However, if we use negative weight ω_0, it is possible to produce non-positive semidefinite posterior covariance P_k and usually this causes problems. If $\omega_0 = \frac{2}{3}$ and $n = 1$ then UT coincides with three points Gauss-Hermite rule, which has also been applied to Kalman-type filter framework; Gauss-Hermite filter [28]. UT needs $2n+1$ points and three point Gauss-Hermite rule generalization to higher dimension needs 3^n points, and thus UT and Gauss-Hermite rules coincide only in the one dimensional case [28, 36].

UKF algorithm for system (1) is given as Algorithm 4. This UKF Algorithm 4 is used in publications [P3, P4, P5, R4, R6]. The extended symmetric sigma point set used in Algorithm 4 is the conventional choice of sigma points [33, 37], but there are also other possible choices [36] such as the simplex sigma-point set, which use the minimum number of sigma-points $(n+1)$ [31]. It is also possible to scale sigma-points so that the predicted covariance is certainly positive semidefinite [30]. Algorithm 4 assumes that the state model noise w is Gaussian and exploits the linearity of the state model (1b), so it is not precisely the same as the conventional UKF. Publication [P4] extends UKF to Gaussian mixture filter framework, where every sigma point is replaced by a Gaussian distribution.

Algorithm 4: Unscented Kalman filter

1. Start with the mean $\hat{x}_0 = \mathrm{E}(x_0)$ and the covariance $\mathrm{P}_0 = \mathrm{V}(x_0)$ of the initial state. Set $k = 1$.
2. Prior mean of state at t_k is $\hat{x}_k^- = \mathrm{F}_{k-1}\hat{x}_{k-1}$, and prior covariance is $\mathrm{P}_k^- = \mathrm{F}_{k-1}\mathrm{P}_{k-1}\mathrm{F}_{k-1}^T + \mathrm{Q}_{k-1}$.
3. Posterior mean of state at t_k is $\hat{x}_k = \hat{x}_k^- + \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\left(y_k - \hat{y}_k\right)$ and posterior covariance is $\mathrm{P}_k = \mathrm{P}_k^- - \mathrm{P}_{xy_k}\mathrm{P}_{yy_k}^{-1}\mathrm{P}_{xy_k}^T$, where

   $$\mathrm{P}_{xy_k} = \sum_{i=0}^{2n} \omega_i \left(\chi_i - \hat{x}_k^-\right)\left(h_k(\chi_i) - \hat{y}_k\right)^T,\ \hat{y}_k = \sum_{i=0}^{2n} \omega_i h_k(\chi_i) \text{ and}$$

   $$\mathrm{P}_{yy_k} = \mathrm{R}_k + \sum_{i=0}^{2n} \omega_i \left(h_k(\chi_i) - \hat{y}_k\right)\left(h_k(\chi_i) - \hat{y}_k\right)^T.$$

   Here ω_i and χ_i are extended symmetric sigma-point set of distribution $\mathrm{N}_n\left(\hat{x}_k^-, \mathrm{P}_k^-\right)$ given in Table 1.1.
4. Increment k and repeat from 2.

This list of different interpretations, variations and extension of Kalman-type filters is not exhaustive. For example, it is possible to use UKF to maintain and propagate information about the higher order moments [32, 34, 87].

3.3 Nonlinear filters

If the true posterior has multiple peaks as in Figure 1, it is unlikely that a Kalman-type filter that computes only the mean and covariance achieves good performance. Hence, we have to use more sophisticated nonlinear filters. A sophisticated nonlinear filter is one that has convergence results. Possible filters are e.g. grid mass filters [45, 75, 76, R2, R3], point mass filters [9, 11, 63, 70], particle filters [6, 16, 17, 21, 24, 25, 63] and Gaussian mixture filters [4, 5, 78, 82, P3–P6]. In this section, we consider particle filters and Gaussian mixture filters.

Particle filter

A particle filter (PF) – or the sequential Monte Carlo, or Bayesian bootstrap filter – simulates a sample (particles with weights) from the posterior distribution and approximates the posterior and especially the moments of the posterior using these weighted particles. A particle filter algorithm for system (1) is given as Algorithm 5.

Algorithm 5: Particle filter

Here we denote particles with superscript $i = 1, \ldots, n_{\mathrm{pf}}$: $x_k^{(i)}$ being the ith particle at time t_k.

1. Initialise samples from the initial distribution $x_0^{(i)} \sim p_{x_0}(x)$, and assign all weights to $w_0^{(i)} = \frac{1}{n_{\mathrm{pf}}}$. Set $k = 1$.
2. Simulate particles $x_k^{(i)}$ from *proposal distribution* $p(x_k|x_{k-1}^{(i)})$ and compute weights of particles $w_k^{(i)} = w_{k-1}^{(i)} p(y_k|x_k^{(i)})$.
3. Normalize weights and compute posterior mean and covariance using particles.
4. Resample (Algorithm 6) if effective sample size is smaller than some threshold value.
5. increment k and continue from 2.

Note that Algorithm 5 is only one special case for system (1), actually quite a narrow case, of particle filters. This particle filter Algorithm 5 is used in publications [P2–P6, R2–R6]. It is well know

that without the resampling step (Step 4 in Algorithm 5) the degeneracy phenomenon occur, which means that after a certain number of recursive steps, all but one particle will have negligible normalized weights [63]. We use systematic resampling [41] (Algorithm 6) every time when the effective sample size (approximation) [42]

$$n_{\text{eff}} = \frac{1}{\sum_{i=1}^{n_{\text{pf}}} \left(w_k^{(i)}\right)^2}$$

is smaller than some threshold value n_{thr}.

Algorithm 6: Systematic resampling

Here we denote the current particles by $\bar{x}^{(i)}$ and their weights by $\bar{w}^{(i)}$. This algorithm simulates n_{pf} particles $x^{(i)}$ with equal weights $w^{(i)} = \frac{1}{n_{\text{pf}}}$ from discrete distribution defined by current particles and weights.

1. Simulate the starting point: $z_1 \sim \text{U}\left(0, \frac{1}{n_{\text{pf}}}\right]$ and set $i = 1$.
2. Compute current comparison point $z_i = z_1 + (i-1)\frac{1}{n_{\text{pf}}}$
3. Set $w^{(i)} = \frac{1}{n_{\text{pf}}}$ and $x^{(i)} = \bar{x}^{(j)}$, where j is set in such a way that $\sum_{k=1}^{j-1} \bar{w}^{(k)} < z_i \leq \sum_{k=1}^{j} \bar{w}^{(k)}$
4. Stop, if $i = n_{\text{pf}}$, otherwise increment i and continue from 2.

The particle filter has several convergence results (see e.g. [16, 17, 25]), especially weak convergence results when there are a finite number of measurements and all measurement are fixed. The definition of weak convergence is found for example in [P6, Definition 8]. Even though the particle filter is very flexible, it requires that the likelihood function $p(y_k|x_k)$ is strictly positive, because otherwise it is possible that all the weights are zero which destroys the particle filter. Unfortunately, all likelihoods of our system (1) are not strictly positive. One heuristic method of handling the situations where all weights are zero is to re-initilize particle filters using, e.g. EKF, which is what we are using in our publications, but after that the convergence results do not hold anymore.

It is also possible to use (a bank of) UKF or another Kalman-type filter to compute the proposal distribution (Step 2 in Algorithm 5)

of PF [89, 90]. Note that if we change the proposal distribution, the equation of weight will also change (Step 2 in Algorithm 5).

Gaussian mixture filter

Gaussian Mixture Filter (GMF), also called Gaussian sum filter, is a filter whose approximate prior and posterior densities are Gaussian Mixtures (GMs), a convex combination of Gaussian densities. We assume (see Section 2) that the prior and the posterior distributions have density functions, but in general GM does not necessarily have a density function [P6, Definition 4]. GMF is an extension of Kalman-type filters. One motivation to use GMF is that any density function p_x may be approximated as density function of GM p_{gm} as closely as we wish in the Lissack-Fu distance sense [14, Chapter 18][6], [52, 78]:

$$\int |p_x(x) - p_{\mathrm{gm}}(x)| \mathrm{d}x. \tag{7}$$

The outline of the conventional GMF algorithm for system (1) is given as Algorithm 7 (for detailed algorithm, see e.g. [P5, P6]). Here we assume that the initial state x_0, the state model noise w_{k-1} and the measurement noise v_k are GMs. Because the initial state and the posterior approximations are GMs then also the prior approximations at each step are GMs (Step 2), see [P6, Theorem 6].

Algorithm 7: Gaussian mixture filter

1. Start with initial state x_0. Set $k = 1$.
2. Compute prior approximation x_k^-.
3. Approximate x_k^- as a new Gaussian mixture if necessary.
4. Compute GM posterior approximation x_k.
5. Reduce the number of components.
6. Increment k and repeat from 2.

---

[6] Note that for all $\epsilon > 0$ there is a continuous density function $p_{\mathrm{c}}(x)$ with compact support such that $\int |p_x(x) - p_{\mathrm{c}}(x)| \mathrm{d}x < \epsilon$ [66, Theorem 3.14].

The heart of the GMF is the approximation of an arbitrary density function with a Gaussian mixture (Step 3). There are numerous approaches to do that. We present one conventional method. More methods are in Section 4. The density function of GM approximation p_{gm} of a density function p_x is defined as [4]

$$p_{\mathrm{gm}}(x) \propto \sum_{i=1}^{n_{\mathrm{gm}}} p_x(x_{\mathrm{g}}^{(i)}) \mathrm{N}_{c_{\mathrm{g}}\mathrm{I}}^{x_{\mathrm{g}}^{(i)}}(x), \tag{8}$$

where the mean values $x_{\mathrm{g}}^{(i)}$ are used to establish a grid in the region of the state space that contains the significant part of the probability mass, n_{gm} is the number of grid points and $c_{\mathrm{g}} > 0$ is determined so that the error in the approximation, e.g. Lissack-Fu distance (7), is minimized. It can be shown that $p_{\mathrm{gm}}(x)$ converges almost everywhere uniformly to any density function of practical concern as the number of components n_{gm} increase and c_{g} approaches zero [4, 78]. Moreover, Lissack-Fu distance (7) converges to zero. This Step (Step 3) is executed only when it is necessary. One possible criterion is to check if all prior covariances do not satisfy inequality $\mathrm{P}_i^- < \epsilon \mathrm{I}$, for some predefined ϵ, where P_i^- is the covariance of the ith component [5]. A more sophisticated method is to execute Step 3 if nonlinearity is significant [P4].

The update step of Algorithm 7 (Step 4) is usually computed as a bank of EKFs, see detailed algorithm e.g. [P6, Algorithm 2]. It is possible to compute the update step using a bank of another Kalman-type filters [P3] or bank of PFs [44]. There are also other methods of computing the posterior weights than what is in the algorithm given in [P6, Algorithm 2], e.g. methods based on quadratic programming [28]. Furthermore, in some cases, it is possible to combine Step 3 and Step 4, e.g. in EGMF [P5].

The crucial point when applying GMF to real-time implementations is the number of components GMF uses. The number of components easily explodes if we do not use some method to reduce the number of components (Step 5). The possible methods are, e.g. forgetting [78, P3, P6], merging [67, 78, P3, P6], resampling [P3], clustering [67] or minimizing some cost function, see e.g. Step 3 or [55]. One possible method for this problem is taking

a sample from the posterior and using the weighted expectation-maximization algorithm to fit an m-component GM [89], but usually this method is quite slow.

It can be shown that GMF (Algorithm 7) with some assumptions and specific approximations convergences (weakly) to the correct posterior when the number of components increases [5], [P6, Section IV]. Even if we use component reduction, especially in higher dimension, conventional approximations (8) yield GMF which is not applicable in real-time implementation. However it may be possible to derive GMF that works notably better than Kalman-type filters and uses only a few components, see Section 4 and publications [P3–P6].

4 Development of the Gaussian mixture filter

In this section, we summarize the contributions of publications [P2–P6] to the development of GMF. We use the assumptions of Section 2 except that we assume that initial state x_0, state model noise w_{k-1} and measurement noise v_k are GMs. Moreover, we use Gaussian mixture posterior approximation.

4.1 Restrictive information

Restrictive information tells that the state is inside some area A. Restrictive information can be modelled with inequality constraints [69, 72, 73]. There are many different methods of taking into account inequality constraints e.g. [23, 62, 69, 72, 73]. However, in our view, the most natural way is to model the restrictive information as a measurement (1c) with zero noise which produces likelihood which is one inside A and zero outside A [P2][7]. An example of using restriction information in such a way is in Figure 2, where on the left hand side is posterior without using restrictive information and the right hand side using it.

[7] This approach also independently proposed in [72] and [73], which were published after abstract of [P2] was submitted.

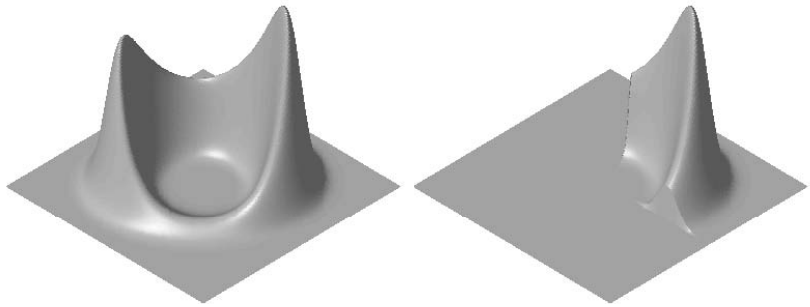

Figure 2: Left: the same as Figure 1. Right: the same posterior with 120° base station sector information.

The left hand side of Figure 2 is the same as Figure 1, where prior is Gaussian and we have two measurements, one linear and one range measurement with Gaussian noise. Restrictive information that is used on the right hand side of Figure 2 is 120° base station sector information. Note that the vertical scales of the left and right plots are different. In this case, we see that sometimes restrictive information notably improves state estimation. Because of this, it is important that we can use that kind of information with Kalman-type filters. This is the focus of publication [P2].

If the area A is restricted by two parallel planes and the distribution of the state is Gaussian (or GM), it is possible to efficiently compute the mean and covariance of the restricted state [69, 72, P2]. In publication [P2], we develop an approximation to extend this method to cases where the area A is a polyhedron and apply it to base station cell-ID information [79] (see [P2] and Figure 3).

The left hand side of Figure 3 is the same as the right hand side of Figure 2, which presents the optimal way to use 120° base station sector information. The right hand side of Figure 3 uses the method presented in publication [P2] to approximate base station sector information and after that it uses conventional EKF. When we compare the left and right hand sides of Figure 3, we see that this approximate way to use base station sector information is quite good. One advantage of this new approximation method (box-

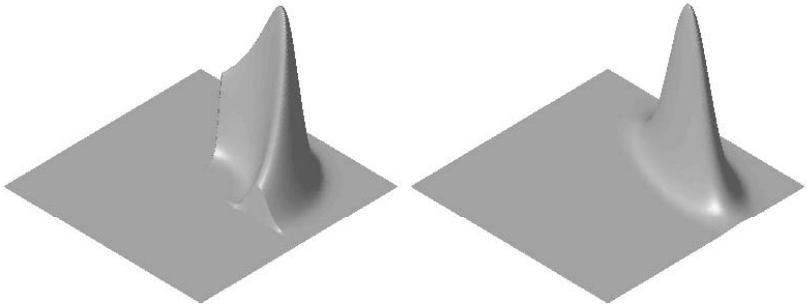

Figure 3: Left: the same as the right hand side of Figure 2. Right: posterior approximation using first approximated 120° base station sector information (see [P2]) and then EKF.

method) is that it approximates restrictive area as polyhedron $\dot{A}$ [P2, Figure 1] so that the mean and covariance of the restricted posterior can be computed using standard one dimensional Gaussian cumulative density function and basic algebra.

We use cell-ID information as restrictive information also in publications [P4, P5]. It is also possible to construct restrictive information from a digital map or floor plan [R6]. The method of using restrictive information with Kalman-type filter [P2] is a basis and an inspiration of the Box Gaussian mixture approximation of Section 4.3 and publications [P5, P6].

4.2 Approximate the likelihood as GM

In Section 4.1 we allowed the components of the measurement noise to be zero, which means that the measurement noise does not have density function. However, henceforth we assume that the measurement noise is GM with density function p_{v_k}. Based on our assumptions, the prior (approximation) is always GM. So it is convenient to consider only Gaussian prior case with Gaussian measurement noise $p_{v_k}(v) = \mathrm{N}^0_{\mathrm{R}_k}(v)$. The generalizations to the GM

cases are straightforward, see [P3, P6]. Our problem is to approximate/compute the posterior (2b)

$$p(x_k|y_{1:k}) \propto p(y_k|x_k)p(x_k|y_{1:k-1}),$$

where the likelihood is

$$p(y_k|x_k) = \mathrm{N}^0_{\mathrm{R}_k}\left(y_k - h_k(x_k)\right)$$

and the prior is

$$p(x_k|y_{1:k-1}) = \mathrm{N}^{x_k^-}_{\mathrm{P}_k^-}(x_k).$$

As we mentioned in Section 3.3, the usual method of computing the posterior is to linearize the measurement equation around the prior mean (EKF) [P3]:

$$h_k(x_k) \approx h_k(x_k^-) + h_k'(x_k^-)(x_k - x_k^-).$$

Quite often, when there are enough measurements, this is a good approximation, but when the likelihood has several peaks in the neighbourhood of the prior mean, this approximation fails. One possible way to avoid this problem is to approximate the likelihood as GM [P3]

$$p(y_k|x_k) \approx \sum_{j=1}^{n_{\mathrm{gm}}} \mathrm{N}^0_{\mathrm{R}_k}\left(y_k - h_k(z_j) - h_k'(z_j)(x_k - z_j)\right), \tag{9}$$

where z_j, $j = 1, \ldots, n_{\mathrm{gm}}$ are the likelihood peaks[8] (also referred to as static solutions). An example for using this approximation is shown in Figure 4.

The left hand side of Figure 4, which is the same as Figure 1, is the posterior when the prior is Gaussian and we have two measurements, one linear and one range measurement with Gaussian noise.

[8] If $h_k(x)$ depends only on the first d elements of $x \in \mathbb{R}^n$ and $z = \begin{bmatrix} z_{\mathrm{peak}} \\ z_{\mathrm{rest}} \end{bmatrix}$, where $z_{\mathrm{peak}} \in \mathbb{R}^d$, maximizes the likelihood function then z is acceptable likelihood peak if and only if $z_{\mathrm{rest}} = 0$. Furthermore, it is reasonable to consider only clearly distinct likelihood peaks.

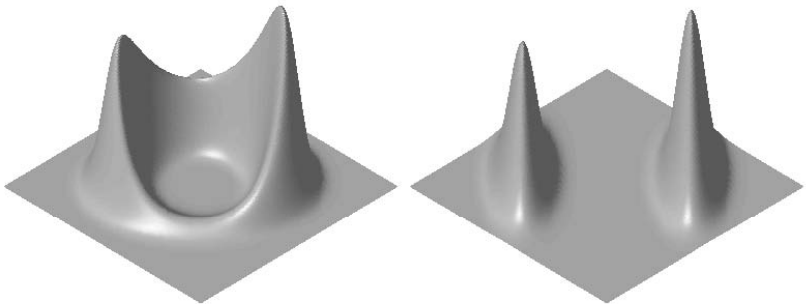

Figure 4: Left: the same as Figure 1. Right: posterior approximation using GM likelihood approximation, $\mathrm{Err}_{\mathrm{GML}} = 0.115$ (10).

The right hand side of Figure 4 is the posterior approximation using a two-component likelihood GM approximation (9). The error

$$\mathrm{Err}_{\mathrm{app}} = \sup_{\xi} |F_x(\xi) - F_{x_{\mathrm{app}}}(\xi)|, \tag{10}$$

where $F_x(\xi) = \mathrm{P}(x \leq \xi)$ is cdf of the correct posterior and $F_{x_{\mathrm{app}}}(\xi)$ is cdf of the approximated posterior. Motivation for using that error statistic come from definition of weak convergence [P6, Definition 8]. Usually this error statistic has to be computed using some multidimensional numerical integration method and so it is not very useful for extensive testing of different filters [76, Section 5]. The error of Figure 4 posterior approximation is $\mathrm{Err}_{\mathrm{GML}} = 0.115$. This error is rather small if we compare it with the error of EKF approximation $\mathrm{Err}_{\mathrm{EKF}} = 0.393$.

We show that in the hybrid positioning case, using this approximation when likelihood has several peaks outperforms Kalman-type filters such as EKF, EKF2 and UKF (see [P3]). In the hybrid positioning case, it is sometimes possible to compute the likelihood peaks in closed form [7, 74, 77], publication [P3, Appendix B] also presents some new cases where it is possible. Moreover, the method of approximating likelihoods as GM, is one way to apply multiple static position solutions in a filtering framework.

Approximating likelihood as GM has also been applied in the radar tracking applications [85, 86]; publication [86] approximates the

measurement noise v_k (1c) as GM and then converts these mixture components to the state space. It is a nice idea, but it does not work with range measurements, because small measurement noise in the range measurement does not generate small likelihood components.

It is possible to approximate almost all likelihoods inside a interesting region with sufficient accuracy as GM (see Algorithm 7 Step 3). For example in Figure 5 the likelihood of 2D range measurement with Gaussian noise is approximated as GM with 18 components using the conventional method (8).[9] However, this method usually needs a huge amount of GM components, especially in higher dimension. Furthermore, many of these components are unnecessary because after multiplying the likelihood and the prior, usually, many of these components have almost zero weight. Hence, it is not worthwhile to apply this approximation method to real-time implementations.

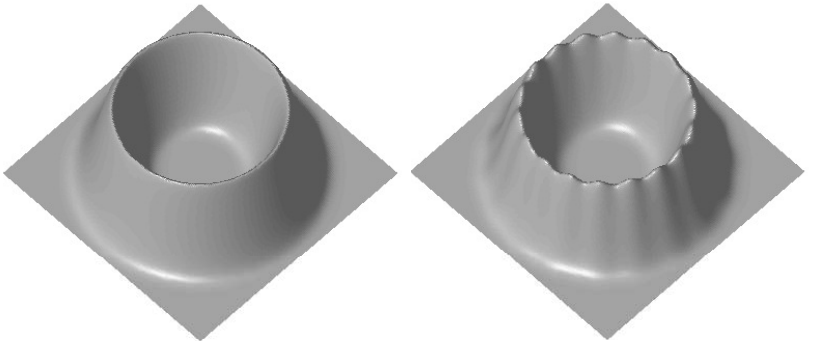

Figure 5: Left: exact likelihood of 2D range measurement with Gaussian noise. Right: GM 18 components approximation of likelihood using (8).

Publication [P3] also presents a non-Bayesian approach to robustifying GMF using likelihood peaks: if the measurements are unlikely, we add new component(s) to the posterior with appropriate weights.

[9] That kind of approximation is used as initial state GM approximation in range-only target tracking application [19].

These new components approximate the likelihood function so that the means of the components are likelihood peaks. This method is justified if there is a possibility that our prior is wrong. This method is related to Hough transformation [59] that uses the sum of likelihood functions instead of a product as in Bayesian filtering.

4.3 Approximate the prior as GM

In Section 4.2, we computed the posterior (2b)

$$p(x_k|y_{1:k}) \propto p(y_k|x_k)p(x_k|y_{1:k-1}),$$

by approximating likelihood $p(y_k|x_k)$ as GM. In this section, we use a bank of EKFs to compute the posterior approximation, but before the update step we approximate the prior $p(x_k|y_{1:k-1})$ as GM. It can be shown that if GM approximation of the prior converges to the correct prior, and the covariance of GM approximation becomes smaller, the posterior approximation of the bank of EKFs converges to the correct posterior [5, P6]. More discussion about the convergence results is in Section 4.5. However, the main interest is not in a convergence result but in developing a filter that works well also with a few mixture components. In this section, we consider two ways to approximate the prior as GM with a small number of mixture components: the Sigma Point GM approximation [P4] and the Box GM approximation [P5, P6]. Another possibility to approximate prior as GM is to approximate the state model noise w_{k-1} (1b) as GM with small covariance [4]. So if the posterior GM components have small covariances, this approximation guarantees that the next prior components have small covariances too.

Sigma Point GM approximation

The Sigma Point GM approximation (SPGMA) is an "extension" of sigma-point to GM framework. SPGMA is given in Table 1.2 [P4, P5]. In Table 1.2, we use a slightly different parametrization than in publication [P4]. Parameter $\tau \in [0,1]$ defines the size of the covariances of SPGMA. Special cases are $\tau = 0$ when SPGMA is actually the

same as the original Gaussian and $\tau = 1$ when SPGMA is the same as the extended symmetric sigma-point set of distribution $\mathrm{N}_n(\hat{x}, \mathrm{P})$, Table 1.1. Parameter α_0 is the weight of the GM component whose mean is the same as the mean of the original distribution $\hat{x}$.

Table 1.2: Sigma Point GM approximation of distribution $\mathrm{N}_n(\hat{x}, \mathrm{P})$

Index (i)	Weight (α_i)	Mean ($\hat{x}_i$)	Covariance (P_i)
0	α_0	$\hat{x}$	$(1-\tau^2)\mathrm{P}$
$1,\ldots,n$	$\frac{1-\alpha_0}{2n}$	$\hat{x} + \tau\sqrt{\frac{n}{1-\alpha_0}\mathrm{P}}e_i$	$(1-\tau^2)\mathrm{P}$
$n+1,\ldots,2n$	$\frac{1-\alpha_0}{2n}$	$\hat{x} - \tau\sqrt{\frac{n}{1-\alpha_0}\mathrm{P}}e_{i-n}$	$(1-\tau^2)\mathrm{P}$

Publication [P4] shows that SPGMA has the same mean, covariance and third moments as the original distribution $\mathrm{N}_n(\hat{x}, \mathrm{P})$ [P4, Appendix B]. An example of using this approximation is presented in Figure 6.

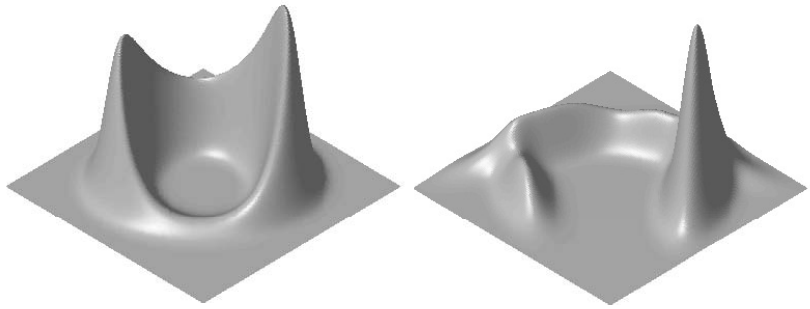

Figure 6: Left: the same as Figure 1. Right: posterior approximation using SPGM prior approximation, $\mathrm{Err}_{\mathrm{SPGMF}} = 0.236$ (10).

The left hand side of Figure 6, which is the same as Figure 1. The right hand side of Figure 6 is the posterior approximation using the SPGM prior approximation, with parameters $\alpha_0 = 0.5$ and $\tau = 0.6$. This approximation is also called the Sigma Point Gaussian Mixture Filter (SPGMF) [P5]. The error (10) of this approximation is

$\mathrm{Err}_{\mathrm{SPGMF}} = 0.236$. This error is better than the error of EKF approximation but worse than the error of GM likelihood approximation of Section 4.2. Note that SPGMA does not need the knowledge of likelihood peaks so it can be used in a wider range of situations than the GM likelihood approximation.

In publications [P4] and [P5] we show that SPGMA with the bank of EKFs (SPGMF [P5]) gives almost the same results as the particle filter in the hybrid positioning case.

Box GM approximation

The idea of the Box GM approximation (BGMA) is to partition the state space into sets A_i, where $i = 1, \ldots, n_{\mathrm{gm}}$, and approximate the distribution inside every set with one GM component using moment matching, see Table 1.3.

Table 1.3: BGMA of n-dimensional distribution, whose density function is $p_x(\xi)$. Sets A_i, where $i = 1, \ldots, n_{\mathrm{gm}}$, are the partition of $\mathbb{R}^n$.

Weight (α_i)	Mean ($\hat{x}_i$)	Covariance (P_i)
$\int_{A_i} p_x(\xi)\mathrm{d}\xi$	$\int_{A_i} x \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi$	$\int_{A_i} \left(x-\hat{x}_i\right)\left(x-\hat{x}_i\right)^T \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi$

The moment matching method guarantees that the mean and covariance of BGMA coincide with the original distribution [P6, Theorem 18]. Here we assume that the distribution is non-singular Gaussian but this basic idea is also applicable to other distributions that have a density function. The partition of the state space can be constructed, for example, using parallel planes [P5, Figure 1] or using a (bounded) polyhedron [P6, Figure 1]. One major advantage of BGMA is that we can use current measurements in an intelligent way to construct the partition of the state space. For example, with range measurement [P5] it is reasonable to align the parallel planes such that the normal vector is perpendicular to the vector $u_{\mathrm{prior}\to\mathrm{bs}}$

from the prior mean to the base station because this minimizes the effect of nonlinearity [P4, Equation (8)]. Vector $u_{\text{prior}\to\text{bs}}$ is the eigenvector of the Hessian matrix of the range measurement whose eigenvalue is 0 [R1].

An example for using BGMF [P5] is presented in Figure 7. The left hand side of Figure 7 is the same as Figure 1. The right hand side of Figure 7 is the posterior approximation using BGMA of the prior [P5], with parameter

$$l = \begin{bmatrix} -\infty & -1.04 & 1.04 & \infty \end{bmatrix} = \Phi^{-1}\left(\begin{bmatrix} 0 & 0.15 & 0.85 & 1 \end{bmatrix}\right).$$

Black lines in the Figure 7 illustrate parallel planes[10] which are using for partitioning the state space. The error (10) of this BGMF approximation is $\text{Err}_{\text{BGMF}} = 0.188$. We see that BGMF with three components gives better approximation than SPGMF with five components. So our illustrative example supports the conclusion of publication [P5] that BGMF outperforms SPGMF.

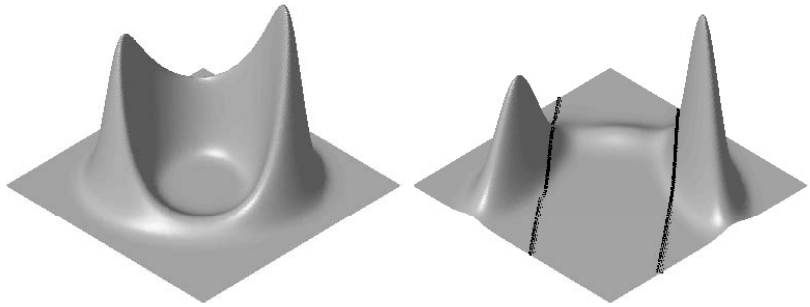

Figure 7: Left: the same as Figure 1. Right: posterior approximation using BGMA of prior, $\text{Err}_{\text{BGMF}} = 0.188$ (10). Black lines illustrate parallel planes that are using for partitioning the state space[10].

[10] These planes (lines) satisfy equation $a^T(x-\hat{x}) = \pm 1.04$ [P5, Section IV-B], where $a = \frac{n}{\sqrt{n^T \mathrm{P} n}}$, $n = \begin{bmatrix} -40 \\ 50 \end{bmatrix}$, $\mathrm{P} = \begin{bmatrix} 90^2 & 0 \\ 0 & 90^2 \end{bmatrix}$ and $\hat{x} = \begin{bmatrix} 50 \\ 40 \end{bmatrix}$ (see page 4). Note $a^T \mathrm{P} a = 1$ and $a^T u_{\text{prior}\to\text{bs}} = 0$ where (see page 24) $u_{\text{prior}\to\text{bs}} = 0 - \hat{x}$.

4.4 Approximate the posterior as GM

In Section 4.2 we computed the posterior (2b)

$$p(x_k|y_{1:k}) \propto p(y_k|x_k)p(x_k|y_{1:k-1}),$$

by approximating the likelihood $p(y_k|x_k)$ as GM, and in Section 4.3 we approximated the prior $p(x_k|y_{1:k-1})$ as GM. In this section, we directly approximate the posterior as GM. This method is called Efficient Gaussian Mixture Filter (EGMF) [P5] and it is related to BGMF when the partition (sets A_i, $i = 1,\ldots,n_{\text{gm}}$) of the state space is constructed using parallel planes, see Section 4.3. The idea of EGMF is to approximate the posterior inside every set A_i with one GM component. The ith GM component is computed using LKF, where the linearization point is the same as the ith linearization point of BGMF $\left(\int_{A_i} x\frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi \text{ Table } 1.3\right)$. The final posterior component is the moment matching approximation of the truncated (using set A_i) output of LKF (see more details in publication [P5]).[11]

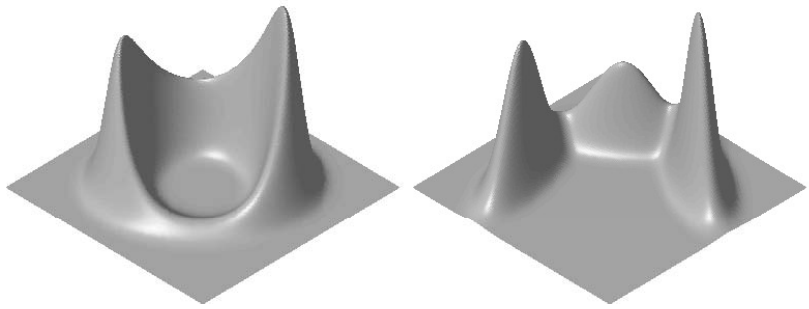

Figure 8: Left: the same as Figure 1. Right: posterior approximation using EGMF, $\text{Err}_{\text{EGMF}} = 0.177$ (10).

An example of using EGMF [P5] is presented in Figure 8. The left hand side of Figure 8 is the same as Figure 1. The right hand side

[11] If the state is constant, that is the state model (1b) is $x_k = x_{k-1}$, then it is possible to use the truncated Gaussian as a prior without approximating it as a Gaussian. Actually it is enough that the nonlinear part of the state is constant, that is the first d dimensions [P6, Equation (8)]. This approach is called the piecewise Gaussian approximation [80, 81].

of Figure 8 is the posterior approximation using EGMF [P5] with the same parameters as BGMF in Section 4.3. The error (10) of this EGMF approximation is $\mathrm{Err}_{\mathrm{EGMF}} = 0.177$. We see that EGMF outperforms BGMF (see Section 4.3 and publication [P5]).

4.5 Convergence result of BGMF

Not only does BGMA have the same mean and covariance as the original Gaussian distribution [P6, Theorem 18], but it also has convergence results. When we increase the number of GM components, BGMA converges weakly to the original Gaussian [P6, Theorem 21 and Corollary 22]. Note that the sets that form the partitioning of the state space do not have to be bounded. Thus, we can increase the number of components only in the dimensions that contribute to the measurement nonlinearity (d first dimensions, see [P6, Equation (8) and Definition 16]). We use notation $\mathrm{BGMF}_{N=}$. for a filter that uses BGMA [P6, Section V]. Here N is the parameter of BGMA [P6, Definition 16]. The number of BGMA components is $(2N^2+1)^d$. An example of using $\mathrm{BGMF}_{N=4}$ [P6] is presented in Figure 9.

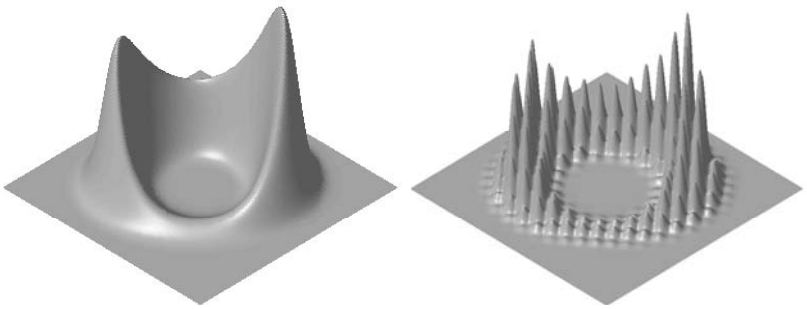

Figure 9: Left: the same as Figure 1. Right: posterior approximation using $\mathrm{BGMF}_{N=4}$, $\mathrm{Err}_{\mathrm{BGMF},\,N=4} = 0.008$ (10).

The left hand side of Figure 9 is the same as Figure 1. The right hand side of Figure 9 is the posterior approximation using $\mathrm{BGMF}_{N=4}$. The error (10) of this $\mathrm{BGMF}_{N=4}$ approximation is $\mathrm{Err}_{\mathrm{BGMF}} = 0.008$. More

results with different values of parameter N are given in Table 1.4 and in Figure 10. Table 1.4 contains also a summary of the previous results. Note that EKF is $\text{BGMF}_{N=0}$. In Table 1.4, we see that it is possible to get much better accuracy than EKF in nonlinear situation with only a few extra Gaussian components. The mean of the error of the particle filter with 10^5 particles (Algorithm 5) in this illustrative example is $\text{Err}_{\text{PF}_{n_{\text{pf}}=10^5}} = 0.017$. In this example, $\text{BGMF}_{N=3}$ having 361 Gaussian components gives better results than $\text{PF}_{n_{\text{pf}}=10^5}$. More results for hybrid positioning application are given in Section 5.

Table 1.4: Summary of different filters' performance in our illustrative problem in Figure 1.

Filter	Sec. & Pub.	n_{gm}	$\text{Err}_{\text{Filter}}$ (10)
EKF	3.2 & [P1]	1	0.393
GML	4.2 & [P3]	2	0.115
BGMF	4.3 & [P5]	3	0.188
EGMF	4.4 & [P5]	3	0.177
SPGMF	4.3 & [P4]	5	0.236
$\text{BGMF}_{N=1}$		9	0.093
$\text{BGMF}_{N=2}$		81	0.033
$\text{BGMF}_{N=3}$	4.5 & [P6]	361	0.013
$\text{BGMF}_{N=4}$		1089	0.008
$\text{BGMF}_{N=5}$		2601	0.005

In publication [P6, Section VI] we have shown that $\text{BGMF}_{N=}$. converges weakly to the correct posterior. Our illustrative example also supports convergence results (see Table 1.4 and Figure 10). Actually from Figure 10 we can see that the error (10) of BGMF in our illustrative problem approximatively satisfy the power function

$$\text{Err}_{\text{BGMF}} \approx \frac{1}{\exp(1)} n_{\text{gm}}^{-0.55}.$$

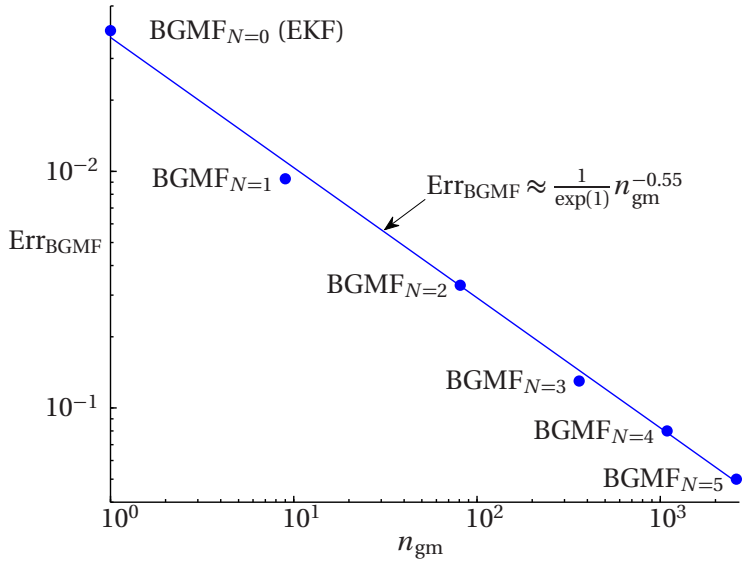

Figure 10: The error (10) of BGMF with different values of parameter N.

5 Hybrid positioning results

In this section, we briefly present some simulation results of different filters in hybrid positioning applications and some practical notes for real world implementations.

5.1 Sensitivity analysis of GMFs

In this section we study the sensitivity of the new GMF algorithms to variations of parameters. GMFs have plenty of different parameters, most of then are known and common for all GMFs, such as the threshold values for forgetting components. In this section, we consider two new parameters. The first one is SPGMF parameter τ [P4, Section 5] and the second one is parameter l. Both filters BGMF [P5, Section IV-B] and EGMF [P5, Section VI] use this l parameter.

First we recomputed the one step comparison of EKF, SPGMF, BGMF and EGMF [P5, Section VII] using a wide range of parameters τ and l. This test is also in [P4, Section 5.1]. The results are shown in Figures 11 and 12[12].

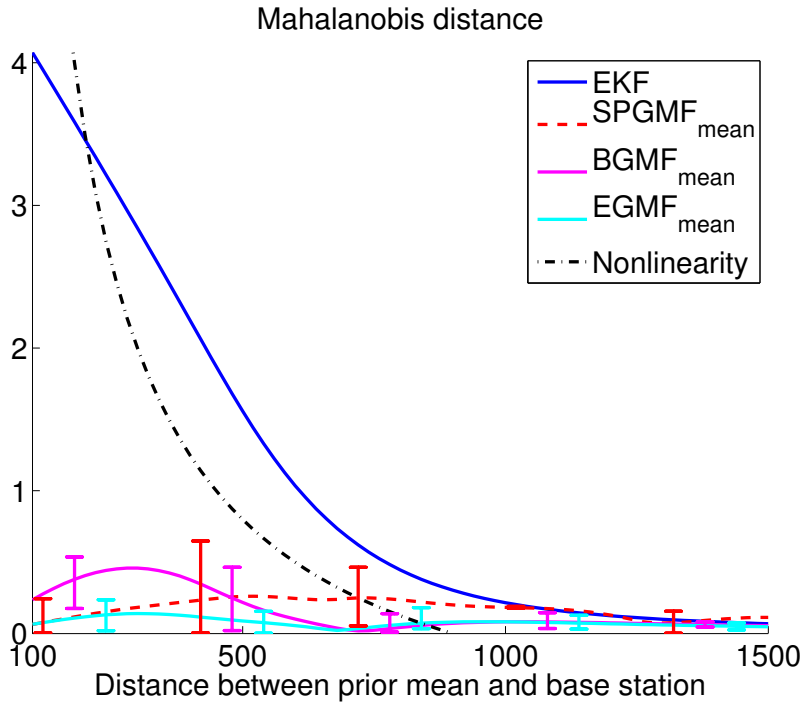

Figure 11: The results of one step comparison of EKF, SPGMF, BGMF and EGMF when the parameter of SPGMF $\tau \in [0.3, 0.7]$ and the parameter of BGMF and EGMF $\Phi(l_2) \in [0.05, 0.25]$.

In this case, we assume that parameter $\tau \in [0.3, 0.7]$ and parameter

$$l = \Phi^{-1}\left(\begin{bmatrix} 0 & \Phi(l_2) & 1-\Phi(l_2) & 1 \end{bmatrix}\right),$$

where $\Phi(l_2) \in [0.05, 0.25]$. This means, for example, that BGMA [P5, Section IV-B] splits the original Gaussian to GM with three mixture

[12] Often users are only interested in the mean of the posterior so we have computed the Mahalanobis distance between the means. Mahalanobis distance, however, does not tell everything about accuracy so we have computed also the Lissack-Fu distance.

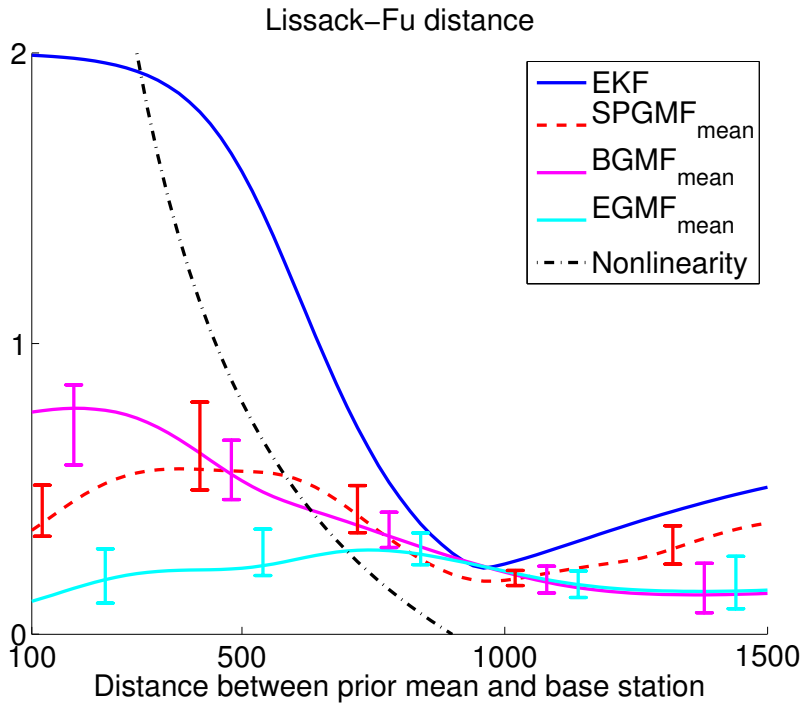

Figure 12: The results of one step comparison of EKF, SPGMF, BGMF and EGMF when the parameter of SPGMF $\tau \in [0.3, 0.7]$ and the parameter of BGMF and EGMF $\Phi(l_2) \in [0.05, 0.25]$.

components so that the weights of these components are $\Phi(l_2)$, $1-2\Phi(l_2)$ and $\Phi(l_2)$. Figures 11 and 12 show the mean statistics when the parameters are uniformly distributed in the current set. Moreover, the figures show the ranges of variation of different GMFs. Based on these figures we can say that in this case all GMFs are more accurate than EKF.

We also recomputed the simulations with "base station test bank" [P5, Section VIII-A] using parameters

$$\tau \in \left\{ \; 0.15, \quad 0.151, \quad \cdots, \quad 0.849, \quad 0.85 \; \right\}$$

and $l = \Phi^{-1}\left(\left[\begin{array}{cccc} 0 & \Phi(l_2) & 1-\Phi(l_2) & 1 \end{array} \right] \right)$, where

$$\Phi(l_2) \in \left\{ \; 0.15, \quad 0.151, \quad \cdots, \quad 0.449, \quad 0.45 \; \right\}.$$

Table 1.5: The Recomputed hybrid position simulations using a wide range of parameters τ and l. The original simulation is found in [P5, Section VIII-A].

$\text{Solver}_{\text{parameter}}$	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
EKF	10	236	465	83	6.6
$\text{SPGMF}_{\tau\in[0.15,0.85]}$	94	207 ± 12	400 ± 28	65 ± 4	3.2 ± 0.7
$\text{BGMF}_{\Phi(l_2)\in[0.15,0.45]}$	54	196 ± 7	375 ± 23	59 ± 4	3.3 ± 0.6
$\text{EGMF}_{\Phi(l_2)\in[0.15,0.45]}$	52	203 ± 15	382 ± 33	60 ± 6	2.7 ± 0.8
Ref	∞	155	287	0	0.1

We use the following threshold parameters: we forget components whose weights are less than 10^{-6}, we merge two components if the distance d_{ij} [P3, Page 834] is less than 10^{-3}, and finally, if necessary, we use resampling so that the number of components is less or equal than 40 [P3]. Note that this is much less than the number of components of the conventional GMF (8).

These simulations were made using the Personal Navigation Filter Framework (PNaFF) [R5]. The "base station test bank" contains base station range measurements, altitude measurements and restrictive information which are base station 120° sector and maximum range information. The results of these simulations are given in Table 1.5. Table 1.5 contains the range of variation of different error statistics. The main reason for using these error statistics is that it is possible to compute these error statistics in reasonable time. More discussion about these error statistics is found in publication [76, Section 5].

We see that all new GMFs: SPGMF, BGMF and EGMF give better results (in all listed criteria, except time) than EKF regardless of the values of parameters τ and l. We see that BGMF has smaller variations of results than EGMF. However, it is possible to obtain slightly better performance with EGMF than BGMF or SPGMF. More discussion about the sensitivity analysis of BGMF is presented in publication [1].

5.2 Summary of hybrid positioning results

These simulations were made using the Personal Navigation Filter Framework (PNaFF) [R5], and the test bank used is the same as the "base station test bank" of publication [P5, Section VIII-A]. See also Sectio 5.1 and [P5, Table II].

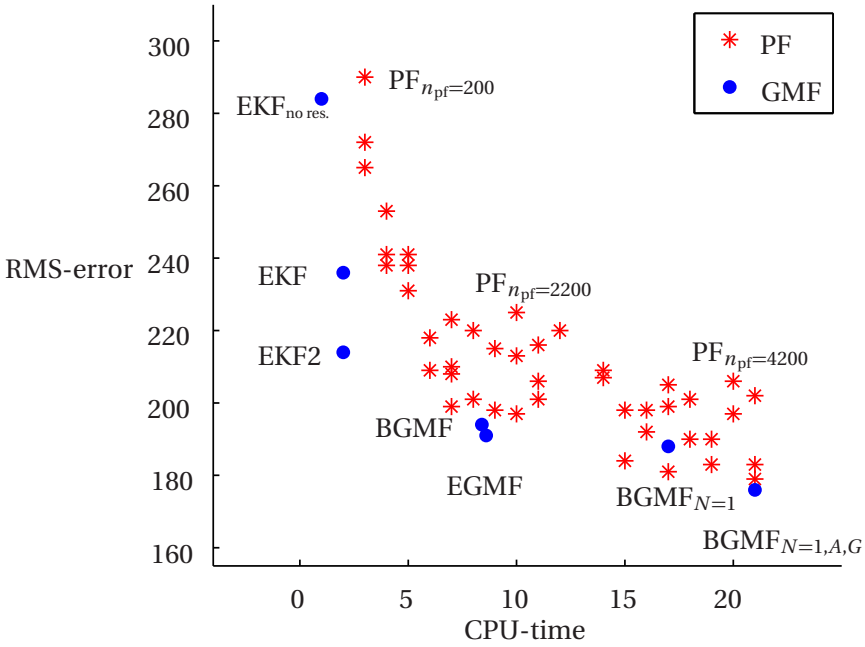

Figure 13: The RMS-error of the different particle and Gaussian mixture filters in hybrid positioning applications.

Figure 13 presents 3D root mean square position error (RMS-error) of different filters versus the relative computational time of our Matlab implementation (CPU-time). The x-axis is set to value 155 (RMS-error for PF$_{n_{\text{pf}}=10^6}$), which is nearly optimal result for RMS-error statistic [P5, Table II]. Figure 13 shows different GMF variants and the particle filter with different number of particles $n = 200, 300, 400, \ldots, 4400, 4500$. So the number of particle filters (red stars in Figure 13) is 44. EKF$_{\text{no res.}}$, EKF, EKF2, BGMF and

EGMF are explained in the publication [P5, Table II]. The difference between BGMF and $\text{BGMF}_{N=1}$ is that $\text{BGMF}_{N=1}$ uses BGMA with parameter $N = 1$ [P6, Definition 16] when it splits the prior Gaussian component to GM whereas BGMF uses the method of publication [P5]. $\text{BGMF}_{N=1,A,G}$ is the same as $\text{BGMF}_{N=1}$ but it adds mixture components that approximate the likelihood when all measurements are unlikely, and approximates the likelihood as a Gaussian mixture when the likelihood has more than one peak (for more details see [P3, P4]). The computational time is scaled so that the computation time of $\text{EKF}_{\text{no res.}}$ is 1. $\text{EKF}_{\text{no res.}}$ is the conventional EKF which does not use restrictive information. All other filters use restrictive information.

Based on publication [R4], these filters (GMFs and PFs) are the most efficient filters for hybrid positioning applications. Note that publication [R4] does not contain GMFs of publications [P4–P6]. It is also good to keep in mind that Kalman-type filters (Section 3.2) are special cases of GMFs. Positioning methods such as static positioning (positioning which uses only the current measurements), grid mass filters, point mass filters and batch least squares do not achieve the same performance as GMFs or PFs [R4].

PF is a random algorithm[13], and thus different realizations produce different results. This phenomenon can be seen in Figure 13. Because of that, we ran the particle filter 100 times, with 4500 particles using the current test bank, and got the following RMS-error: minimum 178, mean 189 and maximum 206. So, $\text{PF}_{n_{\text{pf}}=4500}$ produces almost the same RMS-error results as EGMF, whose RMS-error is 191. However, CPU-time of $\text{PF}_{n_{\text{pf}}=4500}$ is about two times the CPU-time of EGMF and the maximum memory requirement of $\text{PF}_{n_{\text{pf}}=4500}$ in our implementation is 252 kB whereas EGMF uses only 35 kB.

All in all, we see that GMF is a competitive filter for hybrid positioning applications (in our problem statement Section 2). Of course it depends heavily on the application and the comparison

[13] Strictly speaking GMFs are also random algorithms if we use resampling method for the reduction of the number of components [P3]. However the variance of GMFs is very small compared to PFs so we do not take this into account.

criteria which filter is the best and so it is possible to justify the use of some other methods such as the particle filter for some hybrid positioning applications. More hybrid positioning simulation results can be found in publications [P1–P6] and [R1–R8].

5.3 Some notes for real world implementation

In this thesis, we have considered solving the discrete-time system (Equation (1)) and we have assumed that we know the models exactly. However, usually in real world applications, lots of work is needed to find out sufficiently accurate models. It is clear that (one-component) Gaussian error models are no longer adequate for hybrid positioning in urban environments, because of, e.g. the non-line of sight effect. Solutions for these problems are, e.g. better models, robust methods [12, 54, 61] and/or interacting multiple models [58, 84]. Often these methods produce an algorithm which is a special case of GMF. However, the first necessary requirement for a good hybrid position filter is that it works with Gaussian error models. Hence it is reasonable to test filters with Gaussian error models (see Section 5.2) and consider only the filters which do not have a problem with simulated Gaussian errors.

Although our problem statement (Section 2) is very wide it does not cover all possible hybrid position applications. In "fingerprinting" [R7], for example, we do not have an analytic formula of the measurement function h (Equation (1c)) and thus, we do not have the derivative of the measurement function, which is a necessary requirement for many filters of this thesis. Of course it is possible to develop new filters for these applications. For example, we have proposed one method for using "fingerprint" data in hybrid positioning in publication [3].

6 Conclusions and future work

In this thesis, we have studied Gaussian mixture filters, especially, in the situation where the measurement function is nonlinear but

otherwise the system is linear and Gaussian. We considered hybrid positioning applications but most of the results are also applicable to other applications. One natural, popular and good question is, which filter is the best? Unfortunately, the answer is not unique. First of all we have to specify what "the best" means. A good filter gives correct solutions within some tolerance even if there are some blunder measurements and, considering personal positioning application, it is possible to implement in real-time using a mobile device. Secondly, we have seen that it depends heavily on the application and the measurements which filter is good or the best.

If we have enough measurements so that a unique static solution is almost always available, then the posterior is usually unimodal. In that case EKF work quite well and it is fast to compute and has small memory requirements. However, if nonlinearity is significant, it is good to use other Kalman-type filter, such as, EKF2, if the Hessian of the measurement function is easy to compute, otherwise UKF. If we do not have enough measurements for a unique static solution, but we have multiple static solution candidates, it is good to approximate the likelihood as GM.

In case that we have only a few measurements and we do not have static solution available, these methods do not work satisfactorily. First of all, in this hard case, it is good to use all available measurements and other information with known measurement models. For example, cell-ID information improves the performance of the filter. In the hard case, it is good to use some filter that has convergence results and adjusts the number of particles[14] so that the filter works satisfactorily. It is good to keep in mind that when we have only a few measurements, it is not possible to get the same performance as in cases where we have plenty of measurements, even if the filter works correctly. However, the hard case is very challenging for filtering and a Kalman-type filter usually does not work at all, and what is worst, a Kalman-type filter does not even know if it is failing. Because of that, it is practical, in the hard case, to use a filter which

[14] In broad sense, the word "particle" does not only mean the particle of the particle filter. The number of particles is a parameter of the filter so that when number of particles converges to infinity then the filter converges to correct posterior. So in BGMF Gaussian components are particles.

gives at least a consistent error estimate, or to wait until we have enough measurement for a good state estimate.

In this thesis, we have developed a method for using restrictive information (inequality constraints) efficiently with GMF. In the case of multiple static solutions, we have shown how to approximate the likelihood as GM and how to robustify the filter using these static solutions. In some hybrid positioning cases, the new filter, Efficient Gaussian mixture filter, outperforms other filters such as conventional Kalman-type filters as well as the particle filter. EGMF is intended to be used when the computational and memory requirements are crucial and it does not have, in general, convergence result. For the hard case, we have developed the Box Gaussian mixture filter, which is not as efficient as EGMF for a small number of components. However, BGMF converges weakly to the correct posterior at given time instant. All in all, we see that GMF is a competitive filter for hybrid positioning applications. Furthermore EKF, which works well in satellite based positioning solutions, is a special case of GMF.

There is a lot of future study left in the current GMF. For example, how to build a more efficient partitioning of the state space in BGMF, how many components are enough for some given accuracy, how to reduce the number of components more efficiently, and how to select what kind of filter to use and to make this selection adaptively. Of course, it is worthwhile to do more tests with BGMF in different scenarios and also in totally different application.

Naturally there are lots of interesting aspects in real world implementation that we do not cover in this thesis. In real world applications, it is usually necessary to use some robust method. The example of robust methods are in [12, 54, 61]. Many robust methods have been developed for Kalman-type filters. Often it is quite straightforward to apply these methods to GMF, especially when GMF has the form of the bank of Kalman-type filters. Note that BGMF has the form of the bank of Kalman-type filters but EGMF does not. However, there are still some open questions. Our robustifying method, which is presented in [P3], is quite good if we have the static solution available but it is not enough on its own. Because

of this, we need these other robustifying methods. Other current research field is how to use micro electro mechanical system inertial sensor [15] measurements in hybrid positioning, especially if we do not have enough sensors for relative position solutions. One possibility is to use the measurements of these sensors to adjust the parameters of the state model.

References

[1] S. Ali-Löytty. The sensitivity analysis of box Gaussian mixture filter. In P. Koivisto, editor, *Digest of TISE Seminar 2009*, volume 9, 2009. (submitted).

[2] S. Ali-Löytty, J. Collin, H. Leppäkoski, H. Sairo, and N. Sirola. Mathematics and methods for positioning. Lecture notes, Tampere University of Technology, 2007.

[3] S. Ali-Löytty, T. Perälä, V. Honkavirta, and R. Piché. Fingerprint Kalman filter in indoor positioning applications. In *3rd IEEE Multi-conference on Systems and Control (MSC 2009)*, July 2009. (accepted).

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

[5] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-Hall information and system sciences. Prentice-Hall, 1979.

[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, 2002.

[7] S. Bancroft. An algebraic solution of the GPS equations. *IEEE Transactions on Aerospace and Electronic Systems*, 21(7):56–59, 1985.

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

[9] N. Bergman. Bayesian Inference in Terrain Navigation. Licenciate thesis, Linköping University, 1997. Thesis No. 649.

[10] R. G. Brown and P. Y. C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering with MatLab Exercises and Solutions*. John Wiley & Sons, third edition, 1997.

[11] R. S. Bucy and K. D. Senne. Digital synthesis of non-linear filters. *Automatica*, 7(3):287–298, May 1971.

[12] A. Carosio, A. Cina, and M. Piras. The robust statistics method applied to the Kalman filter: theory and application. In *ION GNSS 18th International Technical Meeting of the Satellite Division*, September 2005.

[13] Z. Chen. Bayesian filtering: from Kalman filters to particle filters and beyond. Technical report, Adaptive Systems Laboratory, McMasters University, 2003.

[14] W. Cheney and W. Light. *A Course in Approximation Theory*. The Brooks/Cole series in advanced mathematics. Brooks/Cole Publishing Company, 2000.

[15] J. Collin. *Investigations of Self-Contained Sensors for Personal Navigation*. PhD thesis, Tampere University of Technology, 2006. URL `http://webhotel.tut.fi/library/tutdiss/pdf/collin.pdf`.

[16] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3):736–746, March 2002.

[17] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, 2001.

[18] F. Evennou, F. Marx, and E. Novakov. Map-aided indoor mobile positioning system using particle filter. *IEEE Wireless Communications and Networking Conference, 2005*, 4:2490–2494, March 2005.

[19] B. Fang and S. Wu. Angle-parameterizations range-only target tracking for scalar miss distance measurement system. In *8th International Conference on Signal Processing, 2006*, volume 1, 2006.

[20] U. Forssell, P. Hall, S. Ahlqvist, and F. Gustafsson. Map-aided positioning system. In *FISITA 2002 World Automotive Congress*, 2002.

[21] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F*, 140(2):107–113, April 1993.

[22] M. S. Grewal and A. P. Andrews. *Kalman Filtering Theory and Practice*. Information and system sciences series. Prentice-Hall, 1993.

[23] N. Gupta and R. Hauser. Kalman filtering with equality and inequality state constraints. Technical Report 07/18, Oxford University Computing Laboratory Numerical Analysis Group, Oxford, England, August 2007.

[24] K. Heine. A survey of sequential Monte Carlo methods. Licenciate thesis, Tampere University of Technology, 2005.

[25] K. Heine. *On the Stability of the Discrete Time Filter and the Uniform Convergence of Its Approximations*. PhD thesis, Tampere University of Technology, 2007. URL `http://webhotel.tut.fi/library/tutdiss/pdf/heine.pdf`.

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

[27] Y. Ho and R. Lee. A Bayesian approach to problems in stochastic estimation and control. *IEEE Transactions on Automatic Control*, 9(4):333–339, October 1964.

[28] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5):910–927, May 2000.

[29] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.

[30] S. Julier. The scaled unscented transformation. In *Proceedings of the American Control Conference, 2002*, volume 6, pages 4555–4559, 2002.

[31] S. Julier and J. Uhlmann. Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations. In *Proceedings of the American Control Conference, 2002*, volume 2, pages 887–892, 2002.

[32] S. Julier and J. Uhlmann. Comment on "A new method for the nonlinear transformation of means and covariances in filters and estimators" [authors' reply]. *IEEE Transactions on Automatic Control*, 47(8):1408–1409, August 2002.

[33] S. Julier, J. Uhlmann, and H. 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.

[34] S. J. Julier. A skewed approach to filtering. In *SPIE Conference on Signal and Data Processing of Small Targets*, volume 3373 of *SPIE proceedings series*, pages 271–282, Orlando, Florida, April 1998.

[35] S. J. Julier and J. K. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *Proceedings of AeroSense: the 11th international 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] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *American Control Conference*, volume 3, pages 1628–1632, 1995.

[38] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice-Hall information and system sciences. Prentice-Hall, 2000.

[39] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[40] W. Kim, G.-I. Jee, and J. Lee. Efficient use of digital road map in various positioning for ITS. *IEEE Position Location and Navigation Symposium*, pages 170–176, 2000.

[41] G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian non-linear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[42] 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.

[43] P. Kontkanen, P. Myllymäki, T. Roos, H. Tirri, K. Valtonen, and H. Wettig. Topics in probabilistic location estimation in wireless networks. *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004.*, 2:1052–1056, September 2004.

[44] J. Kotecha and P. Djuric. Gaussian sum particle filtering. *IEEE Transactions on Signal Processing*, 51(10):2602–2612, October 2003.

[45] S. C. Kramer and H. W. Sorenson. Recursive Bayesian estimation using piece-wise constant approximations. *Automatica*, 24(6):789–801, 1988.

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

[47] D. Lainiotis and P. Papaparaskeva. A new class of efficient adaptive nonlinear filters (ANLF). *IEEE Transactions on Signal Processing*, 46(6):1730–1737, June 1998.

[48] D. Lainiotis and P. Papaparaskeva. Efficient algorithms of clustering adaptive nonlinear filters. *IEEE Transactions on Automatic Control*, 44(7):1454–1459, July 1999.

[49] T. Lefebvre, H. Bruyninckx, and J. De Schuller. Comment on "A new method for the nonlinear transformation of means and covariances in filters and estimators" [and authors' reply]. *IEEE Transactions on Automatic Control*, 47(8):1406–1409, August 2002.

[50] T. Lefebvre, H. Bruyninckx, and J. De Schutter. Kalman filters for non-linear systems: a comparison of performance. *International Journal of Control*, 77(7), May 2004.

[51] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1067–1080, November 2007.

[52] J. T.-H. Lo. Finite-dimensional sensor orbits and optimal nonlinear filtering. *IEEE Transactions on Information Theory*, 18(5): 583–588, September 1972.

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

[54] 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):361–371, 1977.

[55] P. Maybeck and B. Smith. Multiple model tracker based on Gaussian mixture reduction for maneuvering targets in clutter. *2005 8th International Conference on Information Fusion*, 1:40–47, July 2005.

[56] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141 of *Mathematics in Science and Engineering*. Academic Press, 1979.

[57] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141-2 of *Mathematics in Science and Engineering*. Academic Press, 1982.

[58] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34(1):103–123, January 1998.

[59] A. Mikhalev and R. F. Ormondroyd. Comparison of Hough transformation and particle filter methods of emitter geolocation using fusion of TDOA data. In *4th Workshop on Positioning, Navigation and Communication 2007 (WPNC'07)*, pages 121–127, Hannover, Germany, 2007.

[60] M. Nørgaard, N. Poulsen, and O. Ravn. New developments in state estimation for nonlinear systems. *Automatica*, 36(11): 1627–1638, 2000.

[61] 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)*, pages 55–64, March 2007.

[62] P. Richards. Constrained Kalman filtering using pseudo-measurements. *IEE Colloquium on Algorithms for Target Tracking*, pages 75–79, May 1995.

[63] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Artech House, Boston, London, 2004.

[64] T. Roos, P. Myllymäki, and H. Tirri. A statistical modeling approach to location estimation. *IEEE Transactions on Mobile Computing*, 1(1):59–69, Jan-Mar 2002.

[65] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen. A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3): 155–164, July 2002.

[66] W. Rudin. *Real and Complex Analysis*. Mathematics Series. McGraw-Hill Book Company, third edition, 1987.

[67] D. J. Salmond. Mixture reduction algorithms for target tracking. *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, pages 7/1–7/4, 1989.

[68] T. Schei. A finite-difference method for linearization in nonlinear estimation algorithms. *Automatica*, 33(11):2053–2058, 1997.

[69] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura. Hand gesture estimation and model refinement using monocular camera-ambiguity limitation by inequality constraints. *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 268–273, April 1998.

[70] M. Šimandl, J. Královec, and T. Söderström. Advanced point-mass method for nonlinear state estimation. *Automatica*, 42 (7):1133–1145, July 2006.

[71] D. Simon. *Optimal State Estimation Kalman, H_∞ and Nonlinear Approaches*. John Wiley & Sons, 2006.

[72] D. Simon and D. L. Simon. Constrained Kalman filtering via density function truncation for turbofan engine health estimation. Technical Report NASA/TM–2006-214129, National Aeronautics and space administration, April 2006.

[73] D. Simon and D. L. Simon. Kalman filtering with inequality constraints for turbofan engine health estimation. *IEE Proceedings of Control Theory and Applications*, 153(3):371–378, May 2006.

[74] N. Sirola. A versatile algorithm for local positioning in closed form. In *Proceedings of the 8th European Navigation Conference GNSS 2004*, Rotterdam, May 2004.

[75] N. Sirola. Nonlinear filtering with piecewise probability densities. Research report 87, Tampere University of Technology, 2007.

[76] N. Sirola. *Mathematical Methods in Personal Positioning and Navigation*. PhD thesis, Tampere University of Technology, August 2007. URL `http://webhotel.tut.fi/library/tutdiss/pdf/sirola.pdf`.

[77] N. Sirola, R. Piché, and J. Syrjärinne. Closed-form solutions for hybrid cellular/GPS positioning. In *Proceedings of the ION GPS/GNSS 2003*, pages 1613–1619, 2003.

[78] H. W. Sorenson and D. L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7(4):465–479, July 1971.

[79] M. A. Spirito, S. Pöykkö, and O. Knuuttila. Experimental performance of methods to estimate the location of legacy handsets in GSM. *IEEE Vehicular Technology Conference*, pages 2716–2720, 2001.

[80] O. Stepanov. *Nonlinear Filtering and Its Application in Navigation*. CSRI Elektropribor, Saint Petersburg, Russia, 1998. in Russian.

[81] O. Stepanov. Comparative investigation of two nonlinear filters for navigation problems. *IEEE Position Location and Navigation Symposium (PLANS2000)*, pages 333–340, 2000.

[82] S. Stergiopoulos, editor. *Advanced Signal Processing Handbook: Theory and Implementation for Radar, Sonar, and Medical Imaging Real-Time Systems*. The electrical engineering and signal processing series. CRC Press, 2000.

[83] J. Syrjärinne. *Studies of modern techniques for personal positioning*. PhD thesis, Tampere University of Technology, Tampere, 2001. Publications 319.

[84] J. Syrjärinne and J. Saarinen. An evaluation of motion model structures within the IMM frame using range-only measurements. In *Proceedings of International Conference on Artificial Intelligence*, volume I, pages 254–260, 1999.

[85] W. I. Tam and D. Hatzinakos. An adaptive Gaussian sum algorithm for radar tracking. In *Proceedings of ICC'97 - International Conference on Communications*, pages 1351–1355, 1997.

[86] W. I. Tam, K. Plataniotis, and D. Hatzinakos. An adaptive Gaussian sum algorithm for radar tracking. *Signal Processing*, 77: 85–104, 1999.

[87] D. Tenne and T. Singh. The higher order unscented filter. *Proceedings of the American Control Conference, 2003.*, 3:2441–2446, June 2003.

[88] R. van der Merwe. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. PhD thesis, OGI School of Science & Engineering at Oregon Health & Science University, April 2004.

[89] R. van der Merwe and E. Wan. Gaussian mixture sigma-point particle filters for sequential probabilistic inference in dynamic state-space models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, volume 6, pages VI–701–4, April 2003.

[90] R. van der Merwe, A. Doucet, J. F. G. de Freitas, and E. Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, 2000.

[91] P. Xu. Biases and accuracy of, and an alternative to, discrete nonlinear filters. *Journal of Geodesy*, 73:35–46, 1999.

PUBLICATION 1

Simo Ali-Löytty, Niilo Sirola and Robert Piché: Consistency of three Kalman filter extensions in hybrid navigation. In *European Journal of Navigation*, Volume 4, Number 1, February 2006.

SCIENCE

Consistency of Three Kalman Filter Extensions in Hybrid Navigation

By Simo Ali-Löytty, Niilo Sirola and Robert Piché, Institute of Mathematics, Tampere University of Technology, Finland, e-mail: simo.ali-loytty@tut.fi

A filter is *consistent* if predicted errors are at least as large as actual errors. In this paper, we evaluate the consistency of three filters and illustrate what could happen if filters are inconsistent. Our application is hybrid positioning which is based on signals from satellites and from mobile phone network base stations. Examples show that the consistency of a filter is very important. We evaluate three filters: EKF, EKF2 and PKF. Extended Kalman Filter (EKF) solves the filtering problem by linearising functions. EKF is very commonly used in satellite-based positioning and it has also been applied in hybrid positioning. We show that nonlinearities are insignificant in satellite measurements but often significant in base station measurements. Because of this, we also apply Second Order Extended Kalman Filter (EKF2) in hybrid positioning. EKF2 is an elaboration of EKF that takes into consideration the nonlinearity of the measurement models. The third filter is called Position Kalman Filter (PKF), which filters a sequence of static positions and velocities. We also check what kind of measurement combinations satisfy CGALIES and FCC requirements for location.

Introduction

In navigation, filters are used to compute an estimate of the position using current and past measurement data. When the number of measurements is insufficient to specify a unique position solution, the filtered estimate may veer away from the true route and get stuck in an incorrect solution branch. For filters to avoid and recover from such mistakes, it is important that they are *consistent*, that is, their predicted errors should be at least as large as actual errors. In section Filters; Inconsistency test, we define a filter consistency and introduce a test for filter inconsistency. In this study, we evaluate the consistency of three filters which are representative of the Kalman type filters used in hybrid navigation based on signals from satellites and from mobile phone network base stations. The reason why we concentrate on Kalman type filters is that they are fast to compute. It is possible, for example with particle filters and grid-based methods, to get a more accurate estimate, but such filters are slower and because of this they do not yet suit mobile positioning.

Problem Statement

Target

In this section, we introduce our problem. First of all, we model the user state x and measurements y as stochastic processes. We have an initial state x_0 and a dynamic system

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1} \qquad (1)$$
$$y_k = h_k(x_k) + v_k,$$

where subscript k represents time moment t_k, $k = 1, 2, \ldots$. We assume that all errors (w_k and v_k) are zero mean, white and independent. The aim is to solve state conditional probability density functions (cpdf) when conditional are past and current measurements. These densities are usually called posterior densities. If conditional are only past measurements then the density functions are usually called prior densities. Knowledge of the posterior density enables one to compute an optimal state estimate with respect to any criterion. For example, the Minimum Mean-square Error (MMSE) estimate is the conditional mean of x_k (Bar-Shalom et al., 2001) (Ristic, et al., 2004). In general and in our case the conditional probability density function cannot be determined analytically.

In our applications, the state $x = \left[r_u^T, v_u^T\right]^T$ consists of user position vector r_u and user velocity vector v_u. State dynamics are specified in (2). Measurements

y consist of satellite pseudorange ρ_s and delta pseudorange $\dot{\rho}_s$ measurements, and base station range measurements ρ_b. We do not model user clock because it is quite difficult to get a realistic clock model. Therefore, we must use the difference measurements of satellites.

State dynamics

We use the Position-Velocity (PV) model, where the user velocity is a random walk process (Brown, 1983). The user state is the solution of a stochastic differential equation which can be written as a difference equation (Maybeck, 1979)

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1}, \tag{2}$$

where

$$\Phi_{k-1} = \begin{bmatrix} I & \Delta t_k I \\ 0 & I \end{bmatrix}, \tag{3}$$

and w_{k-1} is white, zero mean and Gaussian noise, so that

$$Q_{k-1} = \mathrm{V}(w_{k-1}) = \begin{bmatrix} \frac{\Delta t_k^3}{3} Q_c & \frac{\Delta t_k^2}{2} Q_c \\ \frac{\Delta t_k^2}{2} Q_c & \Delta t_k Q_c \end{bmatrix}, \tag{4}$$

$$Q_c = \begin{bmatrix} \sigma_{plane}^2 I_{2\times 2} & 0 \\ 0 & \sigma_{altitude}^2 \end{bmatrix}, \tag{5}$$

where σ_{plane}^2 represents the velocity errors in the East-North plane and $\sigma_{altitude}^2$ represents the velocity errors in the vertical direction.

Measurement equation

In this study, we use quite general measurement equations and we do not restrict ourselves on any specific satellite system (e.g. GPS, Galileo) or mobile phone network (e.g. GSM, 3G).

Satellite measurements

From satellites we get usually two different measurements: PseudoRange (PR) and Delta PseudoRange (DPR) measurements (Kaplan, 1996). The measurement equations are

$$\rho_s = \|r_s - r_u\| + b + \varepsilon_{\rho_s}, \tag{6}$$

$$\dot{\rho}_s = \frac{(r_s - r_u)^T}{\|r_s - r_u\|}(v_s - v_u) + \dot{b} + \varepsilon_{\dot{\rho}_s}, \tag{7}$$

where r_s is the satellite position vector, v_s is the satellite velocity vector, b is clock bias in metres, $\dot{b}$ is clock drift, ε_{ρ_s} is PR error term and $\varepsilon_{\dot{\rho}_s}$ is DPR error term. Let n_s be the number of satellites.

We suppose that $n_s \geq 2$, otherwise we do not use satellite measurements. We get difference measurements when we subtract for example last satellite (PR or DPR) measurements from other measurements.

Base station measurements

There are many possible different measurements in base station positioning, such as Time Of Arrival (TOA), Round Trip Delay (RTD), Received Signal Strength (RSS), Time Difference Of Arrival (TDOA), Angle Of Arrival (AOA), and Cell IDentity (Cell-ID). (Drane et al., 1998) (Spirito et al., 2001) (Syrjärinne, 2001) (Vossiek et al., 2003). We can also use cell average altitude for positioning if it is known. TOA, RTD and RSS have the same mathematical form

$$\rho_b = \|r_b - r_u\| + \varepsilon_{\rho_b} \tag{8}$$

where r_b is the base station position vector and ε_{ρ_b} is error term. In this study, we do not use AOA or Cell-ID measurements.

Initial state

Filtering is performed within the framework of sequential Bayesian estimation, which requires the initial state x_0. We assume that this initial state is Gaussian so with mean x_0 and covariance matrix P_0. One possibility to get the initial state is to use the first position and velocity solutions (see Section Filters, PKF).

Filters

The Kalman filter extensions considered are Extended Kalman Filter (EKF), Second Order Extended Kalman Filter (EKF2) and Position Kalman Filter (PKF). The common feature of these filters and one reason why we call these filters Kalman filter extensions is that these filters 'remember' and use only the last state mean and covariance matrices. These filters solve approximately the filtering problem (see section Problem Statement).

EKF

The Extended Kalman Filter solves the filtering problem by linearising the measurement function. EKF is very commonly used in satellite-based positioning and it has also been applied in hybrid positioning (Ma, 2003). The EKF algorithm is described, for example, in (Bar-Shalom et al, 2001) or (Jazwinski, 1970).

EKF2

The Second Order Extended Kalman Filter is an elaboration of EKF that takes into consideration the nonlinearity of the measurement models. In particular EKF2 increases measurement variance with extra variance which is

$$\sigma^2_{extra} = \frac{1}{2}\mathrm{tr}\left(H_e P^- H_e P^-\right) \quad (9)$$

where H_e is the Hessian matrix of measurement and P^- is covariance matrix of prior density. Modified Gaussian Second Order Filter (Jazwinski, 1970) (Maybeck, 1982) is the same as EKF2. The EKF2 algorithm is given for example in (Bar-Shalom et al., 2001).

PKF

The Position Kalman Filter works by filtering a sequence of static position and velocity solutions. The idea is that we first solve position and velocity and then filter these; this idea is called two-stage estimator (Chaffee and Abel 1992). When we have enough measurements, we can find position and velocity for example by Weighted Least Squares (WLS) or closed-form methods (Sirola et al., 2003). WLS also approximates the errors of solution. At times instants when there are not enough measurements to fix position and velocity, we do not use any measurements. The PKF filter algorithm is Kalman filter algorithm (Jazwinski, 1970), but it is important to notice that errors of position solutions are not independent from state dynamics errors, so the system does not fulfill the assumptions of Kalman Filter.

Nonlinearities

Our model has nonlinearities only in the measurement model. We say that the measurement nonlinearities are significant if they are comparable to, or larger than, the measurement noise (Jazwinski, 1970)

$$\sigma^2_{extra} \geq \sigma^2_{meas}. \quad (10)$$

First we study the significance of nonlinearities in satellite measurements. Now, from (9), we have

$$\sigma^2_{extra} \leq 3\left\|P^-\right\|^2 \left\|H_e\right\|^2. \quad (11)$$

We assume that user velocity is less than 5,000 (m/s), and $\left\|P^-\right\| \leq 100^2$. These two approximations are quite overpessimistic in personal satellite positioning. We know that $\left\|r_s - r_u\right\| \geq 20{,}000$ (km) and $\left\|v_s\right\| \leq 5.000$ (m/s) (Kaplan, 1996). Thus we get

$$\sigma^2_{extra} \leq 3\left\|P^-\right\|^2 \frac{144}{\left\|r^s - r_u\right\|^2} \leq 0.000108. \quad (12)$$

In our simulation when we consider satellite measurement $\sigma^2_{meas} \geq 0.1^2$. So equation (10) does not hold and there are no significant nonlinearities measurements from one satellite. The same results holds if we have more than one for example ten satellite measurements each time instant.

The difference between σ^2_{extra} and σ^2_{meas} is even bigger if we use only satellite pseudorange measurements. Because of this, we linearise all satellites measurements and use those in our simulations.

Secondly, we concentrate on base station measurements nonlinearities. We assume that $P^- = \left\|P^-\right\| I$. Now

$$\sigma^2_{extra} \geq \frac{\left\|P^-\right\|^2}{\left\|r_b - r_u\right\|^2}. \quad (13)$$

In our simulation the variances of base station measurements are 80^2. It is very common in urban areas that the distance from base station is very short, say 250m, but the estimation error can be quite large, say $\left\|P^-\right\| = 150^2$. We get

$$\sigma^2_{extra} \geq 90^2 > 80^2 = \sigma^2_{meas}, \quad (14)$$

so there are in some cases significant nonlinearities in base station range measurements.

Inconsistency tests

We say that filter is consistent if inequality

$$\mathrm{E}\left[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\right] \leq P_k \quad (15)$$

holds in every $k = 1, 2, \ldots$ (Lefebvre, et al., 2004). The Normalised Estimation Error Squared test (NEES) and the Normalised Innovation Squared test (NIS) (Bar-Shalom et al. 2001) are popular ways of testing filter inconsistency (Lefebvre et al., 2004) (Lerro and Bar-Shalom, 1993) (Schlosser and Kroschel, 2004). Both NEES and NIS tests assume that distributions are Gaussian, so that test statistics are chi-square distributed. However we know that distributions are usually not Gaussian. Because of this we apply Chebyshev's inequality to our hypothesis testing. Our null hypothesis is

$$\mathrm{H}_0 : \mathrm{E}\left[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\right] \leq P_k. \quad (16)$$

We assume that $\det(P_k) \neq 0$. Now there is a matrix A_k such that $P_k = A_k^{-1}\left(A_k^T\right)^{-1}$. Hypothesis test statistic is

$$\left\|\psi_k\right\| = \left\|A_k(x_k - \hat{x}_k)\right\|. \quad (17)$$

If H_0 is true then

$$\begin{aligned} n &= \mathrm{tr}\left(A_k P_k A_k^T\right) \geq \mathrm{E}\left[tr\left(\psi_k \psi_k^T\right)\right] \\ &\geq \int_{\left\|\psi_k\right\| \geq \varepsilon} \left\|\psi_k\right\|^2 f(x_k) dx_k \\ &\geq \varepsilon^2 \mathrm{P}\left(\left\|\psi_k\right\| \geq \varepsilon\right), \end{aligned} \quad (18)$$

where n is the dimension of x_k. If α is the risk level then

$$\mathrm{P}\left(\left\|\psi_k\right\| \geq \sqrt{\frac{n}{\alpha}}\right) \leq \alpha \quad (19)$$

So if $\|\psi_k\| \geq \sqrt{\frac{n}{\alpha}}$ is true, it indicates that the null hypothesis can be rejected at the risk level of α. We call this test *the general inconsistency test*. If we also assume that $x_k - \hat{x}_k$ is zero mean and Gaussian then this is the above-mentioned NEES-test.

Simulations

In our simulations, we use East-North-Up (ENU) coordinate system. We assume that errors are zero mean, independent Gaussian white noise, with

$$\sigma_{\rho_r} = 10\text{m}, \sigma_{\dot{\rho}_r} = 0.1\frac{\text{m}}{\text{s}} \text{ and } \sigma_{\rho_s} = 80\text{m} \quad (20)$$

As parameters of state dynamics we use

$$\sigma^2_{plane} = 2\frac{\text{m}^2}{\text{s}^3}, \quad \sigma^2_{altitude} = 1\frac{\text{m}^2}{\text{s}^3}. \quad (21)$$

The first parameter is the same as in (Ma, 2003). The mean of initial state is $x_0 = [r_0^T, 0^T]^T$, r_0 is true place at time t_0 and the covariance matrix of initial state is

$$P_0 = \begin{bmatrix} 100^2 I & 0 \\ 0 & 10^2 I \end{bmatrix} \quad (22)$$

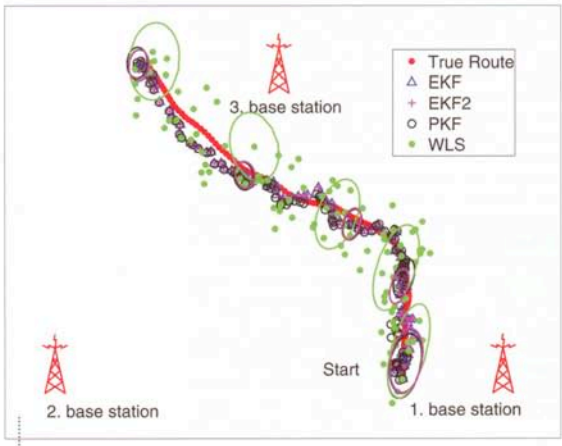

Figure 1 - Simulation example with measurements from three base station and altitude information.

The positions and velocities of satellites are based on real ephemeris data. Ephemeris data was measured in the Tampere University of Technology campus region on 18th June 2003. Base station positions are [1000, 0, 0], [-1000, 0, 0] and [0, 1000, 0], see Figure 1. Measurements and true route are simulated using measurements equations (see section Problem Statement, Measurement equation) and state dynamics (2) with given parameters. In some tests we simulate also altitude measurements. The altitude measurements are modelled as same kind of measurements as base station range measurements, with the 'base station' in the centre of Earth.

A typical simulation is shown Figure 1, which shows base stations, true route (red, 2 minutes), filters mean: PKF (black), EKF (blue), EKF2 (magenta), and covariance ellipses which illustrate dispersion of distribution. The covariance ellipses satisfy the equation

$$(x - \hat{x})^T P_{2\times 2}^{-1} (x - \hat{x}) = 2.2173. \quad (23)$$

The constant was chosen so that if the distribution is Gaussian then there is 67% probability mass inside the ellipse. Ellipses are drawn every half minute. In this example, we get three base stations range measurements and altitude information every second.

From Figure 1 we can see that filters work very similarly in this case and that they give much better position estimates than the static WLS solutions. Table 1 shows that in many cases all filters work very similarly.

We generated one hundred true routes and for every true route 10x18 measurement sets so that every Satellite (SV) and Base Station (BS) set have ten measurement sets for every true route. The satellites and base stations are the same throughout all simulations, although we would get better

Meas.		PKF		EKF		EKF2	
SV	BS	67%	95%	67%	95%	67%	95%
0	1	944	2448	731	2731	764	2160
0	2	944	2448	212	2243	403	2038
0	3	61	117	56	104	56	104
2	Ø	944	2448	710	2119	710	2119
2	0	944	2448	698	2117	698	2117
2	1	944	2448	59	219	60	214
2	2	37	74	35	72	35	72
2	3	35	70	33	69	33	69
3	Ø	944	2448	247	710	247	710
3	0	18	31	14	26	14	26
3	1	18	30	13	25	13	25
3	2	17	29	12	24	12	24
3	3	17	29	12	23	12	23
4	Ø	5	9	5	9	5	9
4	0	5	9	5	9	5	9
4	1	4	8	4	8	4	8
4	2	4	9	4	9	4	9
4	3	4	8	4	8	4	8

Table 1 - Two-dimensional error limits for filters, in simulations have 100 true routes and every true route was run 10 times.

results if we used different satellites and base stations in different time instants. Tables 1 and 2 have error limits from these simulations. Filters' error limits are in Table 1 and WLS error limits are in Table 2, where we also have a column which tells how many times WLS does not have a unique position solution. In these tables the 67% column gives limit so that 67% of all 2 dimensional errors (norm of true position minus posterior mean) are smaller than this limit, and similarly for the 95% column. Every measurement set has also altitude measurement, except the measurement sets where BS column has Ø sign.

In Table 1 we see that EKF and EKF2 errors have noticeable difference only when we have only one or two base station range measurements. If we consider 95% errors limits we see that EKF2 gives better position estimates than EKF. Actually, when we have only one base station range measurement, PKF gives better estimate than EKF although PKF has no position solutions (see Table 2). We see in section Simulations, Consistency that in this case EKF is very often inconsistent. It is useful to notice that the ratio of 95% error limits and 67% error limits is usually about two. There are however some

Meas.		WLS		no unique WLS
SV	BS	67%	95%	solution (%)
0	1	•	•	100
0	2	•	•	100
0	3	150	•	7
2	Ø	•	•	100
2	0	•	•	100
2	1	•	•	100
2	2	85	190	1
2	3	81	187	2
3	Ø	•	•	100
3	0	39	68	0
3	1	36	65	1
3	2	35	62	1
3	3	33	61	2
4	Ø	20	36	0
4	0	20	36	0
4	1	20	36	0
4	2	20	36	0
4	3	20	35	0

Table 2 - Two dimensional error limits for WLS and how many times we do not have unique WLS solution.

cases, especially (0,2) and (2,1), (SV, BS) combinations, where this ratio is much bigger. We see that in all of these cases where this ratio is about three or bigger there is a problem with consistency (see section Simulations, Consistency). PKF has noticeable difference with respect to EKF or EKF2 only when there are no unique position solutions. We can also see that even quite rough altitude information gives much better position estimates than without that information.

In Table 2, we see that WLS position solutions' (estimated) errors are many times larger than estimated errors of filters, so it is reasonable to use statistical approach (filters) with as many measurements as possible if we want more accurate position estimates.

Requirements of CGALIES and FCC

Both American federal Communications Commission (FCC) and European Coordination Group on Access to Location Information for Emergency Services (CGALIES) have their own requirements for emergency call positioning accuracy. These requirements are briefly summarised in Tables 3 and 4.

The limits in Table 3 are intended for 67% of calls, but in the future the requirements may be tightened to apply to 95% of calls. A summary of CGALIES 67% accuracy requirements is approximately 150m in urban environment and 500m in rural environment (Ludden et al., 2002). The main difference between urban and rural environments is that in urban environment there is a high density of mobile phone base stations but the satellite visibility is reduced. Vice versa, in rural environment there is good satellite visibility but the mobile phone base stations are very sparsely laid out (Heinrichs et al., 2004).

If we compare CGALIES and FCC requirements and our simulation results, we can conclude that WLS estimates satisfy all requirements only if they use at

CGALIES	Urban	Rural
Caller cannot provide	10 – 150m	10 – 500m
any information	(10 – 50m)	(10 – 100m)
Caller can provide	25 – 150m	50 – 500m
general information	(10 – 50m)	(30 – 100m)

Table 3 - CGALIES requirements for location (values indicated between parenthesis correspond to the requirements obtained through a questionnaire to the Member States) (Ludden et al., 2002).

FCC	Accuracy for 67% of calls	Accuracy for 95% of calls
Handset-based	50m	150m
Network-based	100m	300m

Table 4 - FCC requirements for location (FCC).

least three satellite measurements and altitude information. WLS estimates satisfy CGALIES and FCC network-based requirements if they use measurements from two satellites, altitude information and at least two base stations. WLS satisfy CGALIES requirements by a whisker with altitude information and measurements from three base stations.

With the above-mentioned measurement sets, all three filters satisfy almost all requirements. Only the FCC handset based requirement of accuracy for 67% of calls is not fulfilled with altitude information and measurements from three base stations. EKF and EKF2 satisfy also CGALIES and FCC network-based requirements with altitude information and measurements from two satellites and one base station. Furthermore, EKF and EKF2 satisfy CGALIES rural environment requirement with measurements from three satellites or two base stations and altitude information. Although in our simulation, using measurements from only one base station does not satisfy any requirements, it is possible that in urban environments with very high base station density these requirements can be satisfied. This is possible if base stations are always sufficiently close to users. Of course, then the base station where measurements come from must be changed quite often.

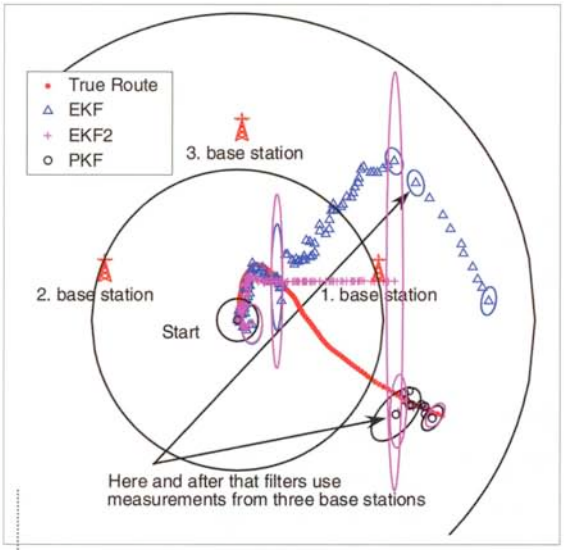

Figure 2 - EKF estimate veers away from the true route and gets stuck in an incorrect solution branch. EKF2 increases covariance matrix.

Consistency

In our simulations, we saw that there are consistency problems with some base station and satellite measurement combinations. Now we study whether the filters work correctly or not. One way to test this is to use the general inconsistency test, which tells if true estimate errors coincide with the error covariance matrix given by the filter (see section Filters, Inconsistency tests).

Figure 2 shows one case where EKF estimate veers away from the true route and gets stuck in an incorrect solution branch. In this case, filters get measurements from two base stations and altitude information for the first 110 seconds. During the last 10 seconds, there is an additional base station measurement, making unique WLS position fixes possible.

The covariance ellipses as defined in (23) are drawn at time instants t_0, t_{60}, t_{110}, t_{111} and t_{120} . We see that here EKF covariance matrices are very small if we compare them to true estimates errors. Because of this, EKF is inconsistent which is also verified by the general inconsistency test (see Figure 3). Note that in this kind of situation NIS test is not relevant, for example in this case EKF passes NIS test when it uses measurements only from two base stations and altitude information. From Figure 2, we can also see that after the third base station becomes available at t_{110}, it takes a long time for EKF to become consistent again.

In this example, the covariance matrices of EKF2 are large enough so that EKF2 is not inconsistent (see Figure 3). Also, although the PKF estimate re-

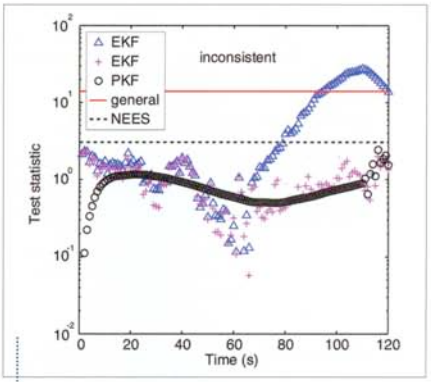

Figure 3 - Inconsistency test of the situation of Figure 2. In this case, only EKF fails this test, test statistic is bigger than red line. Black dashed line is NEES test. Risk level of both test is $\alpha = 0.01$.

mains at the starting point for the first 110 seconds, the filter is not inconsistent because its covariance grows fast enough.

It is useful to notice that both EKF2 and PKF start giving accurate estimates as soon as the third base station becomes available. Note also that although both filters EKF2 and PKF are not inconsistent in this case, we can say that EKF2 gives better (more accurate) estimates.

Meas.		EKF		EKF2	
SV	BS	α=5%	α=1%	α=5%	α=1%
0	1	71	50	14	9
0	2	40	32	15	15
2	Ø	6	0	6	0
2	0	6	0	6	0
2	1	6	5	5	4
3	Ø	4	0	4	0

Table 5 - Percentage of times filters are inconsistent (the general inconsistency test), same simulations than Table 1.

All in all, we can observe from the foregoing example that it is very important that a filter be consistent. Now we study more precisely how often and with which measurement combinations filters are inconsistent. We use the same simulations as in Table 1. All general inconsistency test results which are non-zero are tabulated in Table 5. Tabulated quantity tells how many percent of 1,000 runs have at least one time instant in which the filter is inconsistent at some risk level. We use both 5% and 1% risk level.

First of all, PKF never has problems with consistency and because of this PKF is not mentioned in Table 5. From Tables 5 and 2 we can see that EKF and EKF2 only have problems with consistency when the system is underdetermined (when there is no unique position solution). These problems are significant when there are base station measurements. EKF2 works better than EKF. However, EKF2 does have some problems with consistency. One reason is that although EKF2 takes into consideration the nonlinearity of the base station range measurement models, there are still small errors. These small errors can accumulate because the measurement model depends on the previous state estimate. EKF also has this same problem that measurement model depends on the previous state estimate. PKF does not have this problem and this is one reason why PKF never fails this test.

Conclusions

In this article, we have shown that nonlinearities are insignificant in satellite measurements, but often significant in base station measurements. In these cases EKF2 works better than EKF, and the 95% error limits of EKF2 are smaller and EKF2 is not so often inconsistent. Unfortunately, EKF2 also is sometimes inconsistent. PKF does not have a problem with consistency, but the error limit of PKF is usually larger than in other filters. Our simulations results indicate that static WLS position estimates can fulfill FCC and CGALIES requirements only when there are satellite measurements available. On the other hand, filtered position estimates can fulfill these requirements even using only base station measurements.

Acknowledgements

This study was funded by Nokia Corporation. Niilo Sirola acknowledges the financial support of the Nokia Foundation.

References

Bar-Shalom, Y., Li, R. X. and Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation, Theory Algorithms and Software*, John Wiley & Sons.

Brown, R. G. (1983). *Introduction to Random Signal Analysis and Kalman Filtering*, John Wiley & Sons.

Chaffee, J. W. and Abel, J. S. (1992). The GPS filtering problem, *IEEE Position Location and Navigation Symposium*.

Drane, C., Macnaughtan, M. and Scott, C. (1998). Positioning GSM telephones, *IEEE Communications Magazine* pp. 46–59.

FCC, *Federal Communication Commission website*. www.fcc.gov.

Heinrichs, G., Dovis, F., Gianola, M. and Mulassano, P. (2004). Navigation and communication hybrid positioning with a common receiver architecture, *Proceedings of The European Navigation Conference GNSS 2004*.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*, Vol. 64 of *Mathematics in Science and Engineering*, Academic Press.

Kaplan, E. D., ed. (1996). *Understanding GPS: Principles and Applications*, Artech House, Norwood.

Lefebvre, T., Bruyninckx, H. and De Schutter, J. (2004). Kalman filters for non-linear systems: a comparison of performance, *International Journal of Control* 77(7).

Lerro, D. and Bar-Shalom, Y. (1993). Tracking with debiased consistent converted measurements versus EKF, *IEEE Transactions on Aerospace and Electronic Systems* 29(3).

Ludden, B., Pickford, A., Medland, J., Johnson, H., Axelsson, L. E., Viddal-Ervik, K., Dorgelo, B., Boroski, E. and Malenstein, J. (2002). *Report on Implementation Issues Related to Access to Location Information by Emergency Services (E112) in the European Union (Final Report)*, C.G.A.L.I.E.S.

Ma, C. (2003). Integration of GPS and cellular networks to improve wireless location performance, *Proceedings of ION GPS/GNSS 2003* pp. 1585–1596.

Maybeck, P. S. (1979). *Stochastic Models, Estimation, and Control*, Vol. 141 of *Mathematics in Science and Engineering*, Academic Press.

Maybeck, P. S. (1982). *Stochastic Models, Estimation, and Control*, Vol. 141-2 of *Mathematics in Science and Engineering*, Academic Press.

Ristic, B., Arulampalam, S. and Gordon, N. (2004). *Beyond the Kalman Filter, Particle Filters for Tracking Applications*, Artech House, Boston, London.

Schlosser, M. S. and Kroschel, K. (2004). Limits in tracking with extended Kalman filters, *IEEE Transactions on Aerospace and Electronic Systems* 40(4).

Sirola, N., Piché, R. and Syrjärinne, J. (2003). Closed-form solutions for hybrid cellular/GPS positioning, *Proceedings of ION GPS/GNSS 2003* pp. 1613–1619.

Spirito, M. A., Pöykkö, S. and Knuuttila, O. (2001). Experimental performance of methods to estimate the location of legacy handsets in GSM, *IEEE Vehicular Technology Conference* pp. 2716–2720.

Syrjärinne, J. (2001). Studies of modern techniques for personal positioning, PhD thesis, Tampere University of Technology, Tampere. Publications 319.

Vossiek, M., Wiebking, L., Gulden, P., Wieghardt, J., Hoffmann, C. and Heide, P. (2003). Wireless local positioning, *IEEE microwave magazine* pp. 77–86.

Biography of the Authors

Simo Ali-Löytty is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2004 from the same university. His research interest is filters in personal positioning.

Niilo Sirola is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2003 from the same university. He began his research on personal positioning algorithms in 2000.

Prof. Robert Piché is professor in the Institute of Mathematics at Tampere University of Technology.

Simo Ali-Löytty

Niilo Sirola

Prof. Robert Piché

PUBLICATION 2

Simo Ali-Löytty and Niilo Sirola: A modified Kalman filter for hybrid positioning. In *Proceedings of the 19th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2006)*, Fort Worth TX, September 2006, pages 1679–1686.

A Modified Kalman Filter for Hybrid Positioning

Simo Ali-Löytty and Niilo Sirola, *Institute of Mathematics, Tampere University of Technology, Finland*

BIOGRAPHY

Simo Ali-Löytty is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2004 from the same university. His research interest is filters in personal positioning.

Niilo Sirola received his M.Sc. degree from Tampere University of Technology in 2003. He has been studying positioning algorithms since 2000, and is currently working on his Ph.D. thesis on novel mathematical methods for personal positioning and navigation.

ABSTRACT

This paper presents a new hybrid positioning algorithm whereby restrictive information such as base station sector and maximum range can be used in a Kalman-type filter. This new algorithm is fast to compute and gives almost the same accuracy as the particle filter with millions of particles. Simulations show that in some cases restrictive information such as mobile phone network sector and maximum range information dramatically improve filter accuracy. We also present the mathematical fundamentals of the algorithm.

1 INTRODUCTION

Hybrid positioning means that measurements used in positioning come from many different sources e.g. Global Navigation Satellite System (GNSS), Inertial Measurement Unit (IMU), or local wireless networks such as a cellular network, WLAN, or Bluetooth. Range, pseudorange, deltarange, altitude, and compass measurements are typical measurements in hybrid positioning. Restrictive information can be used in positioning. By restrictive information, we mean the knowledge that the state e.g. position, is inside some area, which is the most often a polyhedron. In the simplest case, the area is a half-space. Base station sector and maximum range information (Cell ID) are examples of restrictive information [17].

Filters are used to compute an estimate of the state using current and past measurement data. Filters also give an approximation of the error covariance matrix. Kalman-type filters approximate the probability distribution of the state as a Gaussian or as a mixture of Gaussians. Extended Kalman Filter (EKF) [8, 10], Second Order Extended Kalman Filter (EKF2) [2], Unscented Kalman Filter (UKF) [9] and Gaussian Mixture Filter (GMF) [16, 7] are examples of Kalman-type filters.

Maybe the most popular example is EKF, which is very commonly used in satellite-based positioning and has also been applied to hybrid positioning. Unfortunately, EKF has a serious consistency problem in highly nonlinear situations [1], i.e. the error covariance matrix is sometimes grossly underestimated. Contrary to satellite based positioning, highly nonlinear situations are common in hybrid positioning. Because of this, many researchers have investigated the use of general nonlinear Bayesian filter, which is usually implemented as a particle filter (SMC, Sequential Monte Carlo) [6, 13] or a point mass based filter [4, 14]. These filters usually work correctly and give good positioning accuracy but require much computation.

An outline of the paper is as follows. After problem formulation in Section 2, the basic idea of the new algorithm is presented in Section 3. Section 4 provides the mathematical fundamentals of the algorithm. In Section 5, we concentrate on hybrid positioning application and illustrate how the new algorithm works in practice with sector and maximum range information. Finally, simulation results are given in Section 6.

We show in Section 6 that in some cases the new hybrid positioning algorithm gives almost the same accuracy as a particle filter. In addition, the new algorithm avoids most of the consistency problems that plague the traditional EKF. The biggest advantage of the new algorithm is that it can use restrictive information and still needs notably less computation time than a particle filter. Because of this, it is conceivable that the algorithm can be implemented in a mobile device such as a mobile phone.

2 BAYESIAN FILTERING

We consider the discrete-time non-linear non-Gaussian system

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_{k-1}, \tag{1}$$

$$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{v}_k, \tag{2}$$

where the vectors $\mathbf{x}_k \in \mathbb{R}^{n_\mathbf{x}}$ and $\mathbf{y}_k \in \mathbb{R}^{n_{\mathbf{y}_k}}$ represent the state of the system and the measurement at time t_k, $k \in \mathbb{N}$, respectively. We assume that errors $\mathbf{w}_k$ and $\mathbf{v}_k$ are white, mutually independent and independent of the initial state $\mathbf{x}_0$. The aim of the filtering is to find conditional probability density function (posterior)

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}), \tag{3}$$

where $\mathbf{y}_{1:k} \triangleq \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$. Posterior can be determined recursively according to the following relations.

Prediction:

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})d\mathbf{x}_{k-1}; \tag{4}$$

Update:

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{\int p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})d\mathbf{x}_k}, \tag{5}$$

where the transition pdf $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ can be derived from (1) and the likelihood $p(\mathbf{y}_k|\mathbf{x}_k)$ can be derived from (2). The initial condition for the recursion is given by the pdf of the initial state $p(\mathbf{x}_0|\mathbf{y}_{1:0}) = p(\mathbf{x}_0)$. Knowledge of the posterior distribution enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error (MMSE) estimate is the conditional mean of $\mathbf{x}_k$ [2, 13]. In general and in our case the conditional probability density function cannot be determined analytically. Because of this, there are many approximative solutions of conditional mean. Some popular Kalman-type approximative solutions of conditional mean include:

EKF (Extended Kalman Filter) Kalman filtering applied to a linearization of system (1), (2). EKF is very commonly used in satellite-based positioning and it has also been applied in hybrid positioning [10]. The EKF algorithm is described for example in [2, 8].

EKF2 (Second Order Extended Kalman Filter) an elaboration of EKF that models nonlinearity better. Modified Gaussian Second Order Filter [8, 12] is the same as EKF2. The EKF2 algorithm is given for example in [2].

PKF (Position Kalman Filter) works by filtering a sequence of static position and velocity solutions [1]. The idea is that we first solve position and velocity and then filter them; this idea is called two-stage estimator [5]. When we have enough measurements, we can find position and velocity for example by Weighted Least Squares (WLS) or closed-form methods [15]. WLS also approximates the errors of the solution. At time instants when there are not enough measurements to fix position and velocity, PKF does not use any measurements.

2.1 Restrictive information

As we have seen, in the underdetermined case sometimes we have situation that posterior is multimodal and that can inflict inconsistency [1]. However, sometimes we have some extra information which may remove the multimodality. We call this extra information restrictive information. As mentioned previously, it is very common that we have restrictive information in hybrid positioning. For example, a mobile phone network, WLAN and Bluetooth base stations have specific range and if we "hear" the base station then we are inside the base station range. Of course restrictive information can also restrict other state variables than only position. More formally, restrictive information is measurement with measurement function (2)

$$h(\mathbf{x}) = \begin{cases} 1, \text{ if } \mathbf{x} \in A \\ 0, \text{ if } \mathbf{x} \notin A \end{cases}, \tag{6}$$

where $A \subset \mathbb{R}^{n_\mathbf{x}}$. In this paper, we consider only a case where restrictive information does not have error, so then error term $\mathbf{v} = 0$ is constant random variable. The likelihood function of restriction information is characteristic function (5), (6)

$$p(1|\mathbf{x}) = \chi_A(\mathbf{x}) = \begin{cases} 1, \text{ if } \mathbf{x} \in A \\ 0, \text{ if } \mathbf{x} \notin A \end{cases}, \tag{7}$$

and respectively $p(0|\mathbf{x}) = \chi_{\overline{A}}(\mathbf{x})$, where $\overline{A}$ is the complement of A. Unfortunately, it is not straightforward to use restrictive information with Kalman-type filters because measurement is a dichotomy variable and the measurement function is not linear.

3 NEW ALGORITHM IN A NUTSHELL

The main idea of the new algorithm is that we use some Kalman-type filter and extend it to use restrictive information. Here we present the new algorithm in the case where posterior is Gaussian, for example EKF, EKF2, PKF.

1. Compute the posterior without restrictive information using Kalman-type filter.

2. If there is new restrictive information, model restrictive information as a likelihood function that is one inside a certain polyhedron and zero outside.
3. Compute new mean and covariance estimates.
4. Approximate the posterior distribution with a Gaussian, with the new mean and covariance.
5. Repeat this every timestep.

The heart of this algorithm is stage 3, where we update posterior mean and covariance. Notice that this algorithm does not require any other measurements, so the algorithm works also if there is only restriction information.

We can apply this algorithm also when posterior is Gaussian mixture. In this case, we apply this algorithm to every mixture component, and finally compute new mixture weights.

4 DERIVATION OF THE ALGORITHM

In this section, we concentrate on the mathematical fundamentals of the new algorithm. First of all, the algorithm changes only the update part (5) so prediction part (4) remains the same as in the ordinary filter.

1. Compute the posterior without restrictive information using Kalman-type filter.

Because of the independence, we can write likelihood function as the product

$$p(\mathbf{y}_k|\mathbf{x}_k) = \chi_A(\mathbf{x}_k)p(\mathbf{y}'_k|\mathbf{x}_k), \tag{8}$$

where A is the intersection of all new restrictive information, this means that state is inside A at time instant t_k and $\mathbf{y}'_k$ are other measurements than restrictive information at time instant t_k. Using Eqs. (5) and (8) we find that

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \propto \chi_A(\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}'_{1:k}), \tag{9}$$

where

$$p(\mathbf{x}_k|\mathbf{y}'_{1:k}) = \frac{p(\mathbf{y}'_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{\int p(\mathbf{y}'_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})d\mathbf{x}_k} \tag{10}$$

is posterior without current restrictive information. This can be computed using the ordinary Kalman-type filter. We use notation that μ_{old} and Σ_{old} represent the approximations of the mean and the covariance matrix of this distribution, respectively. We also approximate $p(\mathbf{x}_k|\mathbf{y}'_{1:k})$ with a Gaussian.

2. If there is new restrictive information, model restrictive information as a likelihood function that is one inside a certain polyhedron and zero outside.

We use the same restrictive information only once within a certain time period, because otherwise our approximation error can accumulate and cause unwanted phenomena. Let the area A be inscribed in the polyhedron (see Fig 1)

$$B = \left\{ x| \begin{bmatrix} b_1^T \\ \vdots \\ b_n^T \end{bmatrix} x \leq \beta \right\} = \{x|\mathrm{B}x \leq \beta\}, \tag{11}$$

where matrix B is organized so that the first inequality reduces probability the most, that is

$$\begin{gathered} \int_{b_1^T x_k \leq \beta_1} p(x_k|\mathbf{y}'_{1:k})dx_k \leq \cdots \leq \int_{b_n^T x_k \leq \beta_n} p(x_k|\mathbf{y}'_{1:k})dx_k \\ \Longleftrightarrow \\ \frac{\beta_1 - b_1^T\mu_{\text{old}}}{\sqrt{b_1^T\Sigma_{\text{old}}b_1}} \leq \cdots \leq \frac{\beta_n - b_n^T\mu_{\text{old}}}{\sqrt{b_n^T\Sigma_{\text{old}}b_n}}. \end{gathered} \tag{12}$$

We use Gram-Schmidt orthonormalization on $\{b_1, \ldots, b_n\}$ with inner product $< x, y >= x^T\Sigma_{\text{old}}y$, and get matrix A whose row space is same than the row space of matrix B and $\mathrm{A}\Sigma_{\text{old}}\mathrm{A}^T = \mathrm{I}$. Now we approximate the likelihood of restrictive information with one inside the polyhedron (see Fig 1)

$$A' = \{x \mid |\mathrm{A}x - \mathrm{A}x_{\text{mid}}| \leq \alpha\} \tag{13}$$

and zero elsewhere. So we replace χ_A in Eq. (9) with $\chi_{A'}$. Vectors $\mathrm{A}x_{\text{mid}}$ and α we select so that B is subset of A' and probability that state is inside A' is as small as possible. Then also A is subset of A'. If

$$\mathrm{X} = \begin{bmatrix} x_1, \ldots, x_N \end{bmatrix} \tag{14}$$

are vertices of polyhedron B, then

$$\alpha = \frac{up - low}{2} \tag{15}$$

and

$$\mathrm{A}x_{\text{mid}} = \mathrm{A}\mu_{\text{old}} - \frac{up + low}{2}, \tag{16}$$

where

$$up = \mathrm{A}\mu_{\text{old}} - \min_{i\in\{1,\ldots,N\}} (\mathrm{AX}e_i) \tag{17}$$

and

$$low = \mathrm{A}\mu_{\text{old}} - \max_{i\in\{1,\ldots,N\}} (\mathrm{AX}e_i)\,. \tag{18}$$

Operators min and max are taken separately with every components. We use this same notation through the paper.

3. Compute new mean and covariance estimates.

In this stage, we compute the approximation of the mean μ_{new} and the covariance matrix Σ_{new} of the modified posterior $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ (9), where A is replaced with A'. This approximation is accurate if the old posterior $p(\mathbf{x}_k|\mathbf{y}'_{1:k})$

is Gaussian and $A = A'$. First we compute probability that state is inside polyhedron A',

$$\begin{aligned} p_{\text{new}} &= \int \chi_{A'}(x_k) p(x_k|\mathbf{y}'_{1:k}) dx_k \\ &= \int_{|z|\leq\alpha} p_{\mathbf{z}}(z) dz \\ &= \prod_{i=1}^{n} \left(\Phi\left(up_i\right) - \Phi\left(low_i\right)\right), \end{aligned} \tag{19}$$

where $\Phi(x)$ is cumulative density function (cdf) of x, when $x \sim \mathrm{N}(0,1)$. The idea of the solution is that with the change variables,

$$\begin{aligned} \mathbf{z} &= \mathrm{A}\mathbf{x}_k - \mathrm{A}x_{\text{mid}}, \\ \mathrm{E}(\mathbf{z}) &= \mu_{\mathbf{z}} = \mathrm{A}\mu_{\text{old}} - \mathrm{A}x_{\text{mid}}, \\ \mathrm{V}(\mathbf{z}) &= \Sigma_{\mathbf{z}} = \mathrm{A}\Sigma_{\text{old}}\mathrm{A}^T = \mathrm{I}, \end{aligned} \tag{20}$$

we can compute the integral iteratively. We use this same idea also in the following integrals, when computing mean and covariance matrix. If posterior is Gaussian mixture then the new weight is proportional to old weight times the probability p_{new}.

The approximation of the mean of the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is

$$\begin{aligned} \mu_{\text{new}} &= \int x_k \frac{\chi_{A'}(x_k) p(x_k|\mathbf{y}'_{1:k})}{p_{\text{new}}} dx_k \\ &= \frac{\mathrm{A}^{-1}}{p_{\text{new}}} \int_{|z|\leq\alpha} z p_{\mathbf{z}}(z) dz + x_{\text{mid}} \\ &= \mathrm{A}^{-1} \int z p_{\mathbf{z}'}(z) dz + x_{\text{mid}} \\ &\overset{(35)}{=} \mu_{\text{old}} + \Sigma_{\text{old}}\mathrm{A}^T\epsilon \end{aligned} \tag{21}$$

where

$$p_{\mathbf{z}'}(z) = \frac{\chi_{|z|\leq\alpha}(z) p_{\mathbf{z}}(z)}{p_{\text{new}}} \tag{22}$$

and

$$\epsilon = \sum_{i=1}^{n} e_i \frac{\exp\left(-\frac{{up_i}^2}{2}\right) - \exp\left(-\frac{{low_i}^2}{2}\right)}{\sqrt{2\pi}\left(\Phi\left(up_i\right) - \Phi\left(low_i\right)\right)}. \tag{23}$$

Details of the computation are given in Appendix A, Eq. (35). Here we assumed that $\mathrm{A} \in \mathbb{R}^{n_\mathrm{A} \times m_\mathrm{A}}$ is square matrix ($n_\mathrm{A} = m_\mathrm{A}$) and non-singular, so polyhedron A' is a bounded "box".

Now we show that Eq. (21) holds also when we have an unbounded area. If $n_\mathrm{A} < m_\mathrm{A}$ we can add more rows to matrix A and elements to vector α, this means that we add more restrictive information. When the added elements α_i, $i \in \{n_\mathrm{A}, \ldots, m_\mathrm{A}\}$ tend to infinity, we get same equation as previously, because ϵ_i, $i \in \{n_\mathrm{A}, \ldots, m_\mathrm{A}\}$ go to zero.

The approximation of the covariance matrix of the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is

$$\begin{aligned} \Sigma_{\text{new}} &= \int (x_k - \mu_{\text{new}})(x_k - \mu_{\text{new}})^T \frac{\chi_{A'}(x_k) p(x_k|\mathbf{y}'_{1:k})}{p_{\text{new}}} dx_k \\ &= \mathrm{A}^{-1} \int (z - \mu_{\mathbf{z}'})(z - \mu_{\mathbf{z}'})^T p_{\mathbf{z}'}(z) dz \mathrm{A}^{-T} \\ &\overset{(37)}{=} \mathrm{A}^{-1}(\mathrm{I} - \Lambda)\mathrm{A}^{-T} \\ &= \Sigma_{\text{old}} - \Sigma_{\text{old}}\mathrm{A}^T\Lambda\mathrm{A}\Sigma_{\text{old}}, \end{aligned} \tag{24}$$

where

$$\mu_{\mathbf{z}'} = \mu_{\mathbf{z}} + \epsilon = \mathrm{A}\mu_{\text{new}} - \mathrm{A}x_{\text{mid}}, \tag{25}$$

$$\Lambda = \mathrm{diag}(\delta) + \mathrm{diag}(\epsilon)\,\mathrm{diag}(\mathrm{A}\mu_{\text{new}} - \mathrm{A}x_{\text{mid}}), \tag{26}$$

where

$$\delta = \sum_{i=1}^{n} \alpha_i e_i \frac{\exp\left(-\frac{{up_i}^2}{2}\right) + \exp\left(-\frac{{low_i}^2}{2}\right)}{\sqrt{2\pi}\left(\Phi\left(up_i\right) - \Phi\left(low_i\right)\right)}. \tag{27}$$

Details of the computation are given in Appendix A, Eq. (37). This equation also works when matrix A is not a square matrix. Proof goes similarily than previously.

4. Approximate the posterior distribution with a Gaussian, with the new mean and covariance.

This stage usually produces the most approximation errors in this algorithm. This is the reason why we use each restrictive information only once.

5. Repeat this every timestep.

5 EXAMPLE OF THE NEW ALGORITHM

In this section, we illustrate how the new algorithm works. In this example, Fig 1, restrictive information consists of two base station Cell IDs, which means that we know the base stations sector and maximum range information. In the figure, dark area (area A in Eq. (9)) is the intersection of these information and so we know that user is inside this area. Black dashed polyhedron represents an polyhedron approximation of the true restrictive area (11). Black solid polyhedron represents the certain polyhedron approximation of the polyhedron approximation of the true restrictive area (13). Red man stands on prior mean and a red dashed ellipse represent prior covariance. The covariances (Σ) are visualized with ellipses that satisfy the equation

$$(x - \mu)^T \Sigma^{-1} (x - \mu) = 2.2173. \tag{28}$$

The constant was chosen so that if the distribution is Gaussian then there is 67% probability mass inside the ellipse. Respectively, blue man stands on posterior mean and a blue solid ellipse represent posterior covariance.

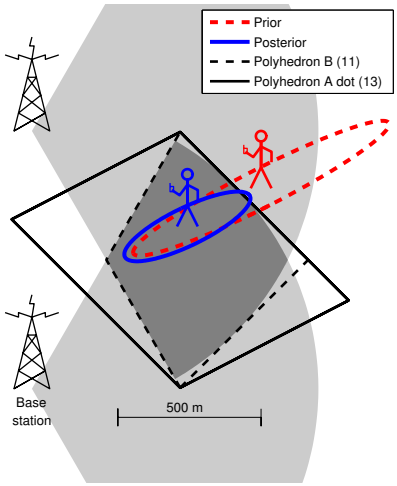

Figure 1: Red means prior distribution. Dashed black polyhedron represents the first approximation of restrictive area (11). Solid black polyhedron represents the certain approximation of the true restrictive area (13). Blue represents the mean and covariance approximations after applying the restrictive information.

It is quite evident that using restrictive information improves estimates notably only when there is much probability mass outside the restrictive area. Usually, this happens two different cases. The first case is that we have only a few measurement, which causes large posterior covariance without restrictive information, but we have quite good restrictive information such as sector information. Other possibly is that Kalman-type filters works totally wrong, this means that filter is inconsistent so that predicted errors are smaller than actual errors [1]. Both of these cases are possible in hybrid positioning, so it is reasonable to use restrictive information in hybrid positioning. In Section 6, we see more specifically how restrictive information improves Kalman-type filter performance in hybrid positioning.

6 SIMULATIONS

In the simulations, we use the position-velocity model, so the state $\mathbf{x} = \left[\mathbf{r}_\mathrm{u}^T,\ \mathbf{v}_\mathrm{u}^T\right]^T$ consists of user position vector $\mathbf{r}_\mathrm{u}$ and user velocity vector $\mathbf{v}_\mathrm{u}$, which are in East-North-Up (ENU) coordinate system. In this model the user velocity is a random walk process [3]. Now the state-dynamic Eq. (1) is

$$\mathbf{x}_k = \Phi_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \tag{29}$$

where

$$\Phi_{k-1} = \begin{bmatrix} \mathrm{I} & \Delta t_k \mathrm{I} \\ 0 & \mathrm{I} \end{bmatrix}, \tag{30}$$

$\Delta t_k = t_k - t_{k-1}$, and $\mathbf{w}_{k-1}$ is white, zero mean and Gaussian noise, with covariance matrix

$$\mathrm{Q}_{k-1} = \begin{bmatrix} \frac{\Delta t_k^3 \sigma_\mathrm{p}^2}{3}\mathrm{I} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^3 \sigma_\mathrm{a}^2}{3} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} \\ \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 & \frac{\Delta t_k \sigma_\mathrm{p}^2}{1}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} & 0 & \frac{\Delta t_k \sigma_\mathrm{a}^2}{1} \end{bmatrix}, \tag{31}$$

where $\sigma_\mathrm{p}^2 = 2\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors on the East-North plane and $\sigma_\mathrm{a}^2 = 0.1\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors in the vertical direction. [11, 1]

In our simulations, we use base station range measurement, altitude measurement, satellite range measurement and satellite deltarange measurement.

$$\begin{aligned} y_\mathrm{b} &= \|\mathbf{r}_\mathrm{b} - \mathbf{r}_\mathrm{u}\| + \epsilon_\mathrm{b}, \\ y_\mathrm{a} &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{r}_\mathrm{u} + \epsilon_\mathrm{a}, \\ y_\mathrm{s} &= \|\mathbf{r}_\mathrm{s} - \mathbf{r}_\mathrm{u}\| + b + \epsilon_\mathrm{s}, \\ \dot{y}_\mathrm{s} &= \frac{(\mathbf{r}_\mathrm{s} - \mathbf{r}_\mathrm{u})^T}{\|\mathbf{r}_\mathrm{s} - \mathbf{r}_\mathrm{u}\|}(\mathbf{v}_\mathrm{s} - \mathbf{v}_\mathrm{u}) + \dot{b} + \dot{\epsilon}_\mathrm{s}, \end{aligned} \tag{32}$$

where $\mathbf{r}_\mathrm{b}$ is base station position vector, $\mathbf{r}_\mathrm{s}$ is satellite position vector, b is clock bias, $\mathbf{v}_\mathrm{s}$ is satellite velocity vector, $\dot{b}$ is clock drift and ϵ:s are error terms. We use satellite measurements only when there is more than one satellite measurement available, so that bias can be eliminated. These are the same measurements equations as in the paper [1]. Restrictive information that we use are base station sector information and maximum range information.

From Fig. 2, we get an idea of how the new algorithm works. These simulations use only a few (one or two at the same time instant) base station range measurements with variance $(100\ \mathrm{m})^2$ and altitude measurement with variance $(300\ \mathrm{m})^2$. In this case, restrictive information keeps the filter always consistent (not inconsistent). In this paper, we use *the general inconsistency test*, with risk level 5% [1]. Restrictive information also decreases the mean error from 442 meters to 93 meters.

In Table 1, we have listed a summary of a hundred 300 second simulations. The simulations use only a few (one or two at the same time instant) base station range measurements with variance $(80\ \mathrm{m})^2$ and altitude measurements with variance $(300\ \mathrm{m})^2$. Summary consist of following columns: *Time* is relative computation time using Matlab and our implementation, so that computation time of EKF is one. This is not entire truth, but gives an idea of complexity of each algorithm. *Err.* μ is 2D position error mean. *Err. 95%* tells that 95 % errors are less than this limit. *Err. ref.* is 2D error to reference posterior mean, which is computed using particle filter with $1 \cdot 10^6$ particles. *Err. <50* is how many percentage of times 2D position error is less than 50 meter. *Inc. %* is how many percentage of time filter is inconsistent with respect to the general inconsistency test, with risk level 5% [1]. Solvers are organized so that mean positioning errors are in descending order. We also test

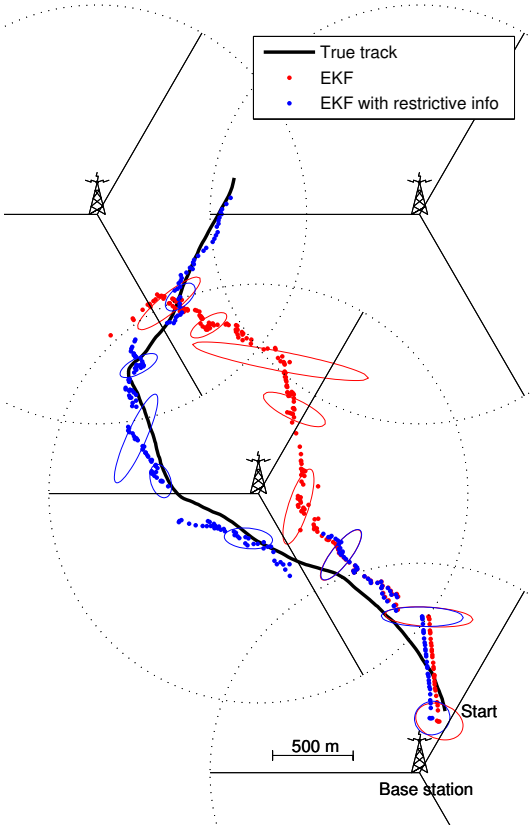

Figure 2: This figure illustrate how the new algorithm, EKF with restrictive information (EKFBOX), improve EKF in mobile phone positioning. In this case EKF and EKFBOX mean of the position errors are 442 m and 93 m, respectively.

positioning using only restrictive information, we call this the BOXsolver. We also use "BOX" suffix when we use the new algorithm for incorporating restrictive information.

Table 1: Summary of 100 different simulations with very poor geometry. Simulations only use a few base station range measurements and very inaccurate altitude measurements; an example of this geometry is in Fig. 2.

Solver	Time $\propto$	Err. μ	Err. 95%	Err. ref	Err. <50	Inc. %
EKF	1	393	1356	358	11	28.1
EKF2	1.5	368	1028	314	8	8.4
BOX	1	314	669	253	3	0.1
EKFBOX	2	192	625	126	19	7.7
EKF2BOX	3	186	582	114	17	5.1
SMC	74	145	436	17	25	0.2
Ref	∞	145	423	0	26	0.1

From Table 1, we see that restrictive information improves greatly filters performance. Every statistic, except CPU time, of the Kalman-type filters is improved by the new algorithm. Also, difference between the reference solution (particle filter), which uses all restrictive information, and EKF(2)BOX-solver is quite small. However EKF(2)BOX-solver's computation time is much less than the particle filter (SMC). SMC and reference solvers are the same solver, but SMC solver uses 10^4 particles, when reference solves uses 10^6 particles. Note that the geometry is quite bad and traditional solvers like EKF(2) give even worse results than BOX-solver, which uses only restrictive information and not range measurements or altitude measurements at all.

In Table 2 we have listed a summary of a hundred simulations, but now simulations use normal suburban geometry. This means that simulations use few base station range measurements, very inaccurate altitude measurements, some (not more than five) satellite pseurange and delta pseudorange measurements, with variance $(20\,\mathrm{m})^2$ and $(2\frac{\mathrm{m}}{\mathrm{s}})^2$ respectively. We see that errors are smaller, of course, than in Table 1. If we concentrate on the error ratio of different filters, we see that results are quite same than previosly. So the new algortihm improves Kalman-type filters much and these EKF(2)BOX-solvers give almost the same accuracy as particle filters.

Table 2: Summary of 100 different simulations with normal suburban geometry. Simulations use few base station range measurements, very inaccurate altitude measurements, some satellite pseurange and delta pseudorange measurements.

Solver	Time $\propto$	Err. μ	Err. 95%	Err. ref	Err. <50	Inc. %
BOX	1	285	644	279	5	0.7
EKF	1	76	255	46	60	2.1
EKF2	1.5	88	330	56	59	2.0
EKFBOX	2	60	195	23	63	0.5
EKF2BOX	2	60	196	21	63	0.4
SMC	50	53	166	4	66	0.1
Ref	∞	53	164	0	66	0.0

7 CONCLUSIONS

We presented a new hybrid positioning algorithm, which makes possible to use restrictive information with Kalman-type filters. The new algorithm needs clearly less computation time than the general nonlinear Bayesian filters such as the particle filter with 10^4 particles. Simulations show that the new algorithm is very well suited to hybrid positioning application, where we have sector and maximum range restrictive information. In this application, the new algorithm improves EKF and EKF2 much and gives almost same accuracy as reference particle filters.

A AUXILIARY INTEGRALS

In this section, we compute some auxiliary integrals. Here $\mathbf{z}$ is $n_{\mathbf{z}}$-dimensional Gaussian random variable with mean $\mu_{\mathbf{z}}$ and covariance matrix is identity matrix. So the density function of $\mathbf{z}$ is

$$p_{\mathbf{z}}(z) = \prod_{i=1}^{n_{\mathbf{z}}} p_{\mathbf{z}_i}(z_i) = \prod_{i=1}^{n_{\mathbf{z}}} \frac{\exp\left(-\frac{(z_i-\mu_{\mathbf{z}_i})^2}{2}\right)}{\sqrt{2\pi}} \tag{33}$$

and so

$$\begin{aligned} p_{\mathbf{z}'}(z) &= \frac{\chi_{|z|\leq\alpha}(z)p_{\mathbf{z}}(z)}{p_{\text{new}}} \\ &= \prod_{i=1}^{n_{\mathbf{z}}} \frac{\chi_{|z_i|\leq\alpha_i}(z_i)\exp\left(-\frac{(z_i-\mu_{\mathbf{z}_i})^2}{2}\right)}{\left(\Phi\left(up_i\right)-\Phi\left(low_i\right)\right)\sqrt{2\pi}} \\ &= \prod_{i=1}^{n_{\mathbf{z}}} p_{\mathbf{z}'_i}(z_i), \end{aligned} \tag{34}$$

which is also a density function.

Auxiliary integral of derivation of Eq. (21)

$$\begin{aligned} \mu_{\mathbf{z}'} &= \int z p_{\mathbf{z}'}(z)dz \\ &= \int (z-\mu_{\mathbf{z}}) \prod_{i=1}^{n_{\mathbf{z}}} p_{\mathbf{z}'_i}(z_i)dz + \mu_{\mathbf{z}} \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j \int (z_j-\mu_{\mathbf{z}_j}) \prod_{i=1}^{n_{\mathbf{z}}} p_{\mathbf{z}'_i}(z_i)dz + \mu_{\mathbf{z}} \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j \frac{\int_{-\alpha_j}^{\alpha_j}(z_j-\mu_{\mathbf{z}_j})p_{\mathbf{z}_j}(z_j)dz_j}{\Phi\left(up_j\right)-\Phi\left(low_j\right)} + \mu_{\mathbf{z}} \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j \frac{\exp\left(-\frac{up_j^2}{2}\right)-\exp\left(-\frac{low_j^2}{2}\right)}{\sqrt{2\pi}\left(\Phi\left(up_j\right)-\Phi\left(low_j\right)\right)} + \mu_{\mathbf{z}} \\ &= \epsilon + \mu_{\mathbf{z}}, \end{aligned} \tag{35}$$

where p_{new} is defined in Eq. (19) and from Eqs. (15), (16) and (20) we get that

$$up = \mu_{\mathbf{z}} + \alpha \quad \text{and} \quad low = \mu_{\mathbf{z}} - \alpha. \tag{36}$$

Auxiliary integral of derivation of Eq. (24)

$$\begin{aligned} \Sigma_{\mathbf{z}'} &= \int (z-\mu_{\mathbf{z}'})(z-\mu_{\mathbf{z}'})^T p_{\mathbf{z}'}(z)dz \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j e_j^T \int (z_j-\mu_{\mathbf{z}'_j})^2 p_{\mathbf{z}'}(z)dz \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j e_j^T \int (z_j-\mu_{\mathbf{z}_j}-\epsilon_j)^2 p_{\mathbf{z}'}(z)dz \\ &= \sum_{j=1}^{n_{\mathbf{z}}} e_j e_j^T \int (z_j-\mu_{\mathbf{z}_j})^2 p_{\mathbf{z}'_j}(z)dz_j - \text{diag}(\epsilon)^2 \\ &= \text{I} - \text{diag}(\delta) - \text{diag}(\epsilon)\,\text{diag}(\text{A}\mu_{\text{new}} - \text{A}x_{\text{mid}}) \end{aligned} \tag{37}$$

ACKNOWLEDGMENTS

This study was partly funded by Nokia Corporation. Simo Ali-Löytty acknowledges the financial support of the Nokia Foundation and the Tampere Graduate School in Information Science and Engineering. The particle filter we used was implemented by Duane Petrovich.

REFERENCES

[1] S. Ali-Löytty, N. Sirola, and R. Piché. Consistency of three Kalman filter extensions in hybrid navigation. In *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, July 2005.

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

[3] R. G. Brown. *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, 1983.

[4] R. S. Bucy and K. D. Senne. Digital synthesis of nonlinear filters. *Automatica*, 7(3):287–298, May 1971.

[5] J. W. Chaffee and J. S. Abel. The GPS filtering problem. *IEEE Position Location and Navigation Symposium*, 1992.

[6] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, 2001.

[7] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5):910–927, May 2000.

[8] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.

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

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

[11] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141 of *Mathematics in Science and Engineering*. Academic Press, 1979.

[12] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141-2 of *Mathematics in Science and Engineering*. Academic Press, 1982.

[13] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Artech House, Boston, London, 2004.

[14] N. Sirola and S. Ali-Löytty. Local positioning with parallelepiped moving grid. In *Proceedings of 3rd Workshop on Positioning, Navigation and Communication 2006 (WPNC'06)*, pages 179–188, Hannover, March 16th 2006.

[15] N. Sirola, R. Piché, and J. Syrjärinne. Closed-form solutions for hybrid cellular/GPS positioning. *Proceedings of ION GPS/GNSS 2003*, pages 1613–1619, 2003.

[16] H. W. Sorenson and D. L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7(4):465–479, July 1971.

[17] M. A. Spirito, S. Pöykkö, and O. Knuuttila. Experimental performance of methods to estimate the location of legacy handsets in GSM. *IEEE Vehicular Technology Conference*, pages 2716–2720, 2001.

PUBLICATION 3

Simo Ali-Löytty and Niilo Sirola: Gaussian mixture filter in hybrid navigation. In *Proceedings of the European Navigation Conference 2007 (ENC 2007)*, Geneva, May 2007, pages 831–837.

Gaussian Mixture Filter in Hybrid Navigation

Simo Ali-Löytty and Niilo Sirola,
Institute of Mathematics, Tampere University of Technology, Finland
simo.ali-loytty@tut.fi

BIOGRAPHY

Simo Ali-Löytty is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2004 from the same university. His research interest is filters in personal positioning.

Niilo Sirola is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2003 from the same university. He began his research on personal positioning algorithms in 2000.

ABSTRACT

This paper studies the Gaussian Mixture Filter (GMF) in hybrid navigation and is divided into three parts focusing on different problems. The first part deals with approximating the likelihood function as a Gaussian mixture. The second part gives a new way to robustify GMF by adding more components, and the third part concentrates on different ways to reduce the number of mixture components so that the algorithm is fast without losing significant information. Finally, we compare GMF with other filters using simulations.

1 INTRODUCTION

Hybrid navigation means navigation using measurements from different sources e.g. Global Navigation Satellite System (e.g. GPS), Inertial Measurement Unit, or local wireless networks such as a cellular network, WLAN, or Bluetooth. Range, pseudorange, deltarange, altitude, restrictive and compass measurements are examples of typical measurements in hybrid navigation.

The most popular example of navigation filters is the Extended Kalman Filter (EKF), which linearizes system and measurement models and then applies the traditional Kalman Filter [6, 9]. EKF is commonly used in satellite based positioning and has also been applied to hybrid navigation [11]. Unfortunately, EKF has a serious consistency problem in highly nonlinear situations, which means that EKF does not work correctly [2]. In highly nonlinear situations, we sometimes have multiple static position solutions, which means that likelihood function has multiple peaks with significant weight. In this case, it is more reasonable to approximate the likelihood as a Gaussian mixture and use GMF (section 4) than to approximate with only one Gaussian as the EKF does.

We present a new method to robustify the GMF in section 5, which takes into consideration that the prior distribution might be wrong. This is usually the case when the filter has inconsistency problems. In a nutshell, the idea of the new method is following: if measurements are unlikely with respect to the prior, the new method adds mixture components so that the new mixture consists of the prior or posterior distribution plus extra component(s) that are approximately proportional to the likelihood function. This is better than the traditional robust approach which can discard good measurements because an erroneous prior makes them seem unlikely. The proposed method does not replace old methods but rather extends them.

One major challenge in using GMF efficiently is keeping the number of components as small as possible without losing significant information. We study in section 6 both the traditional component reduction methods, merging and forgetting, and new resampling algorithms similar to the ones used in particle filters.

In the simulations part in section 7, we compare different GMFs with different Kalman type filters such as EKF and the Unscented Kalman Filter (UKF) [10], as well as with a bootstrap particle filter [4]. We compare the mean estimates given by the filters to the true track and to the mean estimate of the nearly optimal particle filter. We also look at the computation times and different inconsistency statistics. It is interesting to notice that none of the filters dominate with respect to all of the criteria, so with suitable weighting of the criteria we could make any of the filters win the comparison. However, generally it is possible to develop GMF which works better than EKF or UKF especially when we consider a robust viewpoint. Naturally GMF needs more computations than EKF or UKF but usually less than the particle filter.

2 BAYESIAN FILTERING

We consider the discrete-time non-linear non-Gaussian system

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \tag{1}$$

$$y_k = h_k(x_k) + v_k, \tag{2}$$

where the vectors $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_{y_k}}$ represent the state of the system and the measurement at time t_k, $k \in \mathbb{N}$, respectively. We assume that errors w_k and v_k are white, mutually independent and independent of the initial state x_0. We denote the density functions of w_k and v_k by p_{w_k} and p_{v_k}, respectively. The aim of filtering is to find the conditional probability density function (posterior)

$$p(x_k|y_{1:k}),$$

where $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$. The posterior can be determined recursively according to the following relations.
Prediction (prior):

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}; \tag{3}$$

Update (posterior):

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k}, \tag{4}$$

where the transition pdf is

$$p(x_k|x_{k-1}) = p_{w_{k-1}}(x_k - f_{k-1}(x_{k-1}))$$

and the likelihood

$$p(y_k|x_k) = p_{v_k}(y_k - h_k(x_k)). \tag{5}$$

The initial condition for the recursion is given by the pdf of the initial state $p(x_0|y_{1:0}) = p(x_0)$. Knowledge of the posterior distribution (4) enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error (MMSE) estimate is the conditional mean of x_k [6, 17]. In general and in our case, the conditional probability density function cannot be determined analytically.

3 GAUSSIAN MIXTURE FILTER

The idea of GMF [3] is that both prior density (3) and posterior density (4) are Gaussian mixtures

$$p(x) = \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x), \tag{6}$$

where $\mathrm{N}_{\Sigma_i}^{\mu_i}(x)$ is the normal density function with mean μ_i and covariance matrix Σ_i. Weights are non-negative and sum to one. The mean of a Gaussian mixture (6) is

$$\mu = \sum_{i=1}^{p} \alpha_i \mu_i$$

and the covariance

$$\Sigma = \sum_{i=1}^{p} \alpha_i (\Sigma_i + (\mu_i - \mu)(\mu_i - \mu)^T).$$

We assume that the prior (3) and likelihood (5) are

$$p(x) = \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x) \quad \text{and}$$
$$p(y|x) = \sum_{j=1}^{m} \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y).$$

Then the posterior is a Gaussian mixture (see appendix A)

$$p(x|y) = \frac{\sum_{j=1}^{m}\sum_{i=1}^{p} \alpha_i\beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j\mu_i}(y) \, \mathrm{N}_{\hat{\mathrm{P}}_{i,j}}^{\hat{x}_{i,j}}(x)}{\sum_{j=1}^{m}\sum_{i=1}^{p} \alpha_i\beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j\mu_i}(y)}, \tag{7}$$

where

$$\begin{aligned}
\mathrm{P}_{i,j} &= \mathrm{H}_j\Sigma_i\mathrm{H}_j^T + \mathrm{R}, \\
\hat{x}_{i,j} &= \mu_i + \mathrm{K}_{i,j}(y - \mathrm{H}_j\mu_i), \\
\hat{\mathrm{P}}_{i,j} &= (\mathrm{I} - \mathrm{K}_{i,j}\mathrm{H}_j)\Sigma_i \text{ and} \\
\mathrm{K}_{i,j} &= \Sigma_i\mathrm{H}_j^T\mathrm{P}_{i,j}^{-1}.
\end{aligned}$$

4 GAUSSIAN MIXTURE APPROXIMATION OF LIKELIHOOD

Henceforth we assume that errors w_k and v_k are zero-mean Gaussian with covariance matrix Q_k and R_k, respectively. Also we assume that the initial state x_0 is a Gaussian mixture and that the state model (1) is linear so

$$f_{k-1}(x_{k-1}) = \Phi_{k-1}x_{k-1}.$$

If the posterior density $p(x_k|y_{1:k})$ is a Gaussian mixture (6) then the next prior density (3) is (see appendix A (11))

$$\begin{aligned}
p(x_{k+1}|y_{1:k}) &= \sum_{i=1}^{p} \alpha_i \int \mathrm{N}_{\mathrm{Q}_k}^{\Phi_k x_k}(x_{k+1}) \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x_k)dx_k \\
&= \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Phi_k\Sigma_i\Phi_k^T+\mathrm{Q}_k}^{\Phi_k\mu_i}(x_{k+1}),
\end{aligned}$$

which is also a Gaussian mixture. So the actual problem is how to approximate/compute the new posterior density (4), that is, how to approximate the product

$$p(x_k|y_{1:k}) \propto p(y_k|x_k)p(x_k|y_{1:k-1}),$$

where (see (5))

$$p(y_k|x_k) = \frac{\exp\left(-\frac{1}{2}\|y_k - h_k(x_k)\|_{\mathrm{R}_k^{-1}}^2\right)}{\sqrt{\det(2\pi\mathrm{R}_k)}}$$

and

$$\|y_k - h_k(x_k)\|_{\mathrm{R}_k^{-1}}^2 = (y_k - h_k(x_k))^T\mathrm{R}_k^{-1}(y_k - h_k(x_k)).$$

There are many possibilities to do this. We can use the same methodology that Kalman filter extensions use to get

GMF$_{\mathbf{EKF}}$ Extended Kalman Filter (EKF) [6, 9], linearize the measurement function around the prior mean $\hat{x}_{k|k-1}$

$$h_k(x_k) \approx h_k(\hat{x}_{k|k-1}) + h_k'(\hat{x}_{k|k-1})(x_k - \hat{x}_{k|k-1}).$$

GMF$_{\text{EKF2}}$ Second Order Extended Kalman Filter [9] is an elaboration of EKF that models nonlinearity better. Modified Gaussian Second Order Filter [14] is the same as EKF2.

GMF$_{\text{UKF}}$ Unscented Kalman Filter [10] is based on the numerical integration method called the unscented transformation.

Often these are quite good approximations, but sometimes especially when likelihood has several peaks in a neighbourhood of the prior mean, these approximations fail. These are actually situations where the above filters and other unimodal Kalman filter extensions can easily have consistency problems. Because of this, we develop a new approximation that works better in this situation. First we compute (see appendix B)

$$z = \operatorname{argmin}_{x_k} \left[(y_k - h_k(x_k))^T \mathrm{R}_k^{-1} (y_k - h_k(x_k)) \right], \quad (8)$$

which is possibly ambiguous, so we denote the minimizers $z_1, \ldots, z_m$. We consider only the situation where there is a finite number of minimum points. Then we approximate

$$p(y_k|x_k) \approx \sum_{j=1}^{m} \frac{\exp\left(-\frac{1}{2}\|\tilde{y}_k - \mathrm{H}_{k_j} x_k\|^2_{\mathrm{R}_k^{-1}}\right)}{\sqrt{\det(2\pi \mathrm{R}_k)}}, \quad (9)$$

where $\mathrm{H}_{k_j} = h_k'(z_j)$ and $\tilde{y}_k = y_k - h_k(z_j) + \mathrm{H}_{k_j} z_j$. This approximation ensures that the posterior is also a Gaussian mixture, and takes into consideration the possibility that the likelihood has multiple peaks. An other benefit is that this approximation is independent of prior mean, so it is quite good even if the prior is totally wrong. This is a very important feature when we study the robustness of the GMF. We use this approximation in two different ways

GMF$_{\text{new}_1}$ approximate the likelihood as a Gaussian mixture (9) when we find that the likelihood has one or more peaks. When there is an infinite number of peaks or we do not find any peaks, GMF$_{\text{new}_1}$ works like GMF$_{\text{EKF}}$.

GMF$_{\text{new}_2}$ approximate the likelihood as a Gaussian mixture (9) *only* when we find that the likelihood has a finite number and more than one peak, otherwise GMF$_{\text{new}_2}$ works like GMF$_{\text{EKF}}$.

5 NEW WAY TO ROBUSTIFY GMF

Many robust filters are based on using innovation $y_k - h_k(x_k)$ such that too big innovations are down-weighted or the corresponding measurement discarded [8, 12, 16]. These approaches assume that the prior is correct, which is not always true.

We present a new heuristic to robustify GMF. This method does not replace old methods but complements them, because we can use both methods at same time. If measurements are unlikely (see (10)), we will add new components with appropriate weights. These new components approximate likelihood function and thus do not depend on the prior information. If we assume that the prior is possibly wrong, it is justified to add components approximating likelihood function, because likelihood is proportional to posterior if we have constant prior (prior density with very large covariance). Other components might be appropriately scaled posterior components or prior components.

Next we elaborate on the unlikeliness of measurements. We know that

$$p(y_k|y_{1:k-1}) = \int p(y_k|x_k) p(x_k|y_{1:k-1}) dx_k.$$

Now substitute the prior

$$p(x_k|y_{1:k-1}) = \sum_{i=1}^{p} \alpha_i \, \mathrm{N}^{\mu_i}_{\Sigma_i}(x_k)$$

and the approximate likelihood (9) to obtain (compare to (7))

$$p(y_k|y_{1:k-1}) = \sum_{j=1}^{m} \sum_{i=1}^{p} \alpha_i \, \mathrm{N}^{\mu_{i,j}}_{\Sigma_{i,j}}(y_k),$$

which is also a Gaussian mixture where

$$\mu_{i,j} = h_k(z_j) + \mathrm{H}_{k_j}(\mu_i - z_j) \text{ and}$$
$$\Sigma_{i,j} = \mathrm{H}_{k_j} \Sigma_i \mathrm{H}_{k_j}^T + \mathrm{R}_k.$$

Furthermore,

$$\mathrm{P}\left(\|y_k - \mu_{i,j}\|^2_{\Sigma_{i,j}^{-1}} > a_{\text{th}}, \forall i, j \right) < p_{a_{\text{th}}}, \quad (10)$$

where $\int_{a_{\text{th}}}^{\infty} \chi^2_{n_{y_k}}(x) dx = p_{a_{\text{th}}}$, $\chi^2_{n_{y_k}}(x)$ is density function of $\chi^2(n_{y_k})$-distribution. So now measurements are unlikely if

$$\|y_k - \mu_{i,j}\|^2_{\Sigma_{i,j}^{-1}} > a_{\text{th}}, \forall i, j,$$

where a_{th} is some threshold parameter that is a function of the risk level $p_{a_{\text{th}}}$.

6 COMPONENTS REDUCTION

One major challenge in using GMF efficiently is keeping the number of components as small as possible without losing significant information. There is many ways to do so. We use three different types of mixture reduction algorithms: forgetting, merging and resampling.

Forgetting Give zero weight to mixture components whose weights are lower than some threshold value, for example

$$\min\left(0.001, 0.01 \max_i(\alpha_i)\right).$$

After that, normalize weights of the remaining mixture components.

Merging Merge two mixture components to one if distance between components is lower than some threshold value. Distance is for example [18]

$$d_{ij} = \frac{\alpha_i \alpha_j}{\alpha_i + \alpha_j} (\mu_i - \mu_j)^T \Sigma^{-1} (\mu_i - \mu_j).$$

Merge components so that merging preserves the overall mean and covariance. This method, "collapsing by moments", is optimal in the sense of Kullback-Leibler distance [15].

Resampling If after forgetting and merging there are too many mixture components, we can use a resampling algorithm to choose which mixture components to keep, then normalize the weights of these mixture components. This approach induces less approximation error, using L^1-norm, than merging two distant components.

7 SIMULATIONS

In the simulations, we use the position-velocity model, so the state $x = \begin{bmatrix} r_\mathrm{u} \\ v_\mathrm{u} \end{bmatrix}$ consists of user position vector r_u and user velocity vector v_u, which are in East-North-Up (ENU) coordinate system. In this model the user velocity is a random walk process [7]. Now the state-dynamic Eq. (1) is

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1},$$

where

$$\Phi_{k-1} = \begin{bmatrix} \mathrm{I} & \Delta t_k \mathrm{I} \\ 0 & \mathrm{I} \end{bmatrix},$$

$\Delta t_k = t_k - t_{k-1}$, and w_{k-1} is white, zero mean and Gaussian noise, with covariance matrix

$$\mathrm{Q}_{k-1} = \begin{bmatrix} \frac{\Delta t_k^3 \sigma_\mathrm{p}^2}{3}\mathrm{I} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^3 \sigma_\mathrm{a}^2}{3} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} \\ \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 & \frac{\Delta t_k \sigma_\mathrm{p}^2}{1}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} & 0 & \frac{\Delta t_k \sigma_\mathrm{a}^2}{1} \end{bmatrix},$$

where $\sigma_\mathrm{p}^2 = 2\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors on the East-North plane and $\sigma_\mathrm{a}^2 = 0.1\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors in the vertical direction. [13, 2]

In our simulations, we use base station range measurements, altitude measurements, satellite range measurements and satellite doppler measurements (see (2)).

$$\begin{aligned} y^\mathrm{b} &= \|r_\mathrm{b} - r_\mathrm{u}\| + \epsilon_\mathrm{b}, \\ y^\mathrm{a} &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} r_\mathrm{u} + \epsilon_\mathrm{a}, \\ y^\mathrm{s} &= \|r_\mathrm{s} - r_\mathrm{u}\| + b + \epsilon_\mathrm{s}, \\ \dot{y}^\mathrm{s} &= \frac{(r_\mathrm{s} - r_\mathrm{u})^T}{\|r_\mathrm{s} - r_\mathrm{u}\|}(v_\mathrm{s} - v_\mathrm{u}) + \dot{b} + \dot{\epsilon}_\mathrm{s}, \end{aligned}$$

where r_b is a base station position vector, r_s is a satellite position vector, b is clock bias, v_s is a satellite velocity vector, $\dot{b}$ is clock drift and ϵ:s are error terms. We use satellite measurements only when there is more than one satellite measurement available, so that bias can be eliminated. These are the same measurements equations as in the papers [2, 1].

7.1 Example where GMF beats EKF

As mentioned before, EKF has a serious consistency problem in highly nonlinear situations, which means that EKF does not work correctly [2]. One example of highly nonlinear situations is shown in Fig. 1, where we have only range measurements from two base stations. In this case EKF's estimate veers away from the true route and gets stuck in an incorrect solution branch. GMF, which approximates likelihood as a Gaussian mixture (see (9)) finds both peaks of the posterior function. So in this case we can say that GMF works better than EKF. Nevertheless, it is important to notice that before EKF select the incorrect solution branch it gives better estimates than GMF. This is because prior mean, which is the linearization point of EKF, is usually a better state estimate than likelihood peaks, around which GMF is linearized. Thus EKF usually gives smaller 2D root-mean-square position error than the variant of GMF that always uses likelihood peak(s) as linearization points (see section (7.2)).

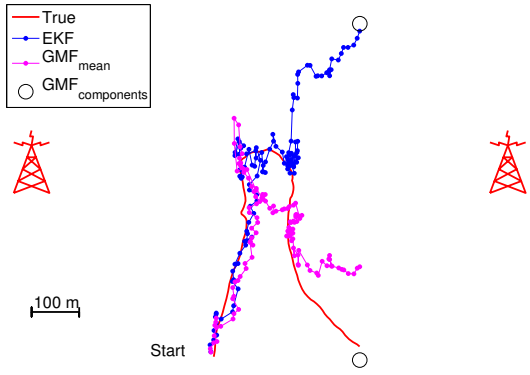

Figure 1: One example where GMF beats EKF. Here true posterior has two peaks and GMF finds them but EKF selects incorrect solution branch.

7.2 Summary of suburban cases

In Table 1, we have listed a summary of two hundred 120 second simulations, which use typical suburban geometry. This means that simulations use a few base station range measurements with variance $(100\ \mathrm{m})^2$, inaccurate altitude measurements with variance $\approx (75\ \mathrm{m})^2$, some (not more than five) satellite pseudorange and doppler measurements, with variance $\approx (25\ \mathrm{m})^2$ and $\approx (2\frac{\mathrm{m}}{\mathrm{s}})^2$ respectively. Summary consist of following columns: *Time* is computation time using Matlab in our implementation, scaled so that computation time of EKF is 1. This gives a rough idea of the time complexity of each algorithm. *Err. rms*

is 2D root mean square position error. *Err. 95%* gives a radius containing 95 % of the 2D errors. *Err. ref.* is 2D error to reference posterior mean, which is computed using a particle filter with 10^6 particles. *Inc. %* is percentage of time where filter is inconsistent with respect to the general inconsistency test, with risk level 5% [2]. Solvers are sorted so that rms errors are in descending order.

Different GMF subscripts indicate how solvers make the updating step, see section 4. Every GMF variant adds components that approximate the likelihood whenever measurements are unlikely (see section 5). Also, every GMF variant keeps the number of mixture components very small (not more than eight) using component reduction (see section 6). PF_{10^3} indicates particle filter with systematic resampling and 10^3 particles [4].

Table 1: Summary of 200 different simulations with typical suburban geometry. Simulations use a few base station range measurements, altitude measurements, some satellite pseudorange and doppler measurements.

Solver	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
$\text{GMF}_{\text{new}_1}$	6	189	381	66	1.7
EKF	1	183	368	36	2.8
UKF	3	183	361	24	1.5
EKF2	1	182	374	27	1.4
GMF_{ekf2}	2	178	366	28	0.9
GMF_{ukf}	5	177	356	24	1.0
GMF_{ekf}	2	173	352	34	2.2
$\text{GMF}_{\text{new}_2}$	6	172	332	41	1.6
PF_{10^3}	5	153	326	22	0.5
Ref	∞	146	307	0	0.1

The following conclusions can be drawn based on simulations and theory.

- The new GMF robustifying heuristic (section 5) improves results a bit, even if simulated data does not have any outliers. That is, GMF_{EKF}, GMF_{EKF2} and GMF_{UKF} give better results than EKF, EKF2 and UKF, respectively. One reason for this is that there are some simulations where EKF, EKF2 and UKF are inconsistent and adding components which approximate the likelihood to GMF gives better results.
- $\text{GMF}_{\text{new}_1}$ has bigger errors than EKF, which indicates that it is not always reasonable to approximate likelihood as a Gaussian mixture.
- $\text{GMF}_{\text{new}_2}$ gives better results than other GMFs, especially $\text{GMF}_{\text{new}_1}$ and EKF. So it is reasonable to approximate likelihood as Gaussian mixture only when there is risk that posterior has multiple significant peaks, e.g. when likelihood has more than one peak.
- It seems that PF_{10^3} gives better results than the GMFs, with simulated data without outliers. The basic PF_{10^3} is not expected to work very well with real data.
- Even if $\text{GMF}_{\text{new}_2}$ gives better results (especially Err. 95%) than traditional GMFs such as EKF, there are still some inconsistency problems.

8 CONCLUSIONS

In this article, we have studied three aspects of the Gaussian Mixture Filter: how to approximate the likelihood function as a Gaussian mixture, a heuristic method to robustify GMF by adding more components, and different ways to reduce the number of mixture components so that the algorithm is fast without losing significant information. We also compared GMFs with different filters using simulations. We test especially two Gaussian mixture filters that use a new way to robustify GMF, use at most eight mixture components and approximate likelihood as Gaussian mixture. $\text{GMF}_{\text{new}_2}$, which approximates likelihood as Gaussian mixture only when likelihood has more than one peak and otherwise computes updating step like EKF, works much better than EKF. $\text{GMF}_{\text{new}_1}$ is the same as $\text{GMF}_{\text{new}_2}$ except that it approximates likelihood as Gaussian mixture always when likelihood has peak(s). Simulations show that the $\text{GMF}_{\text{new}_2}$ works better than EKF, EKF2 or UKF, with respect to the estimation errors. However, $\text{GMF}_{\text{new}_1}$ gives larger errors than EKF. So this new method approximates likelihood as a Gaussian mixture, beats EKF actually only when EKF has possible problems e.g. posterior has multiple peaks. Simulations also show that "robust" GMF gives better results than basic GMF.

A POSTERIOR OF GMF

We assume that prior (3) is

$$p(x) = \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x),$$

and likelihood (5) is

$$p(y|x) = \sum_{j=1}^{m} \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y).$$

Then the posterior is (4)

$$\begin{aligned} p(x|y) &= \frac{\sum_{j=1}^{m} \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y) \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x)}{\int \sum_{j=1}^{m} \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y) \sum_{i=1}^{p} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x) dx} \\ &= \frac{\sum_{j=1}^{m} \sum_{i=1}^{p} \alpha_i \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y) \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x)}{\sum_{j=1}^{m} \sum_{i=1}^{p} \alpha_i \beta_j \int \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y) \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x) dx} \\ &\overset{(11)}{=} \frac{\sum_{j=1}^{m} \sum_{i=1}^{p} \alpha_i \beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j \mu_i}(y) \, \mathrm{N}_{\hat{\mathrm{P}}_{i,j}}^{\hat{x}_{i,j}}(x)}{\sum_{j=1}^{m} \sum_{i=1}^{p} \alpha_i \beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j \mu_i}(y)}, \end{aligned}$$

where

$$\begin{aligned}
\mathrm{P}_{i,j} &= \mathrm{H}_j \Sigma_i \mathrm{H}_j^T + \mathrm{R}, \\
\hat{x}_{i,j} &= \mu_i + \mathrm{K}_{i,j}(y - \mathrm{H}_j \mu_i), \\
\hat{\mathrm{P}}_{i,j} &= (\Sigma_i^{-1} + \mathrm{H}_j^T \mathrm{R}^{-1} \mathrm{H}_j)^{-1} \\
&= (\mathrm{I} - \mathrm{K}_{i,j}\mathrm{H}_j)\Sigma_i \text{ and} \\
\mathrm{K}_{i,j} &= \Sigma_i \mathrm{H}_j^T \mathrm{P}_{i,j}^{-1}.
\end{aligned}$$

Here we use the fact that

$$\begin{aligned}
\mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y)\, \mathrm{N}_{\Sigma_i}^{\mu_i}(x) &= \frac{\exp\left(-\frac{1}{2}a_2\right)}{\sqrt{2\pi}^{(n_y+n_x)}\sqrt{a_1}} \\
&= \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j\mu_i}(y)\, \mathrm{N}_{\hat{\mathrm{P}}_{i,j}}^{\hat{x}_{i,j}}(x),
\end{aligned} \tag{11}$$

where

$$\begin{aligned}
a_1 &= \det(\mathrm{R})\det(\Sigma_i) \\
&= \det\left(\begin{bmatrix} \Sigma_i & 0 \\ 0 & \mathrm{R} \end{bmatrix}\right) \\
&= \det\left(\begin{bmatrix} \mathrm{I} & 0 \\ \mathrm{H}_j & \mathrm{I} \end{bmatrix}\begin{bmatrix} \Sigma_i & 0 \\ 0 & \mathrm{R} \end{bmatrix}\begin{bmatrix} \mathrm{I} & \mathrm{H}_j^T \\ 0 & \mathrm{I} \end{bmatrix}\right) \\
&= \det\left(\begin{bmatrix} \Sigma_i & \Sigma_i \mathrm{H}_j^T \\ \mathrm{H}_j\Sigma_i & \mathrm{H}_j\Sigma_i\mathrm{H}_j^T + \mathrm{R} \end{bmatrix}\right) \\
&= \det\left(\begin{bmatrix} \mathrm{I} & \mathrm{K}_{i,j} \\ 0 & \mathrm{I} \end{bmatrix}\begin{bmatrix} \hat{\mathrm{P}}_{i,j} & 0 \\ \mathrm{H}_j\Sigma_i & \mathrm{P}_{i,j} \end{bmatrix}\right) \\
&= \det(\mathrm{P}_{i,j})\det(\hat{\mathrm{P}}_{i,j})
\end{aligned}$$

and

$$\begin{aligned}
a_2 &= \|y - \mathrm{H}_j x\|^2_{\mathrm{R}^{-1}} + \|x - \mu_i\|^2_{\Sigma_i^{-1}} \\
&= \|x\|^2_{\mathrm{H}_j^T\mathrm{R}^{-1}\mathrm{H}_j + \Sigma_i^{-1}} - 2(y^T\mathrm{R}^{-1}\mathrm{H}_j + \mu_i^T\Sigma_i^{-1})x \ldots \\
&\quad + \|y\|^2_{\mathrm{R}^{-1}} + \|\mu_i\|^2_{\Sigma_i^{-1}} \\
&= \|x - \hat{x}_{i,j}\|^2_{\hat{\mathrm{P}}_{i,j}^{-1}} - \|\mathrm{H}_j^T\mathrm{R}^{-1}y + \Sigma_i^{-1}\mu_i\|^2_{\hat{\mathrm{P}}_{i,j}} \ldots \\
&\quad + \|y\|^2_{\mathrm{R}^{-1}} + \|\mu_i\|^2_{\Sigma_i^{-1}} \\
&= \|x - \hat{x}_{i,j}\|^2_{\hat{\mathrm{P}}_{i,j}^{-1}} - 2\mu_i^T\Sigma_i^{-1}\hat{\mathrm{P}}_{i,j}\mathrm{H}_j^T\mathrm{R}^{-1}y \ldots \\
&\quad + \|y\|^2_{\mathrm{P}_{i,j}^{-1}} + \|\mu_i\|^2_{\mathrm{H}_j^T\mathrm{P}_{i,j}^{-1}\mathrm{H}_j} \\
&= \|x - \hat{x}_{i,j}\|^2_{\hat{\mathrm{P}}_{i,j}^{-1}} + \|y - \mathrm{H}_j\mu_i\|^2_{\mathrm{P}_{i,j}^{-1}}.
\end{aligned}$$

B COMPUTING LIKELIHOOD PEAKS

In this section, we give a strategy to compute (8)

$$z = \mathrm{argmin}_r \left[(y - h(r))^T \mathrm{R}^{-1}(y - h(r))\right], \tag{12}$$

where r is position of mobile station. Assume that we have n_r range measurements and n_l approximate linear measurements, for example pseudorange difference measurements or altitude measurement. Here we assume that

$$\dim(y) = n_\mathrm{r} + n_\mathrm{l} \geq \dim(r) = 3.$$

Now

$$h(r) = \begin{bmatrix} \|r_1 - r\| \\ \vdots \\ \|r_{n_\mathrm{r}} - r\| \\ \mathrm{H}r \end{bmatrix}.$$

Matrix R^{-1} is positive definite so

$$(y - h(r))^T \mathrm{R}^{-1} (y - h(r)) \geq 0,$$

and equality holds only when $y = h(r)$. First, we try to find closed-form solutions z so that $y = h(z)$ [20, 19, 5]. If we have only linear measurements we get

$$\begin{aligned}
z &= \left(\mathrm{H}^T\mathrm{R}^{-1}\mathrm{H}\right)^{-1}\mathrm{H}^T\mathrm{R}^{-1}y \\
&= \mathrm{argmin}_r \left[(y - \mathrm{H}r)^T\mathrm{R}^{-1}(y - \mathrm{H}r)\right],
\end{aligned} \tag{13}$$

if $\left(\mathrm{H}^T\mathrm{R}^{-1}\mathrm{H}\right)^{-1}$ exists, otherwise we have an infinite number of solutions. If there is no closed-form solution and we have at least one range measurement we use iterative weighted least squares (IWLS) to find solution. The initial iterate z_0 for WLS-algorithm can be found, for example, by omitting some measurements and using the closed-form solution, see equation (13), or by one of the following closed-form equations, respectively.

One range meas and two linear meas

$$z = r_{\text{line}} + n_{\text{line}}^T(r_1 - r_{\text{line}})n_{\text{line}},$$

where

$$r_{\text{line}} = \begin{bmatrix} n_{\text{line}}^T \\ \mathrm{H} \end{bmatrix}^{-1} y, \text{ and } n_{\text{line}} = \frac{\mathrm{H}_{1,:}^T \times \mathrm{H}_{2,:}^T}{\|\mathrm{H}_{1,:}^T \times \mathrm{H}_{2,:}^T\|},$$

where we assume that $\|\mathrm{H}_{1,:}^T \times \mathrm{H}_{2,:}^T\| > 0$. Here z is the point on a line $r = r_{\text{line}} + kn_{\text{line}}$, for which distance from sphere $y_1 = \|r_1 - r\|$ is minimum. This straight line does not intersect the sphere because we have assumption that we do not have closed-form solution.

Two range meas and one linear meas We select our indices so that $y_1 \geq y_2$. We assume that matrix $\mathrm{R} = \mathrm{diag}[\sigma_1^2, \sigma_2^2, \sigma_3^2]$. There are three cases:

Separate spheres $(y_1 - \|r_1 - r_2\| < -y_2)$

$$z = r_1 + k\frac{r_2 - r_1}{\|r_2 - r_1\|}, \tag{14}$$

where

$$k = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\left(\frac{y_1}{\sigma_1^2} + \frac{\|r_2 - r_1\| - y_2}{\sigma_2^2}\right).$$

Intersecting spheres $(-y_2 \leq y_1 - \|r_1 - r_2\| \leq y_2)$

$$z = r_0 \pm \sqrt{y_1^2 - \|r_1 - r_0\|^2}\frac{(\mathrm{H}_{1,:}^T - (H_{1,:}n)n)}{\|\mathrm{H}_{1,:}^T - (H_{1,:}n)n\|},$$

where

$$\begin{aligned}
\pm &= \frac{y_3 - \mathrm{H}_{1,:}r_0}{|y_3 - \mathrm{H}_{1,:}r_0|} \\
r_0 &= \frac{r_2 + r_1}{2} + \frac{y_1^2 - y_2^2}{2\|r_2 - r_1\|}n \\
n &= \frac{r_2 - r_1}{\|r_2 - r_1\|}.
\end{aligned}$$

Sphere inside another $(y_2 < y_1 - \|r_1 - r_2\|)$

$$z = r_1 + k\frac{r_2 - r_1}{\|r_2 - r_1\|}, \qquad (15)$$

where

$$k = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\left(\frac{y_1}{\sigma_1^2} + \frac{\|r_2 - r_1\| + y_2}{\sigma_2^2}\right).$$

Three range meas Use only two range measurements and use previous item. If these two spheres intersect then replace $\mathrm{H}_{1,:}^T = r_0 - r_3$ and $\pm = +$.

One interesting observation is that if we have only two range measurements and they do not intersect, then equations (14) or (15) give analytic solution of (12) regardless of dimension of state r.

ACKNOWLEDGMENTS

This study was partly funded by Nokia Corporation. Simo Ali-Löytty acknowledges the financial support of the Nokia Foundation and the Tampere Graduate School in Information Science and Engineering. The particle filter we used was implemented by Duane Petrovich.

REFERENCES

[1] S. Ali-Löytty and N. Sirola. A modified Kalman filter for hybrid positioning. In *Proceedings of ION GNSS 2006*, September 2006.

[2] S. Ali-Löytty, N. Sirola, and R. Piché. Consistency of three Kalman filter extensions in hybrid navigation. In *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, July 2005.

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

[4] 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, 2002.

[5] S. Bancroft. An algebraic solution of the GPS equations. *IEEE Transactions on Aerospace and Electronic Systems*, 21(7):56–59, 1986.

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

[7] R. G. Brown. *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, 1983.

[8] A. Carosio, A. Cina, and M. Piras. The robust statistics method applied to the Kalman filter. In *ION GNSS 18th International Technical Meeting of the Satellite Division*, September 2005.

[9] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.

[10] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *American Control Conference*, volume 3, pages 1628–1632, 1995.

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

[12] 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):361–371, 1977.

[13] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141 of *Mathematics in Science and Engineering*. Academic Press, 1979.

[14] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 141-2 of *Mathematics in Science and Engineering*. Academic Press, 1982.

[15] 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.

[16] T. Perälä and R. Piché. Robust extended Kalman filtering in hybrid positioning applications. In *4th Workshop on Positioning, Navigation and Communication 2007 (WPNC' 07)*, 2007.

[17] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Artech House, Boston, London, 2004.

[18] D. J. Salmond. Mixture reduction algorithms for target tracking. *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, pages 7/1–7/4, 1989.

[19] N. Sirola. A versatile algorithm for local positioning in closed form. In *Proceedings of the 8th European Navigation Conference GNSS 2004, May 16-19, Rotterdam*, 2004.

[20] N. Sirola, R. Piché, and J. Syrjärinne. Closed-form solutions for hybrid cellular/GPS positioning. In *Proceedings of the ION GPS/GNSS 2003*, pages 1613–1619, 2003.

PUBLICATION 4

Simo Ali-Löytty and Niilo Sirola: Gaussian mixture filter and hybrid positioning. In *Proceedings of the 20th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2007)*, Fort Worth TX, September 2007, pages 562–569.

Gaussian Mixture Filters and Hybrid Positioning

Simo Ali-Löytty and Niilo Sirola, *Institute of Mathematics, Tampere University of Technology, Finland*

BIOGRAPHY

Simo Ali-Löytty is a Ph.D. student at Tampere University of Technology. He received his M.Sc. degree in 2004 from the same university. His research interest is filters in personal positioning.

Niilo Sirola received his M.Sc. degree from Tampere University of Technology in 2003. He has been studying positioning algorithms since 2000, and is currently working on his Ph.D. thesis on novel mathematical methods for personal positioning and navigation.

ABSTRACT

This paper presents, develops and compares Gaussian Mixture Filter (GMF) methods for hybrid positioning. The key idea of the developed method is to approximate the prior density as a Gaussian mixture with a small number of mixture components. We show why it is sometimes reasonable to approximate a Gaussian prior with a multi-component Gaussian mixture. We also present both simulated and real data tests of different filters in different scenarios. Simulations show that GMF gives better accuracy than Extended Kalman Filter with lower computational requirements than Particle Filter, making it a reasonable algorithm for the hybrid positioning problem.

1 INTRODUCTION

Hybrid positioning means that measurements used in positioning come from many different sources e.g. Global Navigation Satellite System (GNSS), Inertial Measurement Unit (IMU), or local wireless networks such as a cellular network, WLAN, or Bluetooth. Range, pseudorange, delta range, altitude, base station sector and compass measurements are examples of typical measurements in hybrid positioning.

Positioning filters are used to compute an estimate of state variables such as position, velocity, attitude using current and past measurement data. A consistent filter also provides correct information on the accuracy of its state estimate, e.g. in the form of an estimated error covariance. The Extended Kalman Filter (EKF) [6, 10], which is very commonly used in satellite based positioning, has also been applied to hybrid positioning [12, 9]. Unfortunately, EKF can be badly inconsistent in highly nonlinear situations, and such situations are much more common in hybrid positioning than when using only GNSS measurements [3]. Because of this, many researchers have proposed using a general nonlinear Bayesian filter, which is usually implemented as a particle filter or a point mass filter. If the dynamics and measurement models are correct, these filters usually work and give good positioning accuracy but require much computation time.

In this paper, we consider the family of Gaussian Mixture Filters [18, 4]. Generally, GMF is a filter whose approximate prior and posterior densities are Gaussian mixtures, meaning a linear combination of Gaussian densities where weights are between 0 and 1. GMF is an extension of Kalman type filter. In particular, EKF, Unscented Kalman Filter (UKF) [11] and a bank of EKF are special cases of GMF and this is one motivation for considering GMF in hybrid positioning. A second motivation is the fact that any probability density can be approximated as closely as desired with a Gaussian mixture. The third motivation is that the GMF is a very flexible algorithm. It can take into consideration for example multiple static solutions [2], multiple measurement or dynamic models and mixture error models.

An outline of the paper is as follows. In section 2, we glance at Bayesian filtering. In section 3, we study basics of the Gaussian Mixture Filter. In section 4, we discuss different possibilities when it is reasonable to use GMF. In section 5 we present a method for splitting (approximating) Gaussian as Gaussian mixture. Finally, we present simulation results where we compare different GMFs and a bootstrap particle filter [5].

2 BAYESIAN FILTERING

We consider the discrete-time non-linear non-Gaussian system

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \tag{1}$$

$$y_k = h_k(x_k) + v_k, \tag{2}$$

where the vectors $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_{y_k}}$ represent the state of the system and the measurement at time t_k, $k \in \mathbb{N}$, respectively. We assume that errors w_k and v_k are white, mutually independent and independent of the initial state x_0. We denote the density functions of w_k and v_k by p_{w_k} and p_{v_k}, respectively. The aim of filtering is to find the conditional probability density function (posterior)

$$p(x_k|y_{1:k}),$$

where $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$. The posterior can be determined recursively according to the following relations.
Prediction (prior):

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}; \tag{3}$$

Update (posterior):

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k}, \tag{4}$$

where the transition pdf is

$$p(x_k|x_{k-1}) = p_{w_{k-1}}(x_k - f_{k-1}(x_{k-1}))$$

and the likelihood

$$p(y_k|x_k) = p_{v_k}(y_k - h_k(x_k)). \tag{5}$$

The initial condition for the recursion is given by the pdf of the initial state $p(x_0|y_{1:0}) = p(x_0)$. Knowledge of the posterior distribution (4) enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error (MMSE) estimate is the conditional mean of x_k [6, 16]. In general and in our case, the conditional probability density function cannot be determined analytically.

3 GAUSSIAN MIXTURE FILTER

The idea of GMF [18, 4] (also called Gaussian Sum Filter) is that both prior density (3) and posterior density (4) are Gaussian mixtures

$$p(x) = \sum_{i=1}^{N} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x), \tag{6}$$

where $\mathrm{N}_{\Sigma_i}^{\mu_i}(x)$ is the normal density function with mean μ_i and covariance matrix Σ_i. Weights are non-negative and sum to one. The mean of a Gaussian mixture (6) is (see appendix A)

$$\mu_{\text{gm}} = \sum_{i=1}^{N} \alpha_i \mu_i$$

and the covariance matrix is

$$\Sigma_{\text{gm}} = \sum_{i=1}^{N} \alpha_i (\Sigma_i + (\mu_i - \mu)(\mu_i - \mu)^T).$$

3.1 Example of GMF

Now we assume that the prior (3) and likelihood (5) are

$$p(x) = \sum_{i=1}^{N} \alpha_i \, \mathrm{N}_{\Sigma_i}^{\mu_i}(x) \quad \text{and}$$

$$p(y|x) = \sum_{j=1}^{M} \beta_j \, \mathrm{N}_{\mathrm{R}}^{\mathrm{H}_j x}(y).$$

Then the posterior is a Gaussian mixture (see [2])

$$p(x|y) = \frac{\sum_{j=1}^{M}\sum_{i=1}^{N} \alpha_i \beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j \mu_i}(y) \, \mathrm{N}_{\hat{\mathrm{P}}_{i,j}}^{\hat{x}_{i,j}}(x)}{\sum_{j=1}^{M}\sum_{i=1}^{N} \alpha_i \beta_j \, \mathrm{N}_{\mathrm{P}_{i,j}}^{\mathrm{H}_j \mu_i}(y)},$$

where

$$\begin{aligned}
\mathrm{P}_{i,j} &= \mathrm{H}_j \Sigma_i \mathrm{H}_j^T + \mathrm{R}, \\
\hat{x}_{i,j} &= \mu_i + \mathrm{K}_{i,j}(y - \mathrm{H}_j \mu_i), \\
\hat{\mathrm{P}}_{i,j} &= (\mathrm{I} - \mathrm{K}_{i,j}\mathrm{H}_j)\Sigma_i, \text{ and} \\
\mathrm{K}_{i,j} &= \Sigma_i \mathrm{H}_j^T \mathrm{P}_{i,j}^{-1}.
\end{aligned}$$

3.2 Reduction number of components

One major challenge in using GMF efficiently is keeping the number of components as small as possible without losing significant information. There are many ways to do so. We use three different types of mixture reduction algorithms: forgetting, merging and resampling. [2, 18, 17].

Reducing the number of components makes it possible that when the situation becomes favourable to a one component filter (e.g. EKF) we can merge mixture components into one, and after which the filter works identically to a one component filter.

4 WHERE DO MIXTURE COMPONENTS COME FROM?

It is reasonable to ask why to use Gaussian Mixture Filter, rather than a Gaussian Filter e.g. EKF. It is well known

that in linear-Gaussian case the Kalman Filter is an analytic solution for Bayesian filtering and, of course, in a linear-Gaussian case we should use the Kalman Filter. Nevertheless, there are a lot of possible reasons why it is reasonable use GMF. Sometimes Gaussian density function is not a good enough approximation of our exact posterior. Possible reasons to why exact posterior is not Gaussian and how mixture components arise are:

Initial state is not Gaussian Usually this kind of problems can be computed using multiple parallel Gaussian filters, this method is also called the bank of Gaussian filters. Banks of Gaussian filters are special cases of GMF.

Multiple Models If we use a multiple Gaussian model approach [6] or the error density functions are not Gaussian then we come to GMF.

Nonlinearity of models If our state model (1) and/or measurement model (2) are nonlinear then usually posterior is not Gaussian. Here we consider only the nonlinearity of the measurement model. When the measurement model is nonlinear there are at least two ways to use GMF. It is possible to approximate likelihood function (5) as Gaussian Mixture, which is very reasonable when we have multiple static positions/velocity solutions (see paper [2]). Second possibility is to approximate prior density as Gaussian Mixture so that covariance matrices of mixture components are small enough. This method is the main contribution of this paper. We present this method in section 5.

This is not a complete list of where mixture components arise, but more like an example of different possibilities. One further example is heuristic (robust) approach, which also leads to GMF, as presented in paper [2]. The idea of this heuristic method is to add more mixture components which approximate likelihood when all measurements are unlikely.

5 SPLITTING A SINGLE GAUSSIAN INTO A GAUSSIAN MIXTURE

As we know, EKF in hybrid positioning [3] has a consistency problem. The key reason for inconsistency is nonlinearity. Now we consider how we can measure and overcome nonlinearity. We assume that n-dimensional prior state is

$$x \sim \mathrm{N}(\hat{x}, \mathrm{P}) \tag{7}$$

and measurement model is

$$y = h(x) + v,$$

where measurement error $v \sim \mathrm{N}(0, \sigma^2)$, $(\sigma > 0)$, is independent of the prior state. We consider a scalar measurement case. We say that the nonlinearities are significant if

$$\mathrm{tr}(\mathrm{H}_e \mathrm{P} \mathrm{H}_e \mathrm{P}) \gtrsim \sigma^2,$$

where H_e is Hessian matrix of $h(x)$ [3, 10]. This definition comes from comparing EKF and Second Order EKF (EKF2) [6], which is also called Modified Gaussian Second Order Filter [10, 15]. We use function

$$Nonlinearity = \frac{\sqrt{\mathrm{tr}(\mathrm{H}_e \mathrm{P} \mathrm{H}_e \mathrm{P})}}{\sigma} - 1, \tag{8}$$

which gives positive values when nonlinearity is significant. One possibility to overcome nonlinearity is to approximate prior density function $p(x)$ as a Gaussian mixture

$$p(x) \approx \sum_{i=0}^{N} \alpha_i \, \mathrm{N}_{\mathrm{P}_i}^{\hat{x}_i}(x), \tag{9}$$

so that covariance matrices P_i are smaller than the prior covariance matrix P. This approximation (9) can be formed in many ways. Here we present an approximation loosely based on the sigma points [11] and we call this approximation the sigma point Gaussian mixture (SPGM). This SPGM approximation contains $2n + 1$ mixture components, whose covariance matrices are

$$\mathrm{P}_i = (1 - \tau^2)\mathrm{P},$$

weights α_i and means $\hat{x}_i$ are listed in Table 1. Parameters τ and κ have the restrictions $\tau \in (0, 1)$ and $\kappa > 0$. These restrictions ensure that approximation (9) is a proper density function. We find that SPGM has the same mean, covariance matrix, and third moments as the original Gaussian distribution (7) (see Appendix B).

Table 1: Weights and means of Gaussian mixture approximation (9) of Gaussian

Ind.	α_i	$\hat{x}_i$
0	$\frac{\kappa}{\kappa+n}$	$\hat{x}$
$1, \ldots, n$	$\frac{1}{2(\kappa+n)}$	$\hat{x} + \tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_i$
$n+1, \ldots, 2n$	$\frac{1}{2(\kappa+n)}$	$\hat{x} - \tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_{i-n}$

5.1 Example of SPGM

This example presents a comparison between SPGM-EKF posterior and EKF posterior, when prior distribution is

$$x \sim \mathrm{N}\left(\begin{bmatrix} d \\ 0 \end{bmatrix}, \begin{bmatrix} 100^2 & 0 \\ 0 & 300^2 \end{bmatrix} \right), \tag{10}$$

and we get one base station range measurement (2), where

$$y = 1000, h(x) = \|x\| \text{ and } v \sim \mathrm{N}(0, 100^2).$$

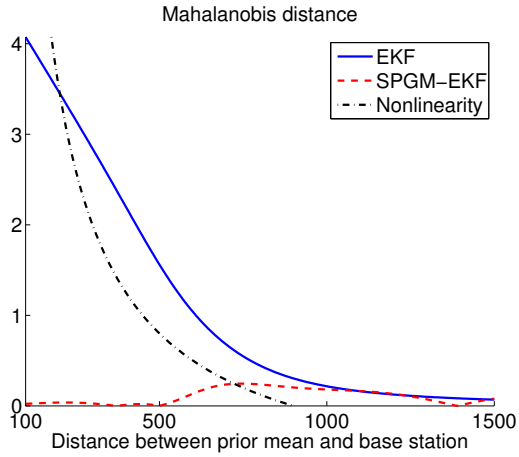

Figure 1: Mahalanobis distance between exact mean and mean approximations. Significance of nonlinearity (see equation (8)) is also shown.

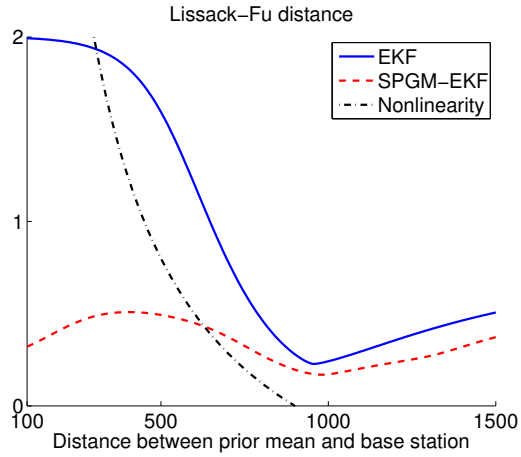

Figure 2: Lissack-Fu distance between exact posterior and posterior approximations. Significance of nonlinearity (see equation (8)) is also shown.

The base station is located in origin and so d is the distance between prior mean and the base station. "Exact" posterior density function $p_{\text{exact}}(x)$ is computed using a point-mass filter, with 500^2 points [8]. Approximation of exact density function is computed using EKF and SPGM-EKF. SPGM-EKF first approximates prior (10) using SPGM approximation, with parameters $\kappa = 4$ and $\tau = \frac{1}{2}$ (see section 5). After that SPGM-EKF computes posterior approximation using EKF for each component and finally assigns these components appropriate weights.

Comparison between SPGM-EKF and EKF contains two parts. First we compute Mahalanobis distance between exact posterior mean and the approximations means

$$\sqrt{(\mu_{\text{exact}} - \mu_{\text{app}})^T \Sigma_{\text{exact}}^{-1} (\mu_{\text{exact}} - \mu_{\text{app}})}.$$

These results are shown in Figure 1. The value of the *Nonlinearity* function (8) is also plotted in Figure 1. We see that Mahalanobis distance between EKF mean and exact mean increases rapidly when nonlinearity becomes more significant. In that case SPGM-EKF gives much better results than EKF. Furthermore, SPGM-EKF gives always as good results as EKF even when there is no significant nonlinearity.

Second, we compute the first order Lissack-Fu distance

$$\int |p_{\text{exact}}(x) - p_{\text{app}}(x)| dx,$$

between exact posterior and the approximations (also called total variation norm). These results are in Figure 2. The value of the *Nonlinearity* function (8) is also plotted in Figure 2. We see that SPGM-EKF gives smaller Lissack-Fu distance than EKF. Difference between SPGM-EKF Lissack-Fu distance and EKF Lissack-Fu distance increases when nonlinearity becomes more significant. EKF Lissack-Fu distance is almost 2 (maximum value) when $d = 100$, so the exact posterior and the EKF posterior approximation are almost totally separate.

In this example, SPGM-EKF gives much better results than EKF when nonlinearity is significant. One interesting finding is that in this example SPGM-EKF works at least as well as EKF in all cases. This is very important and nice property when we consider cases where we have many measurements at the same time and only some of the measurements have significant nonlinearity. This finding predicts also that in these cases SPGM-EKF works better than EKF. Our simulations also verify this assumption (see section 6).

6 SIMULATIONS

In the simulations, we use the position-velocity model, so the state $x = \begin{bmatrix} r_{\text{u}} \\ v_{\text{u}} \end{bmatrix}$ consists of user position vector r_{u} and user velocity vector v_{u}, which are in East-North-Up (ENU) coordinate system. In this model the user velocity is a random walk process [7]. Now the state-dynamic (1) is

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1},$$

where

$$\Phi_{k-1} = \begin{bmatrix} \text{I} & \Delta t_k \text{I} \\ 0 & \text{I} \end{bmatrix},$$

$\Delta t_k = t_k - t_{k-1}$, and w_{k-1} is white, zero mean and Gaussian noise, with covariance matrix

$$\mathrm{Q}_{k-1} = \begin{bmatrix} \frac{\Delta t_k^3 \sigma_\mathrm{p}^2}{3}\mathrm{I} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^3 \sigma_\mathrm{a}^2}{3} & 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} \\ \frac{\Delta t_k^2 \sigma_\mathrm{p}^2}{2}\mathrm{I} & 0 & \frac{\Delta t_k \sigma_\mathrm{p}^2}{1}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^2 \sigma_\mathrm{a}^2}{2} & 0 & \frac{\Delta t_k \sigma_\mathrm{a}^2}{1} \end{bmatrix},$$

where $\sigma_\mathrm{p}^2 = 2\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors on the East-North plane and $\sigma_\mathrm{a}^2 = 0.1\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors in the vertical direction. [14, 3]

In our simulations, we use base station range measurements, altitude measurements, satellite pseudorange measurements and satellite delta range measurements (see (2)).

$$\begin{aligned} y^\mathrm{b} &= \|r_\mathrm{b} - r_\mathrm{u}\| + \epsilon_\mathrm{b}, \\ y^\mathrm{a} &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} r_\mathrm{u} + \epsilon_\mathrm{a}, \\ y^\mathrm{s} &= \|r_\mathrm{s} - r_\mathrm{u}\| + b + \epsilon_\mathrm{s}, \\ \dot{y}^\mathrm{s} &= \frac{(r_\mathrm{s} - r_\mathrm{u})^T}{\|r_\mathrm{s} - r_\mathrm{u}\|}(v_\mathrm{s} - v_\mathrm{u}) + \dot{b} + \dot{\epsilon}_\mathrm{s}, \end{aligned}$$

where r_b is a base station position vector, r_s is a satellite position vector, b is clock bias, v_s is a satellite velocity vector, $\dot{b}$ is clock drift and ϵ:s are error terms. We use satellite measurements only when there is more than one satellite measurement available, so that bias can be eliminated. These are the same measurements equations as in the papers [2, 1].

6.1 Summary of suburban cases

On Table 2, we have listed a summary of two hundred 120 second simulations, which use typical suburban geometry. This means that simulations use a few base station range measurements with variance $(100\ \mathrm{m})^2$, inaccurate altitude measurements with variance $\approx (75\ \mathrm{m})^2$, some (not more than five) satellite pseudorange and delta range measurements, with variance $\approx (25\ \mathrm{m})^2$ and $\approx (2\frac{\mathrm{m}}{\mathrm{s}})^2$ respectively. Summary consist of following columns: *Time* is computation time using Matlab in our implementation, scaled so that computation time of EKF is 1. This gives a rough idea of the time complexity of each algorithm. *Err. rms* is 2D root mean square position error. *Err. 95%* gives a radius containing 95 % of the 2D errors. *Err. ref.* is 2D error to reference posterior mean, which is computed using a particle filter with 10^6 particles. *Inc. %* is a percentage of time where the filter is inconsistent with respect to the general inconsistency test, with risk level 5% [3]. Solvers are sorted so that rms errors are in descending order. These testcases and filters EKF, UKF and $\mathrm{GMF}_{\mathrm{new}_2}$, are the same as in the paper [2].

$\mathrm{GMF}_{\mathrm{new}_2}$ approximates the likelihood as a Gaussian mixture only when we find that the likelihood has a finite number and more than one peak, otherwise $\mathrm{GMF}_{\mathrm{new}_2}$ works like EKF. $\mathrm{GMF}_{\mathrm{new}_2}$ uses also a robust method, that is it adds components that approximate the likelihood whenever measurements are unlikely. See more details in paper [2].

$\mathrm{GMF}_{\mathrm{new}_2}$ is the best GMF from the paper [2]. Other GMFs are named in the following way: Superscript tells how GMF makes the updating step. GMF takes one component at the time and uses same updating formula than the superscript filter e.g. EKF. In the update step, GMF update also weight of the component. Updated weight is proportional to $w_\mathrm{old}\, \mathrm{N}_\mathrm{R}^0(innovation)$. Subscript indicates different sources of mixture components so that

S: means that we *S*plit prior components to Gaussian mixture if nonlinearities are significant (SPGM, see section 5)

A: means that we *A*dd mixture components which approximate likelihood when all measurements are unlikely. (see section 4 and paper [2])

G: means that when likelihood has more than one but finite number of peaks we approximate likelihood as a *G*aussian mixture. (see section 4 and paper [2])

PF_N indicates particle filter with systematic resampling and N particles [5].

Table 2: A summary of 200 different simulations with typical suburban geometry. Simulations use a few base station range measurements, altitude measurements, some satellite pseudorange and delta range measurements.

Solver	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
EKF	1	183	368	36	2.8
UKF	2	183	361	24	1.5
$\mathrm{GMF}_{\mathrm{new}_2}$	4	172	332	41	1.6
$\mathrm{GMF}^{\mathrm{ukf}}_{\mathrm{S,A,G}}$	8	164	321	28	0.2
$\mathrm{GMF}^{\mathrm{ukf}}_{\mathrm{S}}$	7	163	325	18	0.5
$\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S}}$	2	160	327	26	1.0
PF_{10^3}	4	155	322	20	0.3
$\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$	4	153	305	34	0.4
PF_{10^4}	25	146	303	6	0.2
Ref	∞	146	307	0	0.1

These results are the realization of random variables and if we run these simulations again we possibly get a slightly different result. The following conclusions can be drawn based on simulations and theory.

- The new method of splitting Gaussian as Gaussian Mixture using sigma points when there is significant nonlinearity, (SPGM, see section 5) improves

results with every criterion (except time) regardless what updating method we use (EKF or UKF). SPGM-methods gives slightly better results than $\mathrm{GMF}_{\mathrm{new}_2}$.

- We get as good as or even better results when we use methods A and G proposed in [2].
- In this simulated case $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$ and PF_{10^3} give comparable results. Also the computation time is almost equal.
- We can say that $\mathrm{GMF}_{\mathrm{S,A,G}}$ is almost always consistent, because inconsistency is only $0.2\% - 0.4\%$ when the inconsistency of reference solutions is 0.1%.
- PF_{10^4} gives better results than $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$, but needs much more computation time.

6.2 Summary of suburban cases with sector and maximum range information

On Table 3, we have listed a summary of two hundred 120 second simulations, which use typical suburban geometry. Parameters and notations are the same than Table 2, but in these simulations every base station range measurements contains information of 120°-sector and maximum range. So we know that user is inside this particular area, which is restricted using sector and maximum range information. Here these sector and maximum range information (also called restrictive information) are used in the way proposed in paper [1].

Table 3: A summary of 200 different simulations with typical suburban geometry. Simulations use a few base station range measurements with sector and maximum range information, altitude measurements, some satellite pseudorange and delta range measurements.

Solver	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
UKF	2	147	296	22	0.8
EKF	1	145	293	28	1.4
$\mathrm{GMF}^{\mathrm{ukf}}_{\mathrm{S}}$	4	145	296	20	0.6
$\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S}}$	2	139	286	24	0.8
$\mathrm{GMF}^{\mathrm{ukf}}_{\mathrm{S,A,G}}$	5	137	291	24	0.3
$\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$	3	137	285	28	0.5
PF_{10^3}	3	134	281	20	0.4
PF_{10^4}	20	129	287	6	0.1
Ref	∞	129	290	0	0.1

The following conclusions can be drawn based on simulations with sector and maximum range information.

- Use sector and maximum range information improves SPGM-methods results a little, because there is only small nonlinearity and EKF and UKF do not have dramatic inconsistency problem.
- On Table 2, $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$ is the best GMF and here $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$ and $\mathrm{GMF}^{\mathrm{ukf}}_{\mathrm{S,A,G}}$ give comparable results. Because the UKF update method needs more computation than the EKF update method and results are almost comparable then it is better to use the EKF update method.
- Also in this case $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S,A,G}}$ and PF_{10^3} give comparable results and PF_{10^4} which gives a better result needs much more computation time.

6.3 Simulation with real data

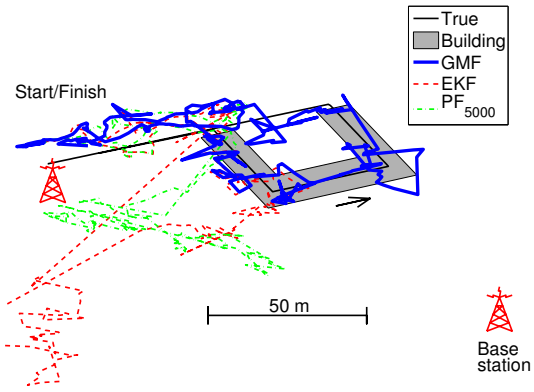

Figure 3: Simulation with real data uses on every time step one simulated base station range measurement, and additionally about $5 - 7$ satellite measurements when true route is outside the building. See also Table 4.

On Table 4 and Figure 3 we give results on our simulations which contains also real data. This test uses about $5 - 7$ satellite pseudorange and satellite delta range measurements when user is outside the building, measured using the iTrax03/16 GPS OEM receiver. This data was measured 16 August 2007 in campus area of the Tampere University of Technology by walking. We use also simulated base station range measurements, so that on every time step we have one base station range measurement. Active base station changes randomly between two possibilities so that the probability that current measurement comes from the same base station as on the previous time step is 0.75. We simulate the range measurement errors from $\epsilon_{\mathrm{b}} \sim \mathrm{N}(0, 10^2)$.

As we see in Figure 3, $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S}}$ is the only filter whose estimated track has approximately the same shape as the true track. $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S}}$ also gives much better statistics (except time) than EKF or PF_{5000}, see Table 4. There is approximately 37 percent of time when the effective sample size of PF_{5000} is smaller than 10, this means that almost all particles have approximately zero weight. In this case PF_{5000} reverts to EKF and after that re-initializes the samples from EKF posterior. Even though $\mathrm{GMF}^{\mathrm{ekf}}_{\mathrm{S}}$ gives good results

Table 4: Statistics of simulation with real data. See also Figure 3.

Solver	Time $\propto$	Err. rms	Err. 95%	Inc. %
EKF	1	55	143	15
PF_{5000}	28	42	71	22
$\mathrm{GMF}_{\mathrm{S}}^{\mathrm{ekf}}$	2	14	26	9

there is some inconsistency problem; we think that some of these inconsistency problems are caused by our model mismatch. Model mismatch is the important reason for the re-initializations of the particle filter. In fact, it seems that particle filter is more vulnerable to modeling errors than GMF.

7 CONCLUSIONS

In this article, we have studied Gaussian Mixture Filter for hybrid positioning application. We have presented a new method for splitting (approximating) a single Gaussian into a Gaussian Mixture so that we get better posterior approximations. Simulations show that GMF ($\mathrm{GMF}_{\mathrm{S,A,G}}^{\mathrm{ekf}}$ to be more specific) gives much better accuracy than Extended Kalman Filter and comparable results with PF_{1000}, with simulated data. With real data simulation $\mathrm{GMF}_{\mathrm{S}}^{\mathrm{ekf}}$ gives better results than PF_{5000}. In fact PF_{5000} works only partially with real data because there is quite often situation that almost all particles have zero weights. So there is a lot of work to improve particle filter performance with real GPS data. In real data cases we find it is important to build good models, because model mismatch causes maybe more inconsistency problems than filter approximations.

A BASICS OF GAUSSIAN MIXTURE

We call a n-dimensional random variable z a Gaussian mixture if it has the density function of form

$$p_z(z) = \sum_{j=1}^{N} \alpha_j \, \mathrm{N}_{\Sigma_j}^{\mu_j}(z). \tag{11}$$

The characteristic function of Gaussian $x \sim \mathrm{N}(\mu, \Sigma)$ is [13]

$$\phi_x(t) = \exp(it^T\mu - \frac{1}{2}t^T\Sigma t).$$

So the characteristic function of Gaussian mixture z (11) is

$$\begin{aligned}\phi_z(t) &= \sum_{j=0}^{N} \alpha_j \exp(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t) \\ &= \sum_{j=0}^{N} \alpha_j \phi_{x_j}(t).\end{aligned}$$

Mean of Gaussian mixture z (11) is

$$\begin{aligned}\mathrm{E}(z) &= \frac{1}{i}(\phi_z'(t)|_{t=0})^T \\ &= \frac{1}{i}\sum_{j=0}^{N} \alpha_j \phi_{x_j}(0)(i\mu_j - \Sigma_j 0) \\ &= \sum_{j=0}^{N} \alpha_j \mu_j \triangleq \mu_{\mathrm{gm}}.\end{aligned} \tag{12}$$

Covariance matrix of Gaussian mixture z (11) is

$$\begin{aligned}\mathrm{V}(z) &= -\phi_z''(t)|_{t=0} - \mathrm{E}(z)\,\mathrm{E}(z)^T \\ &= \sum_{j=0}^{N} \alpha_j \left(\Sigma_j + \mu_j\mu_j^T\right) - \mu_{\mathrm{gm}}\mu_{\mathrm{gm}}^T \\ &= \sum_{j=0}^{N} \alpha_j \left(\Sigma_j + (\mu_j - \mu_{\mathrm{gm}})(\mu_j - \mu_{\mathrm{gm}})^T\right) \\ &\triangleq \Sigma_{\mathrm{gm}}.\end{aligned} \tag{13}$$

The third moments $\mathrm{E}(z^i z^j z^k)$ of Gaussian mixture z (11) can be computed as follows

$$\begin{aligned}\mathrm{E}(zz^T z^k) &= -\frac{1}{i}\frac{\partial}{\partial t^k}\phi_z''(t)|_{t=0} \\ &= \sum_{j=0}^{N} \alpha_j \Big(\Sigma_j\mu_j^k + \mu_j\mu_j^T\mu_j^k \\ &\qquad + \Sigma_j e_k \mu_j^T + \mu_j e_k^T \Sigma_j\Big),\end{aligned} \tag{14}$$

where $k = 1, \ldots, n$, v^l is l. component of the vector v and e_k is vector whose k. component is one and others are zeros.

B SPLITTING A SINGLE GAUSSIAN INTO A GAUSSIAN MIXTURE

In this appendix, we show that the mean, covariance matrix and third moments of the approximation given in Table 1 are the same as those of the approximated distribution (7).

Mean of the approximation is (12)

$$\begin{aligned}\mu_{\mathrm{app}} &= \sum_{j=0}^{2n} \alpha_j \hat{x}_j \\ &= \frac{\kappa}{\kappa+n}\hat{x} + \sum_{j=1}^{n} \frac{1}{2(\kappa+n)}(\hat{x} + \tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_j) \\ &\quad + \sum_{j=n+1}^{2n} \frac{1}{2(\kappa+n)}(\hat{x} - \tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_{j-n}) \\ &= \hat{x} + \sum_{j=1}^{n} \frac{1}{2(\kappa+n)}\tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_j \\ &\quad - \sum_{l=1}^{n} \frac{1}{2(\kappa+n)}\tau\sqrt{\kappa+n}\sqrt{\mathrm{P}}e_l \\ &= \hat{x}.\end{aligned}$$

Covariance matrix of the approximation is (13)

$$\begin{aligned}\Sigma_{\text{app}} &= \sum_{j=0}^{2n} \alpha_j (\mathrm{P}_j + (\hat{x}_j - \mu_{\text{app}})(\hat{x}_j - \mu_{\text{app}})^T) \\ &= (1-\tau^2)\mathrm{P} + \sum_{j=0}^{2n} \alpha_j (\hat{x}_j - \hat{x})(\hat{x}_j - \hat{x})^T \\ &= (1-\tau^2)\mathrm{P} + \tau^2 \sum_{j=1}^{n} (\sqrt{\mathrm{P}} e_j)(\sqrt{\mathrm{P}} e_j)^T \\ &= (1-\tau^2)\mathrm{P} + \tau^2 \sum_{j=1}^{n} \sqrt{\mathrm{P}} e_j e_j^T \sqrt{\mathrm{P}}^T \\ &= \mathrm{P}.\end{aligned}$$

The third moments of the Gaussian (7) are (14)

$$\mathrm{E}(xx^T x^k) = \mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P} + \hat{x}\hat{x}^T\hat{x}^k.$$

Third moments of the approximation are (14)

$$\begin{aligned}&\mathrm{E}(x_{\text{app}} x_{\text{app}}^T x_{\text{app}}^k) \\ &= \sum_{j=0}^{2n} \alpha_j \Big(\mathrm{P}_j \hat{x}_j^k + \hat{x}_j \hat{x}_j^T \hat{x}_j^k + \mathrm{P}_j e_k \hat{x}_j^T + \hat{x}_j e_k^T \mathrm{P}_j \Big) \\ &= (1-\tau^2)\left(\mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P}\right) + \alpha_0 \hat{x}\hat{x}^T\hat{x}^k \\ &\quad + \sum_{j=1}^{n} \alpha_j (\hat{x} + \mathrm{A}e_j)(\hat{x} + \mathrm{A}e_j)^T e_k^T (\hat{x} + \mathrm{A}e_j) \\ &\quad + \sum_{j=n}^{n} \alpha_j (\hat{x} - \mathrm{A}e_j)(\hat{x} - \mathrm{A}e_j)^T e_k^T (\hat{x} - \mathrm{A}e_j) \\ &= (1-\tau^2)\left(\mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P}\right) + \hat{x}\hat{x}^T\hat{x}^k \\ &\quad + 2\sum_{j=1}^{n} \alpha_j \Big(\mathrm{A}e_j e_j^T \mathrm{A}^T \hat{x}^k \\ &\quad + \hat{x}e_j^T \mathrm{A}^T e_k^T \mathrm{A}e_j + \mathrm{A}e_j \hat{x}^T e_k^T \mathrm{A}e_j \Big) \\ &= (1-\tau^2)\left(\mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P}\right) + \hat{x}\hat{x}^T\hat{x}^k \\ &\quad + \sum_{j=1}^{n} \tau^2 \Big(\sqrt{\mathrm{P}} e_j e_j^T \sqrt{\mathrm{P}}^T \hat{x}^k \\ &\quad + \hat{x}e_k^T \sqrt{\mathrm{P}} e_j e_j^T \sqrt{\mathrm{P}}^T + \sqrt{\mathrm{P}} e_j e_j^T \sqrt{\mathrm{P}}^T e_k \hat{x}^T \Big) \\ &= (1-\tau^2)\left(\mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P}\right) + \hat{x}\hat{x}^T\hat{x}^k \\ &\quad + \tau^2(\mathrm{P}\hat{x}^k + \hat{x}e_k^T\mathrm{P} + \mathrm{P}e_k\hat{x}^T) \\ &= \mathrm{P}\hat{x}^k + \mathrm{P}e_k\hat{x}^T + \hat{x}e_k^T\mathrm{P} + \hat{x}\hat{x}^T\hat{x}^k \\ &= \mathrm{E}(xx^T x^k).\end{aligned}$$

where $\mathrm{A} = \tau\sqrt{\kappa + n}\sqrt{\mathrm{P}}$.

ACKNOWLEDGMENTS

This study was partly funded by Nokia Corporation. Simo Ali-Löytty acknowledges the financial support of the Nokia Foundation and the Tampere Graduate School in Information Science and Engineering. The authors would like to thank Ville Honkavirta at Institute of Mathematics, Tampere University of Technology, for collecting the GPS measurements.

REFERENCES

[1] S. Ali-Löytty and N. Sirola. A modified Kalman filter for hybrid positioning. In *Proceedings of ION GNSS 2006*, September 2006.

[2] S. Ali-Löytty and N. Sirola. Gaussian mixture filter in hybrid navigation. In *Proceedings of The European Navigation Conference GNSS 2007*, 2007 (to be published).

[3] S. Ali-Löytty, N. Sirola, and R. Piché. Consistency of three Kalman filter extensions in hybrid navigation. In *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, July 2005.

[4] D. L. Alspach and H. W. Sorenson. Nonlinear bayesian estimation using gaussian sum approximations. *IEEE Transactions on Automatic Control*, 17(4):439–448, Aug 1972.

[5] 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, 2002.

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

[7] R. G. Brown. *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, 1983.

[8] R. S. Bucy and K. D. Senne. Digital synthesis of nonlinear filters. *Automatica*, 7(3):287–298, May 1971.

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

[10] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.

[11] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *American Control Conference*, volume 3, pages 1628–1632, 1995.

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

PUBLICATION 5

Simo Ali-Löytty: Efficient Gaussian mixture filter for hybrid positioning. In *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS 2008)*, Monterey CA, May 2008, pages 60–66.

Efficient Gaussian Mixture Filter for Hybrid Positioning

Simo Ali-Löytty
Department of Mathematics
Tampere University of Technology
Finland

***Abstract*—This paper presents a new way to apply Gaussian Mixture Filter (GMF) to hybrid positioning. The idea of this new GMF (Efficient Gaussian Mixture Filter, EGMF) is to split the state space into pieces using parallel planes and approximate posterior in every piece as Gaussian. EGMF outperforms the traditional single-component positioning filters, for example the Extended Kalman Filter and the Unscented Kalman Filter, in nonlinear hybrid positioning. Furthermore, EGMF has some advantages with respect to other GMF variants, for example EGMF gives the same or better performance than the Sigma Point Gaussian Mixture (SPGM) [1] with a smaller number of mixture components, i.e. smaller computational and memory requirements. If we consider only one time step, EGMF gives optimal results in the linear case, in the sense of mean and covariance, whereas other GMFs gives suboptimal results.**

I. Introduction

Positioning filters, such as GMF [2]–[4], are used to compute an estimate of the state using current and past measurement data. Usually, the mean of the posterior distribution is this estimate. A consistent filter also provides correct information on the accuracy of its state estimate, e.g. in the form of an estimated error covariance. Generally, GMF is a filter whose approximate prior and posterior densities are Gaussian Mixtures (GMs), a linear combination of Gaussian densities where weights are between 0 and 1. GMF is an extension of Kalman type filters. In particular, The Extended Kalman Filter (EKF) [5]–[8], Second Order Extended Kalman Filter (EKF2) [5], [6], [9], Unscented Kalman Filters (UKF) [10] and a bank of these filters, are special cases of GMF.

Hybrid positioning means that measurements used in positioning come from many different sources e.g. Global Navigation Satellite System, Inertial Measurement Unit, or local wireless networks such as a cellular network. Range, pseudo-range, delta range, altitude, base station sector and compass measurements are examples of typical measurements in hybrid positioning. In the hybrid positioning case, it is usual that measurements are nonlinear and because of that posterior density may have multiple peaks (multiple positioning solutions). In these cases, traditional single-component positioning filters, such as EKF, do not give good performance [9]. This is the reason for developing GMF for hybrid positioning [1]. Other possibility is to use a general nonlinear Bayesian filter, which is usually implemented as a particle filter or a point mass filter. These filters usually work correctly and give good positioning accuracy but require much computation time and memory.

An outline of the paper is as follows. In Section II, we glance at Bayesian filtering. In Section III, we study the basics of the GM and of the GMF. In Section IV, we present the new method, box GM approximation, to approximate Gaussian as GM. In Section V we apply the box GM approximation to the filtering framework and get the Box Gaussian Mixture Filter (BGMF). In Section V we also present Sigma Point Gaussian Mixture Filter (SPGMF) [1]. In Section VI, we develop BGMF so that it gives exact mean and covariance in one step linear case. We call that new filter the Efficient Gaussian Mixture Filter (EGMF). In Section VII, we compute one step comparison of EKF, SPGMF, BGMF and EGMF. Finally in Section VIII, we present simulation results where we compare different GMFs and a bootstrap particle filter [11].

II. Bayesian Filtering

We consider the discrete-time non-linear non-Gaussian system

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \tag{1}$$

$$y_k = h_k(x_k) + v_k, \tag{2}$$

where the vectors $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_{y_k}}$ represent the state of the system and the measurement at time t_k, $k \in \mathbb{N}$, respectively. We assume that errors w_k and v_k are white, mutually independent and independent of the initial state x_0. We denote the density functions of w_k and v_k by p_{w_k} and p_{v_k}, respectively. The aim of filtering is to find the conditional probability density function (posterior)

$$p(x_k|y_{1:k}),$$

where $y_{1:k} \stackrel{\triangle}{=} y_1, \ldots, y_k$ are past and current measurements. The posterior can be determined recursively according to the following relations.

Prediction (prior):

$$p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}; \tag{3}$$

Update (posterior):

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k}, \tag{4}$$

where the transition pdf is

$$p(x_k|x_{k-1}) = p_{w_{k-1}}(x_k - f_{k-1}(x_{k-1}))$$

and the likelihood

$$p(y_k|x_k) = p_{v_k}(y_k - h_k(x_k)).$$

The initial condition for the recursion is given by the pdf of the initial state $p(x_0|y_{1:0}) = p(x_0)$. Knowledge of the posterior distribution (4) enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error (MMSE) estimate is the conditional mean of x_k [5], [12]. In general and in our case, the conditional probability density function cannot be determined analytically.

III. Gaussian Mixture Filter

A. Gaussian Mixture

Definition 1 (Gaussian Mixture): an n-dimensional random variable x is N-component Gaussian Mixture (GM) if its characteristic function has the form

$$\varphi_x(t) = \sum_{j=1}^{N} \alpha_j \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right), \tag{5}$$

where $\mu_j \in \mathbb{R}^n$, $\Sigma_j \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite ($\Sigma_j \geq 0$), $\alpha_j \geq 0$ and $\sum_{j=1}^{N}\alpha_j = 1$. We use the abbreviation $x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}$.

If random variable x is GM, $x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}$ and all matrices Σ_j are symmetric positive definite ($\Sigma_j > 0$), then x has a density function

$$p_x(\xi) = \sum_{j=1}^{N} \alpha_j \,\mathrm{N}_{\Sigma_j}^{\mu_j}(x), \tag{6}$$

where $\mathrm{N}_{\Sigma_j}^{\mu_j}(x)$ is the Gaussian density function with mean μ_j and covariance Σ_j

$$\mathrm{N}_{\Sigma_j}^{\mu_j}(x) = \frac{1}{(2\pi)^{\frac{n}{2}}\sqrt{\det\Sigma_j}} e^{-\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)}.$$

Theorem 2: Let $x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}$. Then the mean of x is

$$\mathrm{E}(x) = \sum_{j=1}^{N} \alpha_j\mu_j,$$

and the covariance of x is

$$\mathrm{V}(x) = \sum_{j=1}^{N} \alpha_j \left(\Sigma_j + (\mu_j - \mathrm{E}(x))(\mu_j - \mathrm{E}(x))^T\right).$$

Proof: Using the properties of characteristic function [13], we get

$$\mathrm{E}(x) = \frac{1}{i}\left(\varphi_x'(t)|_{t=0}\right)^T = \sum_{j=1}^{N}\alpha_j\mu_j$$

and

$$\begin{aligned}\mathrm{V}(x) &= -\varphi_x''(t)|_{t=0} - \mathrm{E}(x)\,\mathrm{E}(x)^T \\ &= \sum_{j=1}^{N}\alpha_j(\mu_j\mu_j^T + \Sigma_j) - \mathrm{E}(x)\,\mathrm{E}(x)^T \\ &= \sum_{j=1}^{N}\alpha_j\left(\Sigma_j + (\mu_j - \mathrm{E}(x))(\mu_j - \mathrm{E}(x))^T\right).\end{aligned}$$

■

Theorem 3: Let random variables $x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}$ and $v \sim \mathrm{N}(0, \mathrm{R})$ be independent. Define $y = \mathrm{H}x + v$, where $\mathrm{H} \in \mathbb{R}^{m\times n}$. Then

$$y \sim \mathrm{M}(\alpha_j, \mathrm{H}\mu_j, \mathrm{H}\Sigma_j\mathrm{H}^T + \mathrm{R})_{(j)}.$$

Proof: Because x and v are independent then $\mathrm{H}x$ and v are also independent. Futhermore,

$$\begin{aligned}\varphi_{\mathrm{H}x+v}(t) &\overset{Ind.}{=} \varphi_{\mathrm{H}x}(t)\varphi_v(t) = \varphi_x(\mathrm{H}^Tt)\varphi_v(t) \\ &\overset{(5)}{=} \sum_{j=1}^{N}\alpha_i \exp\left(it^T\mathrm{H}\mu_j - \frac{1}{2}t^T\mathrm{H}\Sigma_j\mathrm{H}^Tt\right)\exp\left(-\frac{1}{2}t^T\mathrm{R}t\right) \\ &= \sum_{j=1}^{N}\alpha_j\exp\left(it^T\mathrm{H}\mu_j - \frac{1}{2}t^T\left(\mathrm{H}\Sigma_j\mathrm{H}^T + \mathrm{R}\right)t\right).\end{aligned}$$

■

B. Gaussian Mixture Filter

GMF is (an approximation of) the Bayesian Filter (see Section II). The idea of GMF [2]–[4] (also called Gaussian Sum Filter) is that both prior density (3) and posterior density (4) are GMs.

Algorithm 1 Linearized GMF

Initial state at time t_0: $x_0 \sim \mathrm{M}(\alpha_{j,0}^+, x_{j,0}^+, \mathrm{P}_{j,0}^+)_{(j)}$
for $k = 1$ to n **do**

1) Prediction step, prior at time t_k (see Thm. 3):

$$\mathrm{M}(\alpha_{j,k}^-, x_{j,k}^-, \mathrm{P}_{j,k}^-)_{(j)},$$

where

$$\begin{aligned}\alpha_{j,k}^- &= \alpha_{j,k-1}^+ \\ x_{j,k}^- &= \mathrm{F}_{k-1}x_{j,k-1}^+ \\ \mathrm{P}_{j,k}^- &= \mathrm{F}_{k-1}\mathrm{P}_{j,k-1}^+{\mathrm{F}_{k-1}}^T + \mathrm{Q}_{k-1}\end{aligned}$$

2) Approximate selected components as GM (see Section IV)
3) Update step, posterior at time t_k [14]:

$$\mathrm{M}(\alpha_{j,k}^+, x_{j,k}^+, \mathrm{P}_{j,k}^+)_{(j)},$$

where

$$\begin{aligned}\alpha_{j,k}^+ &\propto \alpha_{j,k}^- \,\mathrm{N}_{\mathrm{H}_{j,k}\mathrm{P}_{j,k}^-\mathrm{H}_{j,k}^T+\mathrm{R}_k}^{h_k(\bar{x}_{j,k})+\mathrm{H}_{j,k}(x_{j,k}^- - \bar{x}_{j,k})}(y_k) \\ x_{j,k}^+ &= x_{j,k}^- + \mathrm{K}_{j,k}(y_k - h_k(\bar{x}_{j,k}) - \mathrm{H}_{j,k}(x_{j,k}^- - \bar{x}_{j,k})) \\ \mathrm{P}_{j,k}^+ &= (\mathrm{I} - \mathrm{K}_{j,k}\mathrm{H}_{j,k})\mathrm{P}_{j,k}^- \\ \mathrm{K}_{j,k} &= \mathrm{P}_{j,k}^-\mathrm{H}_{j,k}^T(\mathrm{H}_{j,k}\mathrm{P}_{j,k}^-\mathrm{H}_{j,k}^T + \mathrm{R}_k)^{-1}\end{aligned}$$

Here $\mathrm{H}_{j,k} = \frac{\partial h_k(x)}{\partial x}|_{x=\bar{x}_{j,k}}$ and $\bar{x}_{j,k}$ are selected linearization points, e.g. in EKF $\bar{x}_{j,k} = x_{j,k}^-$.
4) Reduce number of components: forgetting, merging and resampling [2], [15], [16].

end for

Algorithm 1 presents one version of GMF, Linearized GMF. Algorithm 1 uses the following assumptions:

1) Initial state x_0 is non-singular GM, which means that x_0 has density function (6).
2) State model (1) is linear

$$x_k = \mathrm{F}_{k-1}x_{k-1} + w_{k-1}.$$

3) Errors $w_k \sim \mathrm{N}(0, \mathrm{Q}_k)$ and $v_k \sim \mathrm{N}(0, \mathrm{R}_k)$ and R_k is non-singular.

Note that it is straightforward to extend Linearized GMF to cases where also the errors w_k and v_k are GMs.

IV. Approximate Gaussian as Gaussian Mixture

As we know, EKF in hybrid positioning [9] has a consistency problem. The key reason for inconsistency is nonlinearity. Now we assume that we have only one measurement and our prior is Gaussian $x \sim \mathrm{N}(\hat{x}, \mathrm{P})$. The application to the general case is given in Section V.

One measure of nonlinearity is [1], [6], [9]

$$Nonlinearity = \sqrt{\frac{\mathrm{tr}(\mathrm{H}_e\mathrm{P}\mathrm{H}_e\mathrm{P})}{\mathrm{R}}} - 1, \tag{7}$$

where H_e is Hessian matrix of scalar measurement $h(x)$, R is a covariance of measurement error and P is a covariance of the state component. One possibility to overcome the nonlinearity problem (i.e. minimize $Nonlinearity$ (7)) is to approximate Gaussian as GM whose components have smaller covariance matrices than the orginal Gaussian. One method for doing so is the Sigma Point Gaussian Mixture (SPGM) [1] (see Section IV-A). One drawback of SPGM is that SPGM splits one Gaussian to $2n_x + 1$ components GM, regardless of our measurement equation and Hessian matrix H_e. Because of this, we present a new method of approximating Gaussian as GM (see Section IV-B). We call this method as a Box GM, because it has connection to the "Box"-method [17].

A. Sigma Point GM approximation

The SPGM is given on Table I. SPGM has the same mean, covariance, and third moments as the original Gaussian distribution $x \sim \mathrm{N}(\hat{x}, \mathrm{P})$ [1].

TABLE I
SPGM $\mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}$ APPROXIMATION OF GAUSSIAN $x \sim \mathrm{N}(\hat{x}, \mathrm{P})$, PARAMETERS $\tau \in (0, 1)$ AND $\kappa > 0$.

Index j	α_j	μ_j	Σ_j
0	$\frac{\kappa}{\kappa+n_x}$	$\hat{x}$	$(1-\tau^2)\mathrm{P}$
$1, \ldots, n_x$	$\frac{1}{2(\kappa+n_x)}$	$\hat{x} + \tau\sqrt{\kappa+n_x}\sqrt{\mathrm{P}}e_j$	$(1-\tau^2)\mathrm{P}$
$n_x+1, \ldots, 2n_x$	$\frac{1}{2(\kappa+n_x)}$	$\hat{x} - \tau\sqrt{\kappa+n_x}\sqrt{\mathrm{P}}e_{j-n}$	$(1-\tau^2)\mathrm{P}$

B. Box GM approximation

The idea of the Box GM approximation is that we split the state space using parallel planes and approximate the Gaussian inside every piece with one GM component using moment matching method. The Box GM approximation of the Gaussian $x \sim \mathrm{N}(\hat{x}, \mathrm{P})$, with $\mathrm{P} > 0$, is

$$x_N \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j)}, \tag{8}$$

where index $j = 1, \ldots, n_{\mathrm{box}}$, and parameters are

$$\alpha_j = \int_{A_j} p_x(\xi)d\xi = \Phi\left(l_j\right) - \Phi\left(l_{j-1}\right),$$

$$\mu_j = \int_{A_j} \xi\frac{p_x(\xi)}{\alpha_j}d\xi = \hat{x} + \mathrm{P}a\epsilon_j, \epsilon_j = \frac{e^{-\frac{l_{j-1}^2}{2}} - e^{-\frac{l_j^2}{2}}}{\sqrt{2\pi}\alpha_j},$$

$$\Sigma_j = \int_{A_j} (\xi-\mu_j)(\xi-\mu_j)^T\frac{p_x(\xi)}{\alpha_j}d\xi$$

$$= \mathrm{P} - \mathrm{P}a\left(\frac{l_j e^{-\frac{l_j^2}{2}} - l_{j-1}e^{-\frac{l_{j-1}^2}{2}}}{\sqrt{2\pi}\alpha_j} + \epsilon_j^2\right)a^T\mathrm{P}.$$

where Φ is the standard normal cumulative density function and sets A_j have the following form

$$A_j = \left\{x | l_j < a^T(x-\hat{x}) \le l_{j+1}\right\},$$

where $a^T\mathrm{P}a = 1$ and vector l is monotonic increasing so that $l_1 = -\infty$ and $l_{n_{\mathrm{box}}+1} = \infty$ so these sets constitute a partition of $\mathbb{R}^{n_x}$.

1) Mean and covariance of the Box GM approximation: In this Section, we compute the mean and the covariance of the GM approximation x_N Eq. (8). First of all, because $\Sigma_j > 0\,\forall j$, $\alpha_j > 0\,\forall j$ and

$$\sum_{j=1}^{n_{\mathrm{box}}}\alpha_j = \sum_{j=1}^{n_{\mathrm{box}}}\int_{A_j} p_x(\xi)d\xi = \int p_x(\xi)d\xi = 1,$$

then x_N is a valid GM. The mean of x_N is

$$\mathrm{E}\left(x_N\right) \overset{Thm.\ 2}{=} \sum_{j=1}^{n_{\mathrm{box}}}\alpha_j\mu_j = \sum_{j=1}^{n_{\mathrm{box}}}\int_{A_j}\xi p_x(\xi)d\xi = \hat{x}.$$

The covariance of x_N is

$$\mathrm{V}\left(x_N\right) \overset{Thm.\ 2}{=} \sum_{j=1}^{n_{\mathrm{box}}}\alpha_j(\mu_j\mu_j^T + \Sigma_j) - \hat{x}\hat{x}^T$$

$$= \sum_{j=1}^{n_{\mathrm{box}}}\int_{A_j}\xi\xi^T p_x(\xi)d\xi - \hat{x}\hat{x}^T = \mathrm{P}.$$

So the mean and the covariance of the GM approximation are the same as the mean and the covariance of the ordinary Gaussian. Because of this, we can say that Box GM approximation is a moment matching method.

2) Contour plot of the Box GM approximation: In Fig. 1 we compare the density function of the Gaussian distribution

$$x \sim \mathrm{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 13 & -12 \\ -12 & 13 \end{bmatrix}\right). \tag{9}$$

and the density function of its approximation by a Box GM with parameters $a \approx \begin{bmatrix} 0.2774 \\ 0 \end{bmatrix}$ and

$$l = \begin{bmatrix} -\infty & -1.28 & 1.28 & \infty \end{bmatrix}$$
$$\approx \Phi^{-1}\left(\begin{bmatrix} 0 & 0.1 & 0.9 & 1 \end{bmatrix}\right).$$

Fig. 1 shows the contour plots of the Gaussian and the Box GM density functions so that 50% of probability is inside the

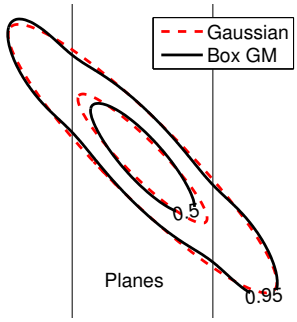

Fig. 1. Contour plot example of the Box GM, when approximating density is Gaussian Eq. (9)

innermost curve and 95% of probability is inside the outermost curve. We see that the density function of the Box GM is quite good approximation of the Gaussian density function Eq. (9).

V. Box Gaussian Mixture Filter and Sigma Point Gaussian Mixture Filter

Box Gaussian Mixture Filter (BGMF) is a straightforward application of the Box GM approximation to the Gaussian Mixture filtering framework. BGMF is a Linearized GMF (see Alg. 1), where the step 2 is the following. First we compute $Nonlinearity$ (7) statistics for every component and measurement. We select the components that have at least one highly nonlinear measurement with $Nonlinearity > 0$. Every selected component we replace by its Box GM approximation (Section IV-B).

The SPGMF, which is presented in the paper [1] with name $\text{GMF}_{\text{S}}^{\text{ekf}}$, is also a Linearized GMF (see Alg. 1) and almost the same as BGMF. Only difference between SPGMF and BGMF is that SPGM approximates selected Gaussians using SPGM approximation (see Table I), while BGMF uses the Box GM approximation (Section IV-B).

VI. Efficient Gaussian Mixture Filter

In this Section, we derive a new GMF, Efficient Gaussian Mixture Filter (EGMF). Prediction step (Eq. (3)) of EGMF is the same as prediction step of Linearized GMF (see Alg. 1). Now we consider update step (Eq. (4)). Assume that prior distribution is a GM

$$x \sim \mathrm{M}(\beta^i, \hat{x}^i, \mathrm{P}^i)_{(i)}$$

and measurement model is (see (2))

$$y = h(x) + v,$$

where $v \sim \mathrm{N}(0, \mathrm{R})$. Now the posterior density function is

$$\begin{aligned} p(x|y) &\propto \sum_{i=1}^{n_{\text{prior}}} \beta^i \, \mathrm{N}_{\mathrm{P}^i}^{\hat{x}^i}(x) \, \mathrm{N}_{\mathrm{R}}^{0}(y - h(x)) \\ &= \sum_{i=1}^{n_{\text{prior}}} \beta^i \sum_{j=1}^{n_i} \chi_{A_j^i}(x) \, \mathrm{N}_{\mathrm{P}^i}^{\hat{x}^i}(x) \, \mathrm{N}_{\mathrm{R}}^{0}\left(y - h(x)\right), \end{aligned}$$

where ${a^i}^T \mathrm{P}^i a^i = 1 \ \forall i$, $A_j^i = \left\{ x | l_j^i < {a^i}^T (x - \hat{x}^i) \le l_{j+1}^i \right\}$, vectors l^i are monotonic increasing so that $l_1^i = -\infty$ and $l_{n_i+1}^i = \infty$. So for all i sets A_j^i constitute a partition of $\mathbb{R}^{n_x}$. BGMF (see Section V) approximates

$$\begin{aligned} &\chi_{A_j^i}(x) \, \mathrm{N}_{\mathrm{P}^i}^{\hat{x}^i}(x) \approx \alpha_j^i \, \mathrm{N}_{\Sigma_j^i}^{\mu_j^i}(x) \ \text{ and} \\ &\mathrm{N}_{\mathrm{R}}^{0}\left(y - h(x)\right) \approx \mathrm{N}_{\mathrm{R}}^{0}\left(y - h(\mu_j^i) - \mathrm{H}_{\mu_j^i}(x - \mu_j^i)\right) \end{aligned}$$

where α_j^i, μ_j^i and Σ_j^i are computed using the Box GM algorithm (see Section IV-B) and $\mathrm{H}_{\mu_j^i} = h'(\mu_j^i)$. So BGMF approximates both prior and likelihood before multiplying them. EGMF first approximates the likelihood as

$$\mathrm{N}_{\mathrm{R}}^{0}\left(y - h(x)\right) \approx \mathrm{N}_{\mathrm{R}}^{0}\left(y - h(\mu_j^i) - \mathrm{H}_{\mu_j^i}(x - \mu_j^i)\right)$$

and then multiplies it with the prior:

$$\begin{aligned} &p(x|y) \propto \\ &\overset{\star}{\approx} \sum_{i=1}^{n_{\text{prior}}} \beta^i \sum_{j=1}^{n_i} \chi_{A_j^i}(x) \, \mathrm{N}_{\mathrm{P}^i}^{\hat{x}^i}(x) \, \mathrm{N}_{\mathrm{R}}^{0}\left(y - h(\mu_j^i) - \mathrm{H}_{\mu_j^i}(x - \mu_j^i)\right) \\ &= \sum_{i=1}^{n_{\text{prior}}} \beta^i \sum_{j=1}^{n_i} \chi_{A_j^i}(x) \gamma_j^i \, \mathrm{N}_{\mathrm{P}_j^i}^{\hat{x}_j^i}(x) \end{aligned} \tag{10}$$

where

$$\begin{aligned} \gamma_j^i &= \mathrm{N}_{\mathrm{H}_{\mu_j^i} \mathrm{P}^i \mathrm{H}_{\mu_j^i}^T + \mathrm{R}}^{h(\mu_j^i) + \mathrm{H}_{\mu_j^i}(\hat{x}^i - \mu_j^i)}(y) \\ \hat{x}_j^i &= \hat{x}^i + \mathrm{K}_j^i (y - h(\mu_j^i) - \mathrm{H}_{\mu_j^i}(\hat{x}^i - \mu_j^i)) \\ \mathrm{P}_j^i &= (\mathrm{I} - \mathrm{K}_j^i \mathrm{H}_{\mu_j^i}) \mathrm{P}^i \\ \mathrm{K}_j^i &= \mathrm{P}^i \mathrm{H}_{\mu_j^i}^T (\mathrm{H}_{\mu_j^i} \mathrm{P}^i \mathrm{H}_{\mu_j^i}^T + \mathrm{R})^{-1}. \end{aligned}$$

Note that $\star$ approximation is exact if $h(x)$ is linear. Then we use the Box GM algorithm (see Section IV-B) and get

$$\begin{aligned} p(x|y) \propto \overset{\star}{\approx} &\sum_{i=1}^{n_{\text{prior}}} \beta^i \sum_{j=1}^{n_i} \chi_{A_j^i}(x) \gamma_j^i \, \mathrm{N}_{\mathrm{P}_j^i}^{\hat{x}_j^i}(x) \\ = &\sum_{i=1}^{n_{\text{prior}}} \sum_{j=1}^{n_i} \beta^i \gamma_j^i \chi_{B_j^i}(x) \, \mathrm{N}_{\mathrm{P}_j^i}^{\hat{x}_j^i}(x) \\ \approx &\sum_{i=1}^{n_{\text{prior}}} \sum_{j=1}^{n_i} \bar{\alpha}_j^i \beta^i \gamma_j^i \, \mathrm{N}_{\bar{\Sigma}_j^i}^{\bar{\mu}_j^i}(x) \end{aligned} \tag{11}$$

where $\bar{\alpha}_j^i$, $\bar{\mu}_j^i$ and $\bar{\Sigma}_j^i$ are computed using the Box GM algorithm, when we noted that

$$\begin{aligned} B_j^i &= \left\{ x | m_j^i < {b^i}^T (x - \hat{x}_j^i) \le m_{j+1}^i \right\} \\ &= \left\{ x | l_j^i < {a^i}^T (x - \hat{x}^i) \le l_{j+1}^i \right\} = A_j^i, \end{aligned}$$

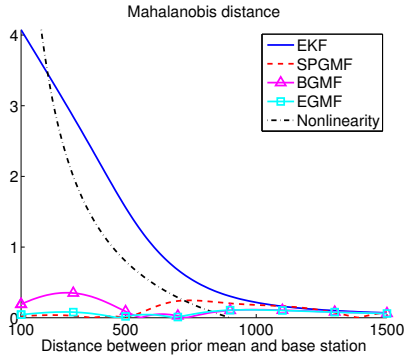

Fig. 2. Mahalanobis distance between exact mean and mean approximations. Significance of nonlinearity (see equation (7)) is also shown.

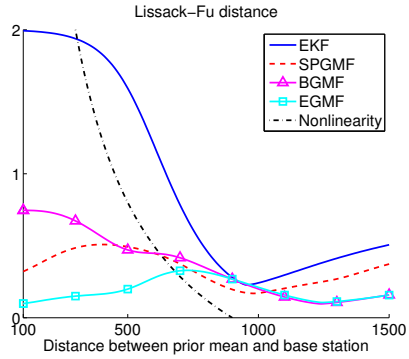

Fig. 3. Lissack-Fu distance between exact posterior and posterior approximations. Significance of nonlinearity (see equation (7)) is also shown.

where

$$b^i = \frac{a^i}{\sqrt{a^{i^T} \mathrm{P}_j^i a^i}} \Rightarrow b^{i^T} \mathrm{P}_j^i b^i = 1$$
$$m^i = \frac{l^i + a^{i^T}(\hat{x}^i - \hat{x}_j^i)}{\sqrt{a^{i^T} \mathrm{P}_j^i a^i}} \Rightarrow m_j^i < m_{j+1}^i \, \forall j \,,$$
$$m_1^i = -\infty \text{ and } m_{n_i}^i = \infty.$$

Now we get that the posterior of EGMF is

$$p(x|y) \approx \frac{\sum_{i=1}^{n_{\text{prior}}} \sum_{j=1}^{n_i} \bar{\alpha}_j^i \beta^i \gamma_j^i \, \mathrm{N}_{\bar{\Sigma}_j^i}^{\bar{\mu}_j^i}(x)}{\sum_{i=1}^{n_{\text{prior}}} \sum_{j=1}^{n_i} \bar{\alpha}_j^i \beta^i \gamma_j^i}.$$

If we consider the linear case and only one time step, EGMF gives a correct mean and a correct covariance, because linearity ensures that $\star$ approximation in Eq. (10) is exact and the Box GM approximation maintains the mean and the covariance in Eq. (11) (see Section IV-B1). SPGMF and BGMF can give a wrong mean and a wrong covariance in one time step linear case.

VII. One step comparison of EKF, SPGMF, BGMF and EGMF

This is the same example as in paper [1], but we have included BGMF (Section V) and EGMF (Section VI) to this example. This example presents a comparison between EKF posterior, SPGMF (Section V) posterior, BGMF posterior and EGMF posterior, when prior distribution is

$$x \sim \mathrm{N}\left(\begin{bmatrix} d \\ 0 \end{bmatrix}, \begin{bmatrix} 100^2 & 0 \\ 0 & 300^2 \end{bmatrix} \right),$$

and we get one base station range measurement (2), where

$$y = 1000, \; h(x) = \|x\| \text{ and } v \sim \mathrm{N}(0, 100^2).$$

The base station is located in origin and so d is the distance between prior mean and the base station. "Exact" posterior density function $p_{\text{exact}}(x)$ is computed using a point-mass filter, with 500^2 points [18]. Approximation of the exact density function is computed using EKF, SPGMF, BGMF and EGMF. The SPGMF uses parameters $\kappa = 4$ and $\tau = \frac{1}{2}$ and the BGMF and EGMF uses parameters $a = \begin{bmatrix} 0 \\ \frac{1}{300} \end{bmatrix}$ and

$$l = \begin{bmatrix} -\infty & -1.28 & 1.28 & \infty \end{bmatrix}$$
$$\approx \Phi^{-1}\left(\begin{bmatrix} 0 & 0.1 & 0.9 & 1 \end{bmatrix} \right).$$

Note that both BGMF and EGMF have only three GM components whereas SPGMF has five GM components.

Comparison between EKF, SPGMF, BGMF and EGMF contains two parts. First we compute the Mahalanobis distance between the mean of the exact posterior mean and means of approximations

$$\sqrt{(\mu_{\text{exact}} - \mu_{\text{app}})^T \Sigma_{\text{exact}}^{-1} (\mu_{\text{exact}} - \mu_{\text{app}})}.$$

These results are shown in Fig. 2. The value of the *Nonlinearity* function (7) is also plotted in Fig. 2. We see that Mahalanobis distance between EKF mean and exact mean increases rapidly when nonlinearity becomes more significant. In that case SPGMF, BGMF and EGMF give much better results than EKF. EGMF has the same or smaller Mahalanobis distance than BGMF. Furthermore, SPGMF, BGMF and EGMF give always as good results as EKF even when there is no significant nonlinearity.

Second, we compute the first order Lissack-Fu distance

$$\int |p_{\text{exact}}(x) - p_{\text{app}}(x)| dx,$$

between exact posterior and the approximations (also called a total variation norm). These results are in Fig. 3. The value of the *Nonlinearity* function (7) is also plotted. We see that SPGMF, BGMF and EGMF give smaller Lissack-Fu distance than EKF. Difference between EGMF Lissack-Fu distance and EKF Lissack-Fu distance increases when nonlinearity becomes more significant. EKF Lissack-Fu distance is almost 2 (maximum value) when $d = 100$, so the exact posterior and the EKF posterior approximation are almost totally separate.

Furthermore, EGMF has Lissack-Fu distance same as or smaller than BGMF.

Overall SPGMF, BGMF and EGMF work almost identically although EGMF and BGMF use only three mixture component versus five SPGMF mixture components. In this example, EGMF gives the best results compared to the other filters. Furthermore SPGMF, BGMF and EGMF all give much better results than EKF when nonlinearity is significant.

VIII. Simulations

In the simulations, we use the position-velocity model, so the state $x = \begin{bmatrix} r_{\mathrm{u}} \\ v_{\mathrm{u}} \end{bmatrix}$ consists of user position vector r_{u} and user velocity vector v_{u}, which are in East-North-Up (ENU) coordinate system. In this model the user velocity is a random walk process [19]. Now the state-dynamic (1) is

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1},$$

where

$$\Phi_{k-1} = \begin{bmatrix} \mathrm{I} & \Delta t_k \mathrm{I} \\ 0 & \mathrm{I} \end{bmatrix},$$

$\Delta t_k = t_k - t_{k-1}$, and w_{k-1} is white, zero mean and Gaussian noise, with covariance

$$\mathrm{Q}_{k-1} = \begin{bmatrix} \frac{\Delta t_k^3 \sigma_{\mathrm{p}}^2}{3}\mathrm{I} & 0 & \frac{\Delta t_k^2 \sigma_{\mathrm{p}}^2}{2}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^3 \sigma_{\mathrm{a}}^2}{3} & 0 & \frac{\Delta t_k^2 \sigma_{\mathrm{a}}^2}{2} \\ \frac{\Delta t_k^2 \sigma_{\mathrm{p}}^2}{2}\mathrm{I} & 0 & \frac{\Delta t_k \sigma_{\mathrm{p}}^2}{1}\mathrm{I} & 0 \\ 0 & \frac{\Delta t_k^2 \sigma_{\mathrm{a}}^2}{2} & 0 & \frac{\Delta t_k \sigma_{\mathrm{a}}^2}{1} \end{bmatrix},$$

where $\sigma_{\mathrm{p}}^2 = 2\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors on the East-North plane and $\sigma_{\mathrm{a}}^2 = 0.01\ \frac{\mathrm{m}^2}{\mathrm{s}^3}$ represents the velocity errors in the vertical direction. [9], [20]

In our simulations, we use base station range measurements, altitude measurements, satellite pseudorange measurements and satellite delta range measurements (see Eq. (2)).

$$\begin{aligned} y^{\mathrm{b}} &= \|r_{\mathrm{b}} - r_{\mathrm{u}}\| + \epsilon_{\mathrm{b}}, \\ y^{\mathrm{a}} &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} r_{\mathrm{u}} + \epsilon_{\mathrm{a}}, \\ y^{\mathrm{s}} &= \|r_{\mathrm{s}} - r_{\mathrm{u}}\| + b + \epsilon_{\mathrm{s}}, \\ \dot{y}^{\mathrm{s}} &= \frac{(r_{\mathrm{s}} - r_{\mathrm{u}})^T}{\|r_{\mathrm{s}} - r_{\mathrm{u}}\|}(v_{\mathrm{s}} - v_{\mathrm{u}}) + \dot{b} + \dot{\epsilon}_{\mathrm{s}}, \end{aligned}$$

where r_{b} is a base station position vector, r_{s} is a satellite position vector, b is clock bias, v_{s} is a satellite velocity vector, $\dot{b}$ is clock drift and ϵ:s are error terms. We use satellite measurements only when there is more than one satellite measurement available, so that bias can be eliminated. These are the same measurements equations as in the papers [1], [15], [17].

Simulations are made using Personal Navigation Filter Framework (PNaFF) [21]. PNaFF is a comprehensive simulation and filtering test bench that we are developing and using in the Personal Positioning Algorithms Research Group. PNaFF uses Earth Centered Earth Fixed (ECEF) coordinate system so we have converted our models from ENU to ECEF.

TABLE II
A summary of 200 different simulations with base station measurements.

Solver	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
$\mathrm{EKF}_{\text{no res.}}$	4	284	622	127	9.9
EKF	10	236	465	83	6.6
EKF2	11	214	433	65	2.8
UKF	23	213	421	58	2.3
SPGMF	62	210	399	69	3.3
PF_{2500}	38	201	397	73	21.9
BGMF	32	194	371	58	2.8
EGMF	32	191	360	57	2.8
Ref	∞	155	287	0	0.1

A. Summary of base station cases

On Table II, we have listed a summary of two hundred 120 second simulations, which use only base station measurements. This means that simulations use base station range measurements with variance $(80\ \mathrm{m})^2$, very inaccurate altitude measurements with variance $(300\ \mathrm{m})^2$ and restrictive information. Restrictive information is in our case base station 120°-sector and maximum range information. So when we have restrictive information we know that user is inside the particular area, which is restricted using sector and maximum range information. Restrictive information are used in the same way as in paper [17]. Summary consist of following columns: *Time* is computation time using Matlab in our implementation, scaled so that computation time of EKF is 10. This gives a rough idea of the relative time complexity of each algorithm. *Err. rms* is 3D root mean square position error. *Err. 95%* gives a radius containing 95 % of the 3D errors. *Err. ref.* is 3D error to reference posterior mean, which is computed using a particle filter with systematic resampling and 10^6 particles [11]. *Inc. %* is a percentage of time where the filter is inconsistent with respect to the general inconsistency test, with risk level 5% [9]. Solvers are sorted so that rms errors are in descending order. PF_N indicates particle filter with systematic resampling and N particles [11], so reference solution is the same as PF_{10^6}. On these simulations BGMF splits, when highly nonlinearity exists, one GM component into four components with equal weights. In that case the parameter l of the Box GM approximation (see Section IV-B) is

$$l = \Phi^{-1}\left(\begin{bmatrix} 0 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & 1 \end{bmatrix}\right).$$

EGMF also uses the same l parameter (see Section VI).

In Table II, the results are the realization of random variables and if we run these simulations again we possibly get a slightly different result. The following conclusions can be drawn based on simulations and theory.

- Both BGMF and EGMF give a better result (in all listed criteria) than SPGMF or PF_{2500}
- Computation time of BGMF and EGMF is approximately half of computation time of SPGMF.
- EGMF gives much better results than traditional EKF ($\mathrm{EKF}_{\text{no res.}}$).

TABLE III
A SUMMARY OF 1000 DIFFERENT SIMULATIONS WITH BASE STATION AND SATELLITE MEASUREMENTS.

Solver	Time $\propto$	Err. rms	Err. 95%	Err. ref	Inc. %
$EKF_{no\ res.}$	6	115	215	23	3.0
EKF	10	101	184	16	2.1
UKF	28	95	174	11	1.1
SPGMF	38	95	173	13	1.2
PF_{2500}	43	95	160	12	1.8
EKF2	11	94	176	12	1.2
BGMF	20	93	170	12	1.0
EGMF	20	92	169	12	1.0
Ref	∞	83	156	0	1.2

- PF_{2500} has a serious inconsistency problem, i.e. it under estimates the state covariance.

B. Summary of mixed cases

In Table III, we have listed a summary of one thousand 120 second simulations, which use both base station and satellite measurements with varying parameters. Parameters are following: variance of base station range measurement $= (30\ \mathrm{m})^2$, variance of satellite pseudorange measurement $\approx (3\ \mathrm{m})^2$ and variance of delta range measurement $\approx (0.1\frac{\mathrm{m}}{\mathrm{s}})^2$. We use restrictive information in the same way as simulations on Section VIII-A. Also notations and filters are the same as simulations on Section VIII-A.

The following conclusions can be drawn based on simulations.

- Order of filter in Table III is almost the same as in Table II.
- Differences between different filters are smaller than in Table II, because there are also satellite measurements (very accurate linear measurements).
- Computation time of BGMF and EGMF is approximately half of computation time of SPGMF.
- EGMF gives much better results than traditional EKF ($EKF_{no\ res.}$).

IX. CONCLUSION

In this article, we have presented two new Gaussian Mixture Filters for the hybrid positioning: the Box GMF and the Efficient GMF. BGMF and EGMF are almost the same filter, because of this their performances are almost the same. Nevertheless EGMF gives slightly better results than BGMF. Both filters outperform the Sigma Point GMF, which outperforms the traditional single-component Kalman type filters such as EKF, UKF and EKF2. EGMF and BGMF also outperform particle filter, when number of particles is selected so that particle filter uses about the same time of computation as EGMF. GMFs and EKF works equivalently if we have only linear measurements, but the more nonlinearity occurs the better result GMFs give compared to EKF.

ACKNOWLEDGMENT

The author would like to thank Niilo Sirola and Robert Piché for their comments and suggestions. This study was partly funded by Nokia Corporation. The author acknowledges the financial support of the Nokia Foundation and the Tampere Graduate School in Information Science and Engineering.

REFERENCES

[1] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter and hybrid positioning," in *Proceedings of ION GNSS 2007, Fort Worth, Texas*, Fort Worth, September 2007, pp. 562–570.

[2] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, July 1971.

[3] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, Aug 1972.

[4] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, ser. Electrical Engineering, T. Kailath, Ed. Prentice-Hall, Inc., 1979.

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

[6] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Mathematics in Science and Engineering. Academic Press, 1970, vol. 64.

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

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

[9] S. Ali-Löytty, N. Sirola, and R. Piché, "Consistency of three Kalman filter extensions in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, Jul. 2005.

[10] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *American Control Conference*, vol. 3, 1995, pp. 1628–1632.

[11] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[12] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Boston, London: Artech House, 2004.

[13] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate analysis*, ser. Probability and mathematical statistics. London Academic Press, 1989.

[14] M. S. Grewal and A. P. Andrews, *Kalman Filtering, Theory and Practice*, ser. Prentice hall information and system sciences. Prentice-Hall, Inc., 1993.

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

[16] D. J. Salmond, "Mixture reduction algorithms for target tracking," *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, pp. 7/1–7/4, 1989.

[17] S. Ali-Löytty and N. Sirola, "A modified Kalman filter for hybrid positioning," in *Proceedings of ION GNSS 2006*, September 2006.

[18] R. S. Bucy and K. D. Senne, "Digital synthesis of non-linear filters," *Automatica*, vol. 7, no. 3, pp. 287–298, May 1971.

[19] R. G. Brown, *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, 1983.

[20] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, ser. Mathematics in Science and Engineering. Academic Press, 1979, vol. 141.

[21] M. Raitoharju, N. Sirola, S. Ali-Löytty, and R. Piché, "PNaFF: a modular software platform for testing hybrid positioning estimation algorithms," in *WPNC08 Workshop on Positioning, Navigation and Communication, Hannover, 28 Mar 2008*, 2008, (accepted). [Online]. Available: http://www.wpnc.net/

PUBLICATION 6

Simo Ali-Löytty: On the convergence of the Gaussian mixture filter. Research report 89, Tampere University of Technology, Institute of Mathematics, 2008.

On the Convergence of the Gaussian Mixture Filter

Simo Ali-Löytty

***Abstract*—This paper presents convergence results for the Box Gaussian Mixture Filter (BGMF). BGMF is a Gaussian Mixture Filter (GMF) that is based on a bank of Extended Kalman Filters. The critical part of GMF is the approximation of probability density function (pdf) as pdf of Gaussian mixture such that its components have small enough covariance matrices. Because GMF approximates prior and posterior as Gaussian mixture it is enough if we have a method to approximate arbitrary Gaussian (mixture) as a Gaussian mixture such that the components have small enough covariance matrices. In this paper, we present the Box Gaussian Mixture Approximation (BGMA) that partitions the state space into specific boxes and matches weights, means and covariances of the original Gaussian in each box to a GM approximation. If the original distribution is Gaussian mixture, BGMA does this approximation separately for each component of the Gaussian mixture. We show that BGMA converges weakly to the original Gaussian (mixture). When we apply BGMA in a Gaussian mixture filtering framework we get BGMF. We show that GMF, and also BGMF, converges weakly to the correct/exact posterior distribution.**

***Index Terms*—Extended Kalman Filter, Filter banks, Filtering techniques, Filtering theory, Gaussian distribution**

I. Introduction

THE problem of estimating the state of a stochastic system from noisy measurement data is considered. We consider the discrete-time nonlinear non-Gaussian system

$$x_k = F_{k-1}x_{k-1} + w_{k-1}, \tag{1a}$$
$$y_k = h_k(x_k) + v_k, \tag{1b}$$

where the vectors $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_{y_k}}$ represent the state of the system and the measurement at time t_k, $k \in \mathbb{N}\backslash\{0\}$, respectively. The state transition matrix F_{k-1} is assumed to be non-singular. We assume that errors w_k and v_k are white, mutually independent and independent of the initial state x_0. The errors as well as the initial state are assumed to have Gaussian mixture distributions. We assume that initial state x_0 and measurement errors v_k have density functions p_{x_0} and p_{v_k}, respectively. We do not assume that state model errors w_k have density functions. These assumptions guarantee that the prior (the conditional probability density function given all past measurements $y_{1:k-1} \triangleq \{y_1, \ldots, y_{k-1}\}$) and the posterior (the conditional probability density function given all current and past measurements $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$) have density functions $p(x_k|y_{1:k-1})$ and $p(x_k|y_{1:k})$, respectively. We use the notation $x^-_{k,\text{exact}}$ for a random variable whose density function is $p(x_k|y_{1:k-1})$ (prior) and $x^+_{k,\text{exact}}$ for a random variable whose density function is $p(x_k|y_{1:k})$ (posterior). The posterior can be determined recursively according to the following relations [1], [2].

Prediction (prior):

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

Update (posterior):

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

where the transitional density

$$p(x_k|x_{k-1}) = p_{w_{k-1}}(x_k - \mathrm{F}_{k-1}x_{k-1})$$

and the likelihood

$$p(y_k|x_k) = p_{v_k}(y_k - h_k(x_k)).$$

The initial condition for the recursion is given by the pdf of the initial state $p_{x_0}(x_0) \triangleq p(x_0|y_{1:0})$. Knowledge of the posterior distribution (3) enables one to compute an optimal state estimate with respect to any criterion. For example, the minimum mean-square error (MMSE) estimate is the conditional mean of x_k [2], [3]. Unfortunately, in general and in our case, the conditional probability density function cannot be determined analytically.

There are many different methods (filters) to compute the approximation of the posterior. One popular approximation is the so-called Extended Kalman Filter [2]–[10], that linearizes the measurement function around the prior mean. EKF works quite well in many applications, where the system model is almost linear and the errors Gaussian but there are plenty of examples where EKF does not work satisfactorily. For example, in satellite positioning systems, EKF works quite well, but in a positioning system based on the range measurements of nearby base stations EKF may diverge [11].

There are also other Kalman Filter extensions to the nonlinear problem, which try to compute the mean and covariance of the posterior, for example Second Order Extended Kalman Filter (EKF2) [3], [4], [11], Iterated Extended Kalman Filter (IEKF) [3], [10] and Unscented Kalman Filters (UKF) [12], [13]. These extensions usually (not always) give better performance than the conventional EKF. However, if the true posterior has multiple peaks, one-component filters that compute only the mean and covariance do not achieve good performance, and because of that we have to use more sophisticated nonlinear filters. Here sophisticated nonlinear filter mean filter that has some convergence results. Possible filters are e.g.

S. Ali-Löytty is with the Department of Mathematics, Tampere University of Technology, Finland e-mail: simo.ali-loytty@tut.fi.

a grid based method (e.g. Point Mass Filter) [2], [14]–[17], Particle Filter [1], [2], [18], [19] and Gaussian Mixture Filter (GMF) [6], [20], [21]. Some comparison of different filters may be found for example in [22], [23].

In this paper we consider Gaussian Mixture Filter, also called Gaussian Sum Filter, which is a filter whose approximate prior and posterior densities are Gaussian Mixtures (GMs), a convex combination of Gaussian densities. One motivation to use GMF is that any continuous density function p_x may be approximated as a density function of GM p_{gm} as closely as we wish in the Lissack-Fu distance sense, which is also norm in $L^1(\mathbb{R}^n)$-space [21] [24, Chapter 18]:

$$\int |p_x(x) - p_{\text{gm}}(x)| \mathrm{d}x. \tag{4}$$

Because the set of all continuous functions, with compact support is dense in $L^1(\mathbb{R}^n)$ [25, Theorem 3.14], we can approximate any density function p_x as a density function of GM [26]. The outline of the conventional GMF algorithm for the system (1) is given as Algorithm 1. In Algorithm 1

Algorithm 1 Gaussian mixture filter

Approximate initial state x_0 as GM x_0^+.
for $k = 1$ to n_{meas} **do**
1) *Prediction*: Compute prior approximation x_k^-.
2) Approximate x_k^- as a new GM $\bar{x}_k^-$ if necessary.
3) *Update*: Compute GM posterior approximation $\bar{x}_k^+$.
4) Reduce the number of components of $\bar{x}_k^+$ and get x_k^+.
end for

all random variables x_0^+, x_k^-, $\bar{x}_k^-$, $\bar{x}_k^+$, and x_k^+ are GMs and approximations of the exact random variables $x_0 \triangleq x_{0,\text{exact}}^+$, $x_{k,\text{exact}}^-$, $x_{k,\text{exact}}^-$, $x_{k,\text{exact}}^+$, and $x_{k,\text{exact}}^+$, respectively. This algorithm stops at time $t_{n_{\text{meas}}}$.

The major contribution of this paper is a new method to approximate a Gaussian mixture as a Gaussian mixture, such that the components have arbitrary small covariance matrices. We call this method the Box Gaussian Mixture Approximation (BGMA) (Section V). We show that BGMA converges weakly to the original GM. One big advantage of BGMA compared to other GM approximations [6], [20], [21] is that BGMA does not require that the norm of the covariance matrices approach zero when the number of mixture components increases. It is sufficient that only parts of the covariance matrices approaches zero when the number of mixture components increases. Thus, BGMA subdivides only those dimensions where we get nonlinear measurements. For example, in positioning applications, nonlinear measurements often depend only on the position. So, using BGMA, it is possible to split only position dimension into boxes instead of the whole state space, which contains usually at least the position vector and the velocity vector. This means that significantly fewer mixture components are needed than in the previous GM approximations.

Another major contribution of this paper is the proof that the general version of the Gaussian Mixture Filter converges weakly to the exact posterior distribution. Especially, the Box Gaussian Mixture Filter (BGMF), which is GMF filter (Algorithm 1) that uses BGMA in Step 2, converges weakly to the exact posterior distribution. In this work BGMF is a generalization of the filter having the same name (BGMF) in our earlier work [27].

An outline of the paper is as follows. In Section II, we study the basics of the GM. In Section III, we give the general algorithm of GMF, which is also the algorithm of BGMF. In Section IV, we present the convergence results of GMF. In Section V, we present the BGMA, show some of its properties and that it converges weakly to the original Gaussian (mixture). In Section VI, we combine the previous sections and present BGMF. Finally in Section VII, we present a small one-step simulation where we compare BGMF and a particle filter [18].

II. Gaussian Mixture

In this section, we define the Gaussian Mixture (GM) distribution and present some of its properties, such as the mean, covariance, linear transformation and sum. Because GM is a convex combination of Gaussians, we first define the Gaussian distribution.

Definition 1 (Gaussian): An n-dimensional random variable x_j is Gaussian if its characteristic function has the form

$$\varphi_{x_j}(t) = \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right), \tag{5}$$

where $\mu_j \in \mathbb{R}^n$ and $\Sigma_j \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite ($\Sigma_j \geq 0$)[1]. We use the abbreviation

$$x_j \sim \mathrm{N}_n(\mu_j, \Sigma_j) \quad \text{or} \quad x_j \sim \mathrm{N}(\mu_j, \Sigma_j).$$

Gaussian random variable is well defined, that is the function (5) is a proper characteristic function [28, p.297].

Theorem 2 (Mean and Covariance of Gaussian): Assume that $x_j \sim \mathrm{N}(\mu_j, \Sigma_j)$. Then $\mathrm{E}(x_j) = \mu_j$ and $\mathrm{V}(x_j) = \Sigma_j$

Proof: We use the properties of the characteristic function [29, p.34] to get

$$\begin{aligned}
\mathrm{E}(x_j) &= \frac{1}{i}\left(\varphi'_{x_j}(t)|_{t=0}\right)^T \\
&= \frac{1}{i}\exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right)(i\mu_j - \Sigma_j t)\Big|_{t=0} \\
&= \mu_j
\end{aligned}$$

and

$$\begin{aligned}
\mathrm{V}(x_j) &= \mathrm{E}(x_j x_j^T) - \mathrm{E}(x_j)\,\mathrm{E}(x_j)^T \\
&= -\varphi''_{x_j}(t)|_{t=0} - \mu_j\mu_j^T \\
&= -\Bigg[\left((i\mu_j - \Sigma_j t)(i\mu_j - \Sigma_j t)^T - \Sigma_j\right)\cdot\ldots \\
&\quad \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right)\Bigg]\Bigg|_{t=0} - \mu_j\mu_j^T \\
&= \mu_j\mu_j^T + \Sigma_j - \mu_j\mu_j^T \\
&= \Sigma_j.
\end{aligned}$$

□

[1] If $\mathrm{A} \geq \mathrm{B}$ then both matrices A and B are symmetric and $x^T(\mathrm{A}-\mathrm{B})x \geq 0$ for all x.

Theorem 3 (Density function of non-singular Gaussian): Assume that $x_j \sim \mathrm{N}(\mu_j, \Sigma_j)$, where $\Sigma_j > 0$ (positive definite matrix)[2]. Then the density function of the random variable x is

$$p_{x_j}(\xi) \triangleq \mathrm{N}^{\mu_j}_{\Sigma_j}(\xi) = \frac{\exp\left(-\frac{1}{2}\|\xi-\mu_j\|^2_{\Sigma_j^{-1}}\right)}{(2\pi)^{\frac{n}{2}}\sqrt{\det(\Sigma_j)}},$$

where $\|\xi-\mu_j\|^2_{\Sigma_j^{-1}} = (\xi-\mu_j)^T\Sigma_j^{-1}(\xi-\mu_j)$.

Proof: We know that the characteristic function $\varphi_{x_j}(t)$ is absolutely integrable. Thus using the properties of the characteristic function [29, p.33] we get

$$\begin{aligned} p_{x_j}(\xi) &= \frac{1}{(2\pi)^n}\int \exp\left(-it^T\xi\right)\varphi_{x_j}(t)\mathrm{d}t \\ &= \frac{1}{(2\pi)^n}\int \exp\left(it^T(\mu_j-\xi)-\frac{1}{2}t^T\Sigma_j t\right)\mathrm{d}t \\ &= \frac{\frac{\sqrt{\det(\Sigma_j)}}{(2\pi)^{\frac{n}{2}}}\int \exp\left(it^T(\mu_j-\xi)-\frac{1}{2}t^T\Sigma_j t\right)\mathrm{d}t}{(2\pi)^{\frac{n}{2}}\sqrt{\det\left(\Sigma_j\right)}} \\ &\overset{\star}{=} \frac{\exp\left(-\frac{1}{2}\left(\xi-\mu_j\right)^T\Sigma_j^{-1}\left(\xi-\mu_j\right)\right)}{(2\pi)^{\frac{n}{2}}\sqrt{\det\left(\Sigma_j\right)}} \end{aligned}$$

$\star$ see [28, p.297]. □

Definition 4 (Gaussian Mixture): An n-dimensional random variable x is an N-component Gaussian Mixture if its characteristic function has the form

$$\varphi_x(t) = \sum_{j=1}^N \alpha_j \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right), \tag{6}$$

where $\mu_j \in \mathbb{R}^n$, $\Sigma_j \in \mathbb{R}^{n\times n}$ is symmetric positive semidefinite, $\alpha_j \geq 0$, and $\sum_{j=1}^N \alpha_j = 1$. We use the abbreviation

$$x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j,N)}.$$

We show that GM is well defined, which means that function (6) is in fact a characteristic function. First, assume that all matrices Σ_j are positive definite. We know that function

$$p(\xi) = \sum_{j=1}^N \alpha_j \,\mathrm{N}^{\mu_j}_{\Sigma_j}(\xi), \tag{7}$$

is a density function, that is $\int p(\xi)\mathrm{d}\xi = 1$ and $p(\xi) \geq 0$ for all ξ. Because

$$\begin{aligned} \int \exp(it^T\xi)p(\xi)\mathrm{d}\xi &= \int \exp(it^T\xi)\left(\sum_{j=1}^N \alpha_j \,\mathrm{N}^{\mu_j}_{\Sigma_j}(\xi)\right)\mathrm{d}\xi \\ &= \sum_{j=1}^N \alpha_j \int \exp(it^T\xi)\,\mathrm{N}^{\mu_j}_{\Sigma_j}(\xi)\mathrm{d}\xi \\ &\overset{(5)}{=} \sum_{j=1}^N \alpha_j \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right), \end{aligned}$$

function (6) is the characteristic function of a continuous n-dimensional Gaussian Mixture. The density function of this distribution is given in equation (7).

Now, let at least one of the covariance matrices Σ_j be singular. Take $\epsilon > 0$ and consider the positive definite symmetric matrices $\Sigma_j^\epsilon = \Sigma_j + \epsilon\mathrm{I}$. Then by what has been proved,

$$\varphi_{x_\epsilon}(t) = \sum_{j=1}^N \alpha_j \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j^\epsilon t\right)$$

is a characteristic function. Because function (6) is the limit of characteristic functions

$$\lim_{\epsilon\to 0}\varphi_{x_\epsilon}(t) = \sum_{j=1}^N \alpha_j \exp\left(it^T\mu_j - \frac{1}{2}t^T\Sigma_j t\right),$$

and it is continuous at $t = 0$, then this function (6) is a characteristic function [28, p.298].

Theorem 5 (Mean and Covariance of mixture): Assume that

$$\varphi_x(t) = \sum_{j=1}^N \alpha_j \varphi_{x_j}(t)$$

where $\mathrm{E}(x_j) = \mu_j \in \mathbb{R}^n$, $\mathrm{V}(x_j) = \Sigma_j \in \mathbb{R}^{n\times n}$, $\alpha_j \geq 0$, and $\sum_{j=1}^N \alpha_j = 1$. Then

$$\mathrm{E}(x) = \sum_{j=1}^N \alpha_j\mu_j \triangleq \mu \quad \text{and}$$

$$\mathrm{V}(x) = \sum_{j=1}^N \alpha_j\left(\Sigma_j + (\mu_j-\mu)(\mu_j-\mu)^T\right).$$

Proof: We use the properties of the characteristic function [29, p.34] to get

$$\begin{aligned} \mathrm{E}(x) &= \frac{1}{i}\left(\varphi'_x(t)|_{t=0}\right)^T \\ &= \sum_{j=1}^N \alpha_j \frac{1}{i}\left(\varphi'_{x_j}(t)|_{t=0}\right)^T \\ &= \sum_{j=1}^N \alpha_j\mu_j \triangleq \mu \end{aligned}$$

and

$$\begin{aligned} \mathrm{V}(x) &= -\mathrm{E}(x)\,\mathrm{E}(x)^T - \varphi''_x(t)|_{t=0} \\ &= -\mu\mu^T + \sum_{j=1}^N \alpha_j\left(-\varphi''_{x_j}(t)|_{t=0}\right) \\ &= -\mu\mu^T + \sum_{j=1}^N \alpha_j\left(\Sigma_j + \mu_j\mu_j^T\right) \\ &= \sum_{j=1}^N \alpha_j\left(\Sigma_j + \mu_j\mu_j^T - \mu\mu^T\right) \\ &= \sum_{j=1}^N \alpha_j\left(\Sigma_j + (\mu_j-\mu)(\mu_j-\mu)^T\right). \end{aligned}$$

□

[2] If $\mathrm{A} > \mathrm{B}$ then both matrices A and B are symmetric and $x^T(\mathrm{A}-\mathrm{B})x > 0$ for all $x \neq 0$.

Note that Theorem 5 does not assume that the distribution is a Gaussian mixture, these results are valid for all mixtures.

Theorem 6 (Linear transformation and sum of GM): Assume that an n-dimensional random variable

$$x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j,N)}$$

and an m-dimensional random variable

$$v \sim \mathrm{M}(\beta_k, r_k, \mathrm{R}_k)_{(k,M)}$$

are independent. Define a random variable $y = \mathrm{H}x + v$, where matrix $\mathrm{H} \in \mathbb{R}^{m \times n}$. Then

$$y \sim \mathrm{M}(\alpha_{j(l)}\beta_{k(l)}, \mathrm{H}\mu_{j(l)} + r_{k(l)}, \mathrm{H}\Sigma_{j(l)}\mathrm{H}^T + \mathrm{R}_{k(l)})_{(l,NM)},$$

where $j(l) = [(l-1) \mod N]+1$ and $k(l) = \lceil \frac{l}{N} \rceil$.[3] We also use the abbreviation

$$y \sim \mathrm{M}(\alpha_j\beta_k, \mathrm{H}\mu_j + r_k, \mathrm{H}\Sigma_j\mathrm{H}^T + \mathrm{R}_k)_{(j*k,NM)}.$$

Proof: Since x and v are independent, also $\mathrm{H}x$ and v are independent.

$$\begin{aligned}
&\varphi_{\mathrm{H}x+v}(t) \overset{\text{ind.}}{=} \varphi_{\mathrm{H}x}(t)\varphi_v(t) \\
&= \mathrm{E}\left(\exp\left(it^T(\mathrm{H}x)\right)\right)\varphi_v(t) \\
&= \mathrm{E}\left(\exp\left(i\left(\mathrm{H}^T t\right)^T x\right)\right)\varphi_v(t) \\
&= \varphi_x(\mathrm{H}^T t)\varphi_v(t) \\
&= \sum_{j=1}^{N} \alpha_j \exp\left(it^T\mathrm{H}\mu_j - \frac{1}{2}t^T\mathrm{H}\Sigma_j\mathrm{H}^T t\right)\cdot\ldots \\
&\quad \sum_{k=1}^{M} \beta_k \exp\left(it^T r_k - \frac{1}{2}t^T\mathrm{R}_k t\right) \\
&= \sum_{l=1}^{NM} \alpha_{j(l)}\beta_{k(l)} \exp\Big(it^T\left(\mathrm{H}\mu_{j(l)} + r_{k(l)}\right)\ldots \\
&\qquad\qquad - \frac{1}{2}t^T\left(\mathrm{H}\Sigma_{j(l)}\mathrm{H}^T + \mathrm{R}_{k(l)}\right)t\Big).
\end{aligned}$$

□

Corollary 7: Assume that an n-dimensional random variable

$$x \sim \mathrm{M}(\alpha_j, \mu_j, \Sigma_j)_{(j,N)}$$

and

$$y = \mathrm{A}x + b,$$

where $\mathrm{A} \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Then

$$y \sim \mathrm{M}(\alpha_j, \mathrm{A}\mu_j + b, \mathrm{A}\Sigma_j\mathrm{A}^T)_{(j,N)}.$$

Proof: Now $b \sim \mathrm{M}(1, b, 0)_{(k,1)}$. Constant random variable b and x are independent, so using Theorem 6 we get

$$y \sim \mathrm{M}(\alpha_j, \mathrm{A}\mu_j + b, \mathrm{A}\Sigma_j\mathrm{A}^T)_{(j,N)}.$$

□

Note that if $x \sim \mathrm{N}(\mu_1, \Sigma_1)$ then $x \sim \mathrm{M}(1, \mu_j, \Sigma_j)_{(j,1)}$. So Theorem 6 and Corollary 7 hold also for Gaussian distributions.

[3]Ceiling function $\lceil x \rceil = \min\{n \in \mathbb{Z} | n \geq x\}$ and modulo function $(a \mod n) = a + n\lceil -\frac{a}{n} \rceil$.

III. Algorithm of Gaussian Mixture Filter

In this section, we give the algorithm of Gaussian Mixture Filter for the system (1) (Algorithm 2). The subsections III-A–III-D present the details of this algoritm. Algorithm 2 uses the following assumptions:

1) Initial state
$$x_0 \sim \mathrm{M}(\alpha^+_{i,0}, \mu^+_{i,0}, \Sigma^+_{i,0})_{(i,n_0)}$$
is a continuous Gaussian Mixture, that is, $\Sigma^+_{i,0} > 0$ for all i.
2) Errors are GMs
$$w_k \sim \mathrm{M}(\gamma_{j,k}, \bar{w}_{j,k}, \mathrm{Q}_{j,k})_{(j,n_{w_k})} \text{ and}$$
$$v_k \sim \mathrm{M}(\beta_{j,k}, \bar{v}_{j,k}, \mathrm{R}_{j,k})_{(j,n_{v_k})},$$
where all $\mathrm{R}_{j,k} > 0$.
3) Measurement functions are of the form
$$h_k(x) = \bar{h}_k(x_{1:d}) + \bar{\mathrm{H}}_k x. \tag{8}$$
This means that the nonlinear part $\bar{h}_k(x_{1:d})$ only depends on the first d dimensions ($d \leq n_x$). We assume that functions $\bar{h}_k(x_{1:d})$ are twice continuously differentiable in $\mathbb{R}^d \setminus \{s_1, \ldots, s_{n_s}\}$.[4]

Algorithm 2 Gaussian mixture filter

Initial state at time t_0: $x^+_0 \sim \mathrm{M}(\alpha^+_{i,0}, \mu^+_{i,0}, \Sigma^+_{i,0})_{(i,n_0)}$
for $k = 1$ to n_{meas} **do**
1) Prediction (see Sec. III-A):
$$x^-_k \sim \mathrm{M}(\alpha^-_{i*j,k}, \mu^-_{i*j,k}, \Sigma^-_{i*j,k})_{(i*j,n^-_k)}$$
2) Approximate x^-_k as a new GM $\bar{x}^-_k$ if necessary (see Sec. III-B):
$$\bar{x}^-_k \sim \mathrm{M}(\bar{\alpha}^-_{i,k}, \bar{\mu}^-_{i,k}, \bar{\Sigma}^-_{i,k})_{(i,\bar{n}^-_k)}$$
3) Update (see Sec. III-C):
$$\bar{x}^+_k \sim \mathrm{M}(\bar{\alpha}^+_{i*j,k}, \bar{\mu}^+_{i*j,k}, \bar{\Sigma}^+_{i*j,k})_{(i*j,\bar{n}^+_k)}$$
4) Reduce the number of components (see Sec. III-D):
$$x^+_k \sim \mathrm{M}(\alpha^+_{i,k}, \mu^+_{i,k}, \Sigma^+_{i,k})_{(i,n_k)}$$
end for

A. Prediction, Step (1)

Prediction is based on Eq. (1a) and Thm. 6 (see also Eq. (2)).

$$x^-_k \sim \mathrm{M}(\alpha^-_{i*j,k}, \mu^-_{i*j,k}, \Sigma^-_{i*j,k})_{(i*j,n^-_k)},$$

where

$$\begin{aligned}
n^-_k &= n_{k-1}n_{w_{k-1}}, \\
\alpha^-_{i*j,k} &= \alpha^+_{i,k-1}\gamma_{j,k-1}, \\
\mu^-_{i*j,k} &= \mathrm{F}_{k-1}\mu^+_{i,k-1} + \bar{w}_{j,k-1} \text{ and} \\
\Sigma^-_{i*j,k} &= \mathrm{F}_{k-1}\Sigma^+_{i,k-1}{\mathrm{F}_{k-1}}^T + \mathrm{Q}_{j,k-1}.
\end{aligned}$$

[4]For example, in positioning applications that are based on range measurements and a constant velocity model $n_x = 6$ (position+velocity), $d = 3$ (position) and s_i is position vector of the ith base station [11], [27]

B. Approximate GM as a new GM, Step (2)

There are different methods to compute Step (2). Here we present one conventional method. Another method, namely, the Box Gaussian Mixture Approximation, is given in Section V. The density function of a new GM approximation $p_{\bar{x}_k^-}$ is [20]

$$p_{\bar{x}_k^-}(\xi) \propto \sum_{i=1}^{\bar{n}_{k,\mathrm{g}}^-} p_{x_k^-}(\xi_{\mathrm{g}}^{(i)})\, \mathrm{N}_{c_{\mathrm{g}}\mathrm{I}}^{\xi_{\mathrm{g}}^{(i)}}(\xi), \tag{9}$$

where the mean values $\xi_{\mathrm{g}}^{(i)}$ are used to establish a grid in the region of the state space that contains the significant part of the probability mass, $\bar{n}_{k,\mathrm{g}}^-$ is the number of grid points and $c_{\mathrm{g}} > 0$ is determined such that the error in the approximation, e.g. the Lissack-Fu distance (4), is minimized. So

$$\bar{x}_k^- \sim \mathrm{M}(\bar{\alpha}_{i,k}^-, \bar{\mu}_{i,k}^-, \bar{\Sigma}_{i,k}^-)_{(i,\bar{n}_k^-)},$$

where

$$\begin{aligned}
\bar{n}_k^- &= \bar{n}_{k,\mathrm{g}}^-,\\
\bar{\alpha}_{i,k}^- &= \frac{p_{x_k^-}(\xi_{\mathrm{g}}^{(i)})}{\sum_{i=1}^{\bar{n}_{k,\mathrm{g}}^-} p_{x_k^-}(\xi_{\mathrm{g}}^{(i)})},\\
\bar{\mu}_{i,k}^- &= \xi_{\mathrm{g}}^{(i)} \text{ and}\\
\bar{\Sigma}_{i,k}^- &= c_{\mathrm{g}}\mathrm{I}.
\end{aligned}$$

It can be shown that $p_{\bar{x}_k^-}(x)$ converges almost everywhere uniformly to the density function of x_k^- as the number of components $\bar{n}_k^-$ increases and c_{g} approaches zero [20], [21]. Moreover, the Lissack-Fu distance (4) of the approximation converges to zero.

Step (2) is executed only when necessary. If it is not necessary then $\bar{x}_k^- = x_k^-$. A conventional criterion is to check if some prior covariances do not satisfy inequality $\mathrm{P}_i^- < \epsilon\mathrm{I}$, for some predefined ϵ, where P_i^- is the covariance of the ith component [6, p.216]. Note that finding reasonable grid points $\xi_{\mathrm{g}}^{(i)}$ and an optimal constant $c_{\mathrm{g}} > 0$ usually requires some heavy computation.

C. Update, Step 3

The update Eq. (3) is usually computed approximately using a bank of EKFs. In this paper we use that approximation. It is possible to compute the update step using a bank of other Kalman-type filters [30] or a bank of PFs [31]. Using the bank of EKFs approximation we get

$$\begin{aligned}
p_{\bar{x}_k^+}(\xi) &\propto p_{v_k}(y_k - h_k(\xi)) p_{\bar{x}_k^-}(\xi)\\
&= \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \beta_{j,k}\, \mathrm{N}_{\mathrm{R}_{j,k}}^{\bar{v}_{j,k}}(y_k - h_k(\xi))\, \bar{\alpha}_{i,k}^-\, \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi)\\
&\approx \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \bar{\alpha}_{i,k}^-\beta_{j,k}\, \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi) \cdot \ldots\\
&\quad \mathrm{N}_{\mathrm{R}_{j,k}}^{\bar{v}_{j,k}}\Big(y_k - h_k(\bar{\mu}_{i,k}^-) - \mathrm{H}_{i,k}(\xi - \bar{\mu}_{i,k}^-)\Big)\\
&= \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \bar{\alpha}_{i,k}^-\beta_{j,k}\, \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi) \cdot \ldots\\
&\quad \mathrm{N}_{\mathrm{R}_{j,k}}^{\mathrm{H}_{i,k}\xi}\Big(y_k - h_k(\bar{\mu}_{i,k}^-) + \mathrm{H}_{i,k}\bar{\mu}_{i,k}^- - \bar{v}_{j,k}\Big)\\
&\overset{\text{Thm. 25}}{=} \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \bar{\alpha}_{i,k}^-\beta_{j,k}\, \mathrm{N}_{\bar{\Sigma}_{i*j,k}^+}^{\bar{\mu}_{i*j,k}^+}(\xi) \cdot \ldots\\
&\quad \mathrm{N}_{\mathrm{H}_{i,k}\bar{\mu}_{i,k}^-}^{\mathrm{H}_{i,k}\bar{\Sigma}_{i,k}^-\mathrm{H}_{i,k}^T + \mathrm{R}_{j,k}}\Big(y_k - h_k(\bar{\mu}_{i,k}^-) + \mathrm{H}_{i,k}\bar{\mu}_{i,k}^- - \bar{v}_{j,k}\Big),
\end{aligned} \tag{10}$$

where $\mathrm{H}_{i,k} = \frac{\partial h_k(\xi)}{\partial \xi}\big|_{\xi=\bar{\mu}_{i,k}^-}$. So

$$\bar{x}_k^+ \sim \mathrm{M}(\bar{\alpha}_{i*j,k}^+, \bar{\mu}_{i*j,k}^+, \bar{\Sigma}_{i*j,k}^+)_{(i*j,\bar{n}_k^+)}, \tag{11}$$

where

$$\begin{aligned}
\bar{n}_k^+ &= n_{v_k}\bar{n}_k^-,\\
\bar{\alpha}_{i*j,k}^+ &= \frac{\bar{\alpha}_{i,k}^-\beta_{j,k}\, \mathrm{N}_{\mathrm{H}_{i,k}\bar{\Sigma}_{i,k}^-\mathrm{H}_{i,k}^T + \mathrm{R}_{j,k}}^{h_k(\bar{\mu}_{i,k}^-) + \bar{v}_{j,k}}(y_k)}{\sum_{j=1}^{n_{v_k}}\sum_{i=1}^{\bar{n}_k^-} \bar{\alpha}_{i,k}^-\beta_{j,k}\, \mathrm{N}_{\mathrm{H}_{i,k}\bar{\Sigma}_{i,k}^-\mathrm{H}_{i,k}^T + \mathrm{R}_{j,k}}^{h_k(\bar{\mu}_{i,k}^-) + \bar{v}_{j,k}}(y_k)},\\
\bar{\mu}_{i*j,k}^+ &= \bar{\mu}_{i,k}^- + \mathrm{K}_{i*j,k}(y_k - h_k(\bar{\mu}_{i,k}^-) - \bar{v}_{j,k}),\\
\bar{\Sigma}_{i*j,k}^+ &= (\mathrm{I} - \mathrm{K}_{i*j,k}\mathrm{H}_{i,k})\,\bar{\Sigma}_{i,k}^- \text{ and}\\
\mathrm{K}_{i*j,k} &= \bar{\Sigma}_{i,k}^-\mathrm{H}_{i,k}^T\Big(\mathrm{H}_{i,k}\bar{\Sigma}_{i,k}^-\mathrm{H}_{i,k}^T + \mathrm{R}_{j,k}\Big)^{-1}.
\end{aligned}$$

D. Reduce the number of components, Step 4

One major challenge when using GMF efficiently is keeping the number of components as small as possible without losing significant information. There are many ways to do so. We use two different types of mixture reduction algorithms: forgetting and merging [21], [30], [32].

1) Forgetting components: We re-index the posterior approximation $\bar{x}_k^+$ Eq. (11) such that

$$\bar{x}_k^+ \sim \mathrm{M}(\bar{\alpha}_{i,k}^+, \bar{\mu}_{i,k}^+, \bar{\Sigma}_{i,k}^+)_{(i,\bar{n}_k^+)},$$

where $\bar{\alpha}_{i,k}^+ \geq \bar{\alpha}_{i+1,k}^+$. Let $\epsilon_{\mathrm{f}} = \frac{1}{2N}$ be the threshold value. Let $\bar{n}_{k,\mathrm{f}}^+$ be the index such that

$$\sum_{i=1}^{\bar{n}_{k,\mathrm{f}}^+} \bar{\alpha}_{i,k}^+ \geq 1 - \epsilon_{\mathrm{f}}$$

We forget all mixture components whose index $i > \bar{n}_{k,\mathrm{f}}^+$ and after normalization we get $\bar{x}_{k,\mathrm{f}}^+$. Now

$$\bar{x}_{k,\mathrm{f}}^+ \sim \mathrm{M}(\bar{\alpha}_{i,k,\mathrm{f}}^+, \bar{\mu}_{i,k,\mathrm{f}}^+, \bar{\Sigma}_{i,k,\mathrm{f}}^+)_{(i,\bar{n}_{k,\mathrm{f}}^+)}, \tag{12}$$

where

$$\bar{\alpha}^+_{i,k,\mathrm{f}} = \frac{\bar{\alpha}^+_{i,k}}{\sum_{j=1}^{\bar{n}^+_{k,\mathrm{f}}} \bar{\alpha}^+_{j,k}},\ \bar{\mu}^+_{i,k,\mathrm{f}} = \bar{\mu}^+_{i,k} \text{ and } \bar{\Sigma}^+_{i,k,\mathrm{f}} = \bar{\Sigma}^+_{i,k}.$$

2) Merging components: Our merging procedure is iterative. We merge two components, say the i_1th component and the i_2th component, into one component using moment matching method if they are sufficiently similar, that is if (for simplicity we suppress indices k and f) both

$$\|\bar{\mu}^+_{i_1} - \bar{\mu}^+_{i_2}\| \leq \epsilon_{\mathrm{m}_1} \text{ and} \tag{13a}$$
$$\|\bar{\Sigma}^+_{i_1} - \bar{\Sigma}^+_{i_2}\| \leq \epsilon_{\mathrm{m}_2} \tag{13b}$$

inequalities hold. Here we assume that the threshold values $\epsilon_{\mathrm{m}_1} \underset{N\to\infty}{\longrightarrow} 0$ and $\epsilon_{\mathrm{m}_2} \underset{N\to\infty}{\longrightarrow} 0$. The new component, which replaces components i_1 and i_2, is a component whose weight, mean and covariance matrix are

$$\bar{\alpha}^+_{i_1,\mathrm{m}} = \bar{\alpha}^+_{i_1} + \bar{\alpha}^+_{i_2}$$
$$\bar{\mu}^+_{i_1,\mathrm{m}} = \frac{\bar{\alpha}^+_{i_1}}{\bar{\alpha}^+_{i_1,\mathrm{m}}}\bar{\mu}^+_{i_1} + \frac{\bar{\alpha}^+_{i_2}}{\bar{\alpha}^+_{i_1,\mathrm{m}}}\bar{\mu}^+_{i_2} \text{ and}$$
$$\bar{\Sigma}^+_{i_1,\mathrm{m}} = \frac{\bar{\alpha}^+_{i_1}}{\bar{\alpha}^+_{i_1,\mathrm{m}}}\left(\bar{\Sigma}^+_{i_1} + \left(\bar{\mu}^+_{i_1} - \bar{\mu}^+_{i_1,\mathrm{m}}\right)\left(\bar{\mu}^+_{i_1} - \bar{\mu}^+_{i_1,\mathrm{m}}\right)^T\right) + \ldots$$
$$\frac{\bar{\alpha}^+_{i_2}}{\bar{\alpha}^+_{i_1,\mathrm{m}}}\left(\bar{\Sigma}^+_{i_2} + \left(\bar{\mu}^+_{i_2} - \bar{\mu}^+_{i_1,\mathrm{m}}\right)\left(\bar{\mu}^+_{i_2} - \bar{\mu}^+_{i_1,\mathrm{m}}\right)^T\right),$$

respectively. After re-indexing (forgetting component i_2) we merge iteratively more components until there are no sufficiently similar components, components that statisfy inequalities (13). Herewith, after re-indexing, we get

$$x^+_k \sim \mathrm{M}(\alpha^+_{i,k}, \mu^+_{i,k}, \Sigma^+_{i,k})_{(i,n_k)}.$$

IV. Convergence results of GMF

In this section, we present the convergence results of GMF. First we present some well know convergence results.

Definition 8 (Weak convergence): Let x and x_N, where $N \in \mathbb{N}$, be n-dimensional random variables. We say that x_N converges (weakly) to x if

$$F_{x_N}(\xi) \underset{N\to\infty}{\longrightarrow} F_x(\xi),$$

for all points ξ for which the cumulative density function $F_x(\xi)$ is continuous. We use the abbreviation

$$x_N \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x.$$

Theorem 9: The following conditions are equivalent

1) $x_N \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x$.
2) $\mathrm{E}(g(x_N)) \underset{N\to\infty}{\longrightarrow} \mathrm{E}(g(x))$ for all continuous functions g that vanish outside a compact set.
3) $\mathrm{E}(g(x_N)) \underset{N\to\infty}{\longrightarrow} \mathrm{E}(g(x))$ for all continuous bounded functions g.
4) $\mathrm{E}(g(x_N)) \underset{N\to\infty}{\longrightarrow} \mathrm{E}(g(x))$ for all bounded measurable functions g such that $\mathrm{P}(x \in C(g)) = 1$, where $C(g)$ is the continuity set of g.

Proof: See, for example, the book [33, p.13]. □

Theorem 10 (Slutsky Theorems): 1) If

$$x_N \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x,$$

and if $f : \mathbb{R}^n \to \mathbb{R}^k$ is such that $\mathrm{P}(x \in C(f)) = 1$, where $C(f)$ is the continuity set of f, then

$$f(x_N) \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} f(x).$$

2) If $\{x_N\}$ and $\{y_N\}$ are independent, and if $x_N \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x$ and $y_N \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} y$, then

$$\begin{bmatrix} x_N \\ y_N \end{bmatrix} \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} \begin{bmatrix} x \\ y \end{bmatrix},$$

where x and y are taken to be independent.

Proof: See, for example, the book [33, p.39, p.42]. □

Now we show the convergence results of GMF (Algorithm 2). The outline of the convergence results of GMF is given in Algorithm 3. The details of the convergence results are given in Sections IV-A–IV-D. The initial step of Algorithm 3 is self-evident because we assume that the initial state is a Gaussian mixture. Furthermore if our (exact) initial state has an arbitrary density function it is possible to approximate it as a Gaussian mixture such that the approximation weakly converges to the exact initial state (Sec. III-B).

Algorithm 3 Outline of showing the convergence results of the Gaussian mixture filter (Algorithm 2)

Initial state: Show that $x^+_0 \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{0,\mathrm{exact}}$.
for $k = 1$ to n_{meas} **show**
1) Prediction, Sec. IV-A:
$$x^+_{k-1} \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k-1,\mathrm{exact}} \Longrightarrow x^-_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^-_{k,\mathrm{exact}}.$$
2) Approximation, Sec. IV-B:
$$x^-_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^-_{k,\mathrm{exact}} \Longrightarrow \bar{x}^-_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^-_{k,\mathrm{exact}}.$$
3) Update, Sec. IV-C:
$$\bar{x}^-_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^-_{k,\mathrm{exact}} \Longrightarrow \bar{x}^+_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k,\mathrm{exact}}.$$
4) Reduce the number of components, Sec. IV-D:
$$\bar{x}^+_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k,\mathrm{exact}} \Longrightarrow x^+_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k,\mathrm{exact}}.$$
end for

A. Convergence results of Step 1 (prediction)

Here we show that if $x^+_{k-1} \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k-1,\mathrm{exact}}$ then $x^-_k \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^-_{k,\mathrm{exact}}$ (Thm. 11).

Theorem 11 (Prediction convergence): If

$$x^+_{k-1} \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x^+_{k-1,\mathrm{exact}},$$

w_{k-1} and $\{x^+_{k-1,N}|N\in\mathbb{N}\}$[5] are independent, and w_{k-1} and $x^+_{k-1,\text{exact}}$ are independent then

$$x^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}.$$

Proof: Because w_{k-1} and $\{x^+_{k-1,N}|N\in\mathbb{N}\}$ are independent then w_{k-1} and $\{\mathrm{F}_{k-1}x^+_{k-1,N}|N\in\mathbb{N}\}$ are independent. From Thm. 10 we see that

$$\mathrm{F}_{k-1}x^+_{k-1} \xrightarrow[N\to\infty]{\text{w}} \mathrm{F}_{k-1}x^+_{k-1,\text{exact}}$$

and

$$\begin{bmatrix} \mathrm{F}_{k-1}x^+_{k-1} \\ w_{k-1} \end{bmatrix} \xrightarrow[N\to\infty]{\text{w}} \begin{bmatrix} \mathrm{F}_{k-1}x^+_{k-1,\text{exact}} \\ w_{k-1} \end{bmatrix}.$$

Because

$$x^-_k = \begin{bmatrix} \mathrm{I} & \mathrm{I} \end{bmatrix} \begin{bmatrix} \mathrm{F}_{k-1}x^+_{k-1} \\ w_{k-1} \end{bmatrix} \text{ and}$$
$$x^-_{k,\text{exact}} = \begin{bmatrix} \mathrm{I} & \mathrm{I} \end{bmatrix} \begin{bmatrix} \mathrm{F}_{k-1}x^+_{k-1,\text{exact}} \\ w_{k-1} \end{bmatrix}$$

it follows that

$$x^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}.$$

□

B. Convergence results of Step 2 (approximation)

Here we show that if $x^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}$ then $\bar{x}^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}$. It is enough to show that

$$F_{x^-_k}(\xi) - F_{\bar{x}^-_k}(\xi) \underset{N\to\infty}{\longrightarrow} 0,$$

for all ξ. If we use the conventional approximation method see Sec. III-B and if we use the new method (BGMA) see Thm. 21 and Corollary 22.

Furthermore, we require that the most of the covariance matrices $\bar{\Sigma}^-_{k,i,N}$ of the components of our GM approximation $\bar{x}^-_{k,N}$ are arbitrary small. That is if $\epsilon > 0$ then there is N_0 such that for all $N > N_0$

$$\sum_{j=1}^{d} \left(\bar{\Sigma}^-_{k,i,N}\right)_{j,j} < \epsilon, \tag{14}$$

for almost all i. Both the conventional approximation (Sec. III-B) and BGMA (Sec. V and Corollary 20) satisfy this requirement.

C. Convergence results of Step 3 (update)

Here we show that if $\bar{x}^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}$ then $\bar{x}^+_k \xrightarrow[N\to\infty]{\text{w}} x^+_{k,\text{exact}}$. The distribution $\bar{x}^+_k$ is computed from the prior approximation $\bar{x}^-_k$ using the bank of EKF approximations (Sec. III-C). We use the abbreviation $\bar{x}^{+,\text{Bayes}}_k$ for the distribution that is obtained from the prior approximation $\bar{x}^-_k$ using the exact update Eq. (3) (see also Eq. (15)). First we show that if $\bar{x}^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}}$ then $\bar{x}^{+,\text{Bayes}}_k \xrightarrow[N\to\infty]{\text{w}} x^+_{k,\text{exact}}$ (Thm. 12). After that it is enough to show that

$$F_{\bar{x}^{+,\text{Bayes}}_k}(\xi) - F_{\bar{x}^+_k}(\xi) \underset{N\to\infty}{\longrightarrow} 0,$$

for all ξ (Thm. 13).

Theorem 12 (Correct posterior convergence): Assume that

$$\bar{x}^-_k \xrightarrow[N\to\infty]{\text{w}} x^-_{k,\text{exact}},$$

and the density functions of $\bar{x}^-_k$ and $x^-_{k,\text{exact}}$ are $p_{\bar{x}^-_k}(\xi)$ and $p_{x^-_{k,\text{exact}}}(\xi)$, respectively. Now

$$\bar{x}^{+,\text{Bayes}}_k \xrightarrow[N\to\infty]{\text{w}} x^+_{k,\text{exact}}.$$

Proof: Using the assumptions and Thm. 9 we get that

$$\int p(y_k|\xi)p_{\bar{x}^-_{k,N}}(\xi)\mathrm{d}\xi \underset{N\to\infty}{\longrightarrow} \int p(y_k|\xi)p_{x^-_{k,\text{exact}}}(\xi)\mathrm{d}\xi,$$

where the likelihood $p(y_k|\xi) = p_{v_k}(y_k - h_k(\xi))$. Furthermore, all these integrals are positive because

$$p(y_k|\xi)p_{\bar{x}^-_{k,N}}(\xi) > 0 \text{ and } p(y_k|\xi)p_{x^-_{k,\text{exact}}}(\xi) > 0,$$

for all ξ. Respectively, because a set $\{x \,|\, x < z\}$ is open[6], we get that

$$\int_{-\infty}^{z} p(y_k|\xi)p_{\bar{x}^-_{k,N}}(\xi)\mathrm{d}\xi \underset{N\to\infty}{\longrightarrow} \int_{-\infty}^{z} p(y_k|\xi)p_{x^-_{k,\text{exact}}}(\xi)\mathrm{d}\xi,$$

for all z. Combining these results we get that

$$F_{\bar{x}^{+,\text{Bayes}}_k}(z) \underset{N\to\infty}{\longrightarrow} F_{x^+_{k,\text{exact}}}(z),$$

for all z, where

$$F_{\bar{x}^{+,\text{Bayes}}_k}(z) = \frac{\int_{-\infty}^{z} p(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi}{\int p(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi} \text{ and}$$
$$F_{x^+_{k,\text{exact}}}(z) = \frac{\int_{-\infty}^{z} p(y_k|\xi)p_{x^-_{k,\text{exact}}}(\xi)\mathrm{d}\xi}{\int p(y_k|\xi)p_{x^-_{k,\text{exact}}}(\xi)\mathrm{d}\xi}. \tag{15}$$

□

Theorem 13 (Bank of EKFs convergence): Let

$$F_{\bar{x}^+_k}(z) = \frac{\int_{-\infty}^{z} p_{\text{EKF}}(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi}{\int p_{\text{EKF}}(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi} \text{ and}$$
$$F_{\bar{x}^{+,\text{Bayes}}_k}(z) = \frac{\int_{-\infty}^{z} p(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi}{\int p(y_k|\xi)p_{\bar{x}^-_k}(\xi)\mathrm{d}\xi},$$

where the likelihood

$$p(y_k|\xi) = p_{v_k}(y_k - h_k(\xi))$$

and the bank of EKF likelihood approximations[7] (see Eq. (10))

$$p_{\text{EKF}}(y_k|\xi) = p_{v_k}(y_k - h_k(\bar{\mu}^-_{i,k}) - \mathrm{H}_{i,k}(\xi - \bar{\mu}^-_{i,k})).$$

Then

$$F_{\bar{x}^{+,\text{Bayes}}_k}(\xi) - F_{\bar{x}^+_k}(\xi) \underset{N\to\infty}{\longrightarrow} 0.$$

[5]Usually we suppress the index N (parameter of GMF), that is $x^+_{k-1,N} \triangleq x^+_{k-1}$.

[6]Here sign $''<''$ is interpreted elementwise.

[7]Note that current approximation is also a function of index i (see Eq. (10)).

Proof: It is enough to show that

$$\int \left| p(y_k|\xi) - p_{\text{EKF}}(y_k|\xi) \right| p_{\bar{x}_k^-}(\xi) \mathrm{d}\xi \underset{N\to\infty}{\longrightarrow} 0.$$

Now

$$\begin{aligned}
&\int \left| p(y_k|\xi) - p_{\text{EKF}}(y_k|\xi) \right| p_{\bar{x}_k^-}(\xi) \mathrm{d}\xi \\
&\leq \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \beta_{j,k} \bar{\alpha}_{i,k}^- \int \left| \mathrm{N}_{\mathrm{R}_{j,k}}^{\bar{v}_{j,k}}(z) - \mathrm{N}_{\mathrm{R}_{j,k}}^{\bar{v}_{j,k}}(\tilde{z}_i) \right| \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi) \mathrm{d}\xi \\
&= \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \frac{\beta_{j,k} \bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \mathrm{R}_{j,k}\right)}} \cdot \ldots \\
&\int \left| \exp\left(-\frac{1}{2} \|z\|^2_{\mathrm{R}_{j,k}^{-1}} \right) - \exp\left(-\frac{1}{2} \|\tilde{z}_i\|^2_{\mathrm{R}_{j,k}^{-1}} \right) \right| \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi) \mathrm{d}\xi \\
&= \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \frac{\beta_{j,k} \bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \mathrm{R}_{j,k}\right)}} \epsilon_{i,j},
\end{aligned}$$

where $z = y_k - h_k(\xi)$, $\tilde{z}_i = y_k - h_k(\bar{\mu}_{i,k}^-) - \mathrm{H}_{i,k}(\xi - \bar{\mu}_{i,k}^-)$ and $\epsilon_{i,j}$ is

$$\int \left| \exp\left(-\frac{1}{2} \|z\|^2_{\mathrm{R}_{j,k}^{-1}} \right) - \exp\left(-\frac{1}{2} \|\tilde{z}_i\|^2_{\mathrm{R}_{j,k}^{-1}} \right) \right| \mathrm{N}_{\bar{\Sigma}_{i,k}^-}^{\bar{\mu}_{i,k}^-}(\xi) \mathrm{d}\xi.$$

It is easy to see that

$$\epsilon_{i,j} < 1. \tag{16}$$

Based on the assumptions (see p. 4) we know that almost all $\bar{\mu}_{i,k}^-$ have a neighbourhood $\bar{C}_i$ such that

$$\left| \xi^T h_{kj}''(x) \xi \right| \leq c_{\mathrm{H}} \xi^T \begin{bmatrix} \mathrm{I}_{d\times d} & 0 \\ 0 & 0 \end{bmatrix} \xi, \text{ for all } \xi \in \mathbb{R}^{n_x} \tag{17}$$

where c_{H} is some constant, $j \in \{1, \ldots, n_y\}$, n_y is the number of measurements (length of vector y), d see p. 4 and $x \in \bar{C}_i$. We select $\bar{C}_i$ such that it is as big as possible (union of all possible sets). Especially we see that if $x \in \bar{C}_i$ then $\begin{bmatrix} x_{1:d} \\ \bar{x} \end{bmatrix} \in \bar{C}_i$, where $\bar{x} \in \mathbb{R}^{n_x - d}$ is an arbitrary vector.

The index set I_1 contains the index i if both inequalities (14) and (17) hold, the rest of the indices belong to the index set I_2. Now

$$\begin{aligned}
&\int |p(y_k|\xi) - p_{\text{EKF}}(y_k|\xi)| p_{\bar{x}_k^-}(\xi) \mathrm{d}\xi \\
&= \sum_{j=1}^{n_{v_k}} \sum_{i=1}^{\bar{n}_k^-} \frac{\beta_{j,k} \bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \mathrm{R}_{j,k}\right)}} \epsilon_{i,j}, \\
&\overset{(16)}{\leq} \sum_{j=1}^{n_{v_k}} \sum_{i\in I_1} \frac{\beta_{j,k} \bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \mathrm{R}_{j,k}\right)}} \epsilon_{i,j} + \sum_{i\in I_2} \frac{\bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \bar{\mathrm{R}}_k\right)}},
\end{aligned}$$

where $\det\left(2\pi \bar{\mathrm{R}}_k\right) = \min_j \det\left(2\pi \mathrm{R}_{j,k}\right)$. Since almost all indices belong to the index set I_1,

$$\sum_{i\in I_2} \frac{\bar{\alpha}_{i,k}^-}{\sqrt{\det\left(2\pi \bar{\mathrm{R}}_k\right)}} \underset{N\to\infty}{\longrightarrow} 0.$$

Appendix C (Lemma 27) shows that $\epsilon_{i,j} \underset{N\to\infty}{\longrightarrow} 0$ when $i \in I_1$. □

D. Convergence results of Step 4 (reduce the number of components)

Here we show that if $\bar{x}_k^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$ then $x_k^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$. First we show that if $\bar{x}_k^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$ then $\bar{x}_{k,\mathrm{f}}^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$, (Thm. 14), see Eq. (12).

Theorem 14 (Forgetting components): If

$$\bar{x}_k^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$$

then

$$\bar{x}_{k,\mathrm{f}}^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+.$$

(See Sec. III-D1.)

Proof: Take arbitrary $\epsilon > 0$, then there is an n_1 such that

$$|F_{x_{k,\text{exact}}^+}(\xi) - F_{\bar{x}_k^+}(\xi)| \leq \frac{\epsilon}{2},$$

for all ξ when $N > n_1$. Now

$$\begin{aligned}
&|F_{x_{k,\text{exact}}^+}(\xi) - F_{\bar{x}_{k,\mathrm{f}}^+}(\xi)| \\
&= |F_{x_{k,\text{exact}}^+}(\xi) - F_{\bar{x}_k^+}(\xi) + F_{\bar{x}_k^+}(\xi) - F_{\bar{x}_{k,\mathrm{f}}^+}(\xi)| \\
&\leq |F_{x_{k,\text{exact}}^+}(\xi) - F_{\bar{x}_k^+}(\xi)| + |F_{\bar{x}_k^+}(\xi) - F_{\bar{x}_{k,\mathrm{f}}^+}(\xi)| \\
&\leq \frac{\epsilon}{2} + \frac{1}{2N} \leq \epsilon,
\end{aligned}$$

for all $\xi \in \mathbb{R}^{n_x}$, when $N \geq \max(n_1, \frac{1}{\epsilon})$. This completes the proof (see Def. 8). □

Theorem 15 (Merging components): If

$$\bar{x}_{k,\mathrm{f}}^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+$$

then

$$x_k^+ \underset{N\to\infty}{\overset{\mathrm{w}}{\longrightarrow}} x_{k,\text{exact}}^+.$$

(See Sec. III-D2.)

Proof: Based on Thm. 14 it is enough to show that

$$|F_{\bar{x}_{k,\mathrm{f}}^+}(\xi) - F_{x_k^+}(\xi)| \underset{N\to\infty}{\longrightarrow} 0,$$

for all ξ. Because all cumulative density functions are continuous and

$$\begin{aligned}
\|\bar{\mu}_{i_1}^+ - \bar{\mu}_{i_1,\mathrm{m}}^+\| &\underset{N\to\infty}{\longrightarrow} 0, \\
\|\bar{\mu}_{i_2}^+ - \bar{\mu}_{i_1,\mathrm{m}}^+\| &\underset{N\to\infty}{\longrightarrow} 0, \\
\|\bar{\Sigma}_{i_1}^+ - \bar{\Sigma}_{i_1,\mathrm{m}}^+\| &\underset{N\to\infty}{\longrightarrow} 0 \text{ and} \\
\|\bar{\Sigma}_{i_2}^+ - \bar{\Sigma}_{i_1,\mathrm{m}}^+\| &\underset{N\to\infty}{\longrightarrow} 0
\end{aligned}$$

then

$$|F_{\bar{x}_{k,\mathrm{f}}^+}(\xi) - F_{x_k^+}(\xi)| \underset{N\to\infty}{\longrightarrow} 0.$$

This completes the proof (see Def. 8). □

V. Box Gaussian Mixture Approximation

In this section, we define the Box Gaussian Mixture Approximation (BGMA) and present some of its properties. Finally, we show that BGMA converges weakly to the original distribution.

Definition 16 (BGMA): The Box Gaussian Mixture Approximation of $x \sim \mathrm{N}_n(\mu, \Sigma)$, note $n \stackrel{\triangle}{=} n_x$, where $\Sigma > 0$, is

$$x_N \sim \mathrm{M}(\alpha_i, \mu_i, \Sigma_i)_{(i,(2N^2+1)^d)},$$

where the multi-index $i \in \mathbb{Z}^d$, with $d \leq n$ and $\|i\|_\infty \leq N^2$. The parameters are defined as

$$\begin{aligned} \alpha_i &= \int_{A_i} p_x(\xi)\mathrm{d}\xi, \\ \mu_i &= \int_{A_i} \xi \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi, \text{ and} \\ \Sigma_i &= \int_{A_i} (\xi-\mu_i)(\xi-\mu_i)^T \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi, \end{aligned} \tag{18}$$

where the sets

$$A_i = \left\{ x \middle| l(i) < \mathrm{A}(x-\mu) \leq u(i) \right\},$$

constitute a partition of $\mathbb{R}^n$. We assume that $\mathrm{A} = \left[\begin{array}{cc} \mathrm{A}_{11} & 0 \end{array}\right]$, $\mathrm{A}_{11} \in \mathbb{R}^{d\times d}$ and

$$\mathrm{A}\Sigma\mathrm{A}^T = \mathrm{I}. \tag{19}$$

Here the limits $l(i)$ and $u(i)$ are

$$\begin{aligned} l_j(i) &= \begin{cases} -\infty, & \text{if } i_j = -N^2 \\ \frac{i_j}{N} - \frac{1}{2N}, & \text{otherwise} \end{cases}, \\ u_j(i) &= \begin{cases} \infty, & \text{if } i_j = N^2 \\ \frac{i_j}{N} + \frac{1}{2N}, & \text{otherwise} \end{cases}. \end{aligned}$$

Now we show that the assumption Eq. (19) enables feasible computation time for the parameters of BGMA.

Theorem 17 (Parameters of BGMA): Let

$$x_N \sim \mathrm{M}(\alpha_i, \mu_i, \Sigma_i)_{(i,(2N^2+1)^d)},$$

be the BGMA of $x \sim \mathrm{N}_n(\mu, \Sigma)$, where $\Sigma > 0$ (see Def. 16). Then the parameters are

$$\begin{aligned} \alpha_i &= \prod_{j=1}^{d} \left(\Phi\left(u_j(i)\right) - \Phi\left(l_j(i)\right)\right), \\ \mu_i &= \mu + \Sigma\mathrm{A}^T\epsilon_i, \text{ and} \\ \Sigma_i &= \Sigma - \Sigma\mathrm{A}^T\Lambda_i\mathrm{A}\Sigma, \end{aligned}$$

where

$$\begin{aligned} \Phi(x) &= \int_{-\infty}^{x} N_1^0(\xi)\mathrm{d}\xi, \\ \Lambda_i &= \mathrm{diag}(\delta_i + \mathrm{diag}(\epsilon_i\epsilon_i^T)), \\ \epsilon_i &= \sum_{j=1}^{d} e_j \frac{e^{-\frac{1}{2}l_j(i)^2} - e^{-\frac{1}{2}u_j(i)^2}}{\sqrt{2\pi}\left(\Phi\left(u_j(i)\right) - \Phi\left(l_j(i)\right)\right)}, \text{ and} \\ \delta_i &= \sum_{j=1}^{d} e_j \frac{u_j(i)e^{-\frac{1}{2}u_j(i)^2} - l_j(i)e^{-\frac{1}{2}l_j(i)^2}}{\sqrt{2\pi}\left(\Phi\left(u_j(i)\right) - \Phi\left(l_j(i)\right)\right)}, \end{aligned}$$

where $e_j \in \mathbb{R}^d$ is the jth column of the identity matrix I. The sets A_i, and limits $l(i)$ and $u(i)$ are given in Def. 16.

Proof: We use the following block matrix notation

$$\Sigma = \left[\begin{array}{cc} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{array}\right], \quad \bar{\mathrm{A}} = \left[\begin{array}{cc} \mathrm{A}_{11} & 0 \\ -\mathrm{D}\Sigma_{21}\Sigma_{11}^{-1} & \mathrm{D} \end{array}\right],$$

where $\mathrm{D} = \left(\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right)^{-\frac{1}{2}}$. Because $\Sigma > 0$ then $\mathrm{D} > 0$. We see that $\bar{\mathrm{A}}\Sigma\bar{\mathrm{A}}^T = \mathrm{I}$. We use the variable transformation

$$\bar{x} = \bar{\mathrm{A}}(x-\mu).$$

Because $x \sim \mathrm{N}(\mu, \Sigma)$ then $\bar{x} \sim \mathrm{N}(0, \mathrm{I})$, and if $x \in A_i$ then $\bar{x} \in B_i$ and vice versa. Here

$$B_i = \left\{ \bar{x} \middle| \left[\begin{array}{c} l(i) \\ -\infty \end{array}\right] < \bar{x} \leq \left[\begin{array}{c} u(i) \\ \infty \end{array}\right] \right\}.$$

Now we compute parameters Eq. (18)

$$\begin{aligned} \alpha_i &= \mathrm{P}(\bar{x} \in B_i) = \prod_{j=1}^{d} \left(\Phi\left(u_j(i)\right) - \Phi\left(l_j(i)\right)\right), \\ \mu_i &= \mu + \int_{B_i} \bar{\mathrm{A}}^{-1}\eta \frac{p_{\bar{x}}(\eta)}{\alpha_i}\mathrm{d}\eta = \mu + \Sigma\mathrm{A}^T\epsilon_i, \\ \Sigma_i &= \bar{\mathrm{A}}^{-1} \int_{B_i} (\eta-\epsilon_i)(\eta-\epsilon_i)^T \frac{p_{\bar{x}}(\eta)}{\alpha_i}\mathrm{d}\eta\bar{\mathrm{A}}^{-T} \\ &= \Sigma - \Sigma\mathrm{A}^T\Lambda_i\mathrm{A}\Sigma. \end{aligned}$$

Here we have used the knowledge that $\bar{\mathrm{A}}^{-1} = \Sigma\bar{\mathrm{A}}^T$. □

In Fig. 1, we compare the density function of the Gaussian distribution

$$x \sim \mathrm{N}_2\left(\left[\begin{array}{c} 0 \\ 0 \end{array}\right], \left[\begin{array}{cc} 13 & 12 \\ 12 & 13 \end{array}\right]\right)$$

and the density function of its BGMA with parameters $d = 2$, $N = 2$ and

$$\mathrm{A} = \left[\begin{array}{cc} \frac{1}{\sqrt{13}} & 0 \\ -\frac{12}{5\sqrt{13}} & \frac{\sqrt{13}}{5} \end{array}\right].$$

Fig. 1 shows the contour plots of the Gaussian and the BGMA density functions such that 50% of the probability mass is inside the innermost curve and 95% of the probability mass is inside the outermost curve.

Theorem 18 shows that BGMA has the same mean and covariance as the original distribution.

Theorem 18 (Mean and Covariance of BGMA): Let

$$x_N \sim \mathrm{M}(\alpha_i, \mu_i, \Sigma_i)_{(i,(2N^2+1)^d)},$$

be the BGMA of $x \sim \mathrm{N}_n(\mu, \Sigma)$, where $\Sigma > 0$ (see Def. 16). Then

$$\mathrm{E}(x_N) = \mu \quad \text{and} \quad \mathrm{V}(x_N) = \Sigma.$$

Proof: Now

$$\begin{aligned} \mathrm{E}(x_N) &\stackrel{\text{Thm. 5}}{=} \sum_i \alpha_i\mu_i \stackrel{\text{Def. 16}}{=} \sum_i \alpha_i \int_{A_i} \xi \frac{p_x(\xi)}{\alpha_i} d\xi \\ &= \sum_i \int_{A_i} \xi p_x(\xi) d\xi = \int \xi p_x(\xi)\mathrm{d}\xi = \mu, \end{aligned}$$

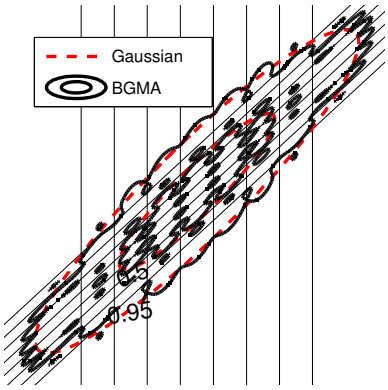

Fig. 1. Example of the BGMA

and

$$\begin{aligned}
\mathrm{V}(x_N) &\overset{\text{Thm. 5}}{=} \sum_i \alpha_i \left(\Sigma_i + (\mu_i-\mu)(\mu_i-\mu)^T\right)\\
&= \sum_i \alpha_i \left(\Sigma_i + \mu_i(\mu_i-\mu)^T\right)\\
&\overset{\text{Def. 16}}{=} \sum_i \alpha_i \left(\int_{A_i} \xi(\xi-\mu_i)^T \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi + \mu_i(\mu_i-\mu)^T\right)\\
&\overset{\text{Def. 16}}{=} \sum_i \alpha_i \int_{A_i} \xi\xi^T \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi - \mu\mu^T\\
&= \int \xi\xi^T p_x(\xi)\mathrm{d}\xi - \mu\mu^T = \Sigma.
\end{aligned}$$

□

Lemma 19 considers bounded boxes of Def. 16, i.e. boxes with parameters $d = n$ and $\|i\|_\infty < N^2$. Lemma 19 shows that a ball with radius $r_{\text{in}} = \frac{1}{2N\|\mathrm{A}\|}$ fits inside all boxes, and all boxes fit inside a ball whose radius is $r_{\text{out}} = \frac{\sqrt{n}}{2N}\|\mathrm{A}^{-1}\|$. Note that the proportion of these radiuses $\frac{r_{\text{in}}}{r_{\text{out}}} = \frac{1}{\sqrt{n}\kappa(\mathrm{A})}$ does not depend on the parameter N. Here $\kappa(\mathrm{A}) = \|\mathrm{A}\|\|\mathrm{A}^{-1}\|$ is the condition number of matrix A.

Lemma 19: Let

$$\begin{aligned}
A &= \left\{x \middle| -\frac{1}{2N}\mathbf{1} < \mathrm{A}x \le \frac{1}{2N}\mathbf{1}\right\},\\
R_{\text{in}} &= \left\{x \middle| \|x\| \le r_{\text{in}}\right\} \text{ and } R_{\text{out}} = \left\{x \middle| \|x\| \le r_{\text{out}}\right\},
\end{aligned}$$

where $\mathbf{1}$ is a vector allof whose elements are ones,

$$r_{\text{in}} = \frac{1}{2N\|\mathrm{A}\|}, \quad r_{\text{out}} = \frac{\sqrt{n}}{2N}\|\mathrm{A}^{-1}\|$$

and A is non-singular. Now $R_{\text{in}} \subset A \subset R_{\text{out}}$.

Proof: If $x \in R_{\text{in}}$ then

$$\|\mathrm{A}x\| \le \|\mathrm{A}\|\|x\| \le \frac{1}{2N}.$$

So $R_{\text{in}} \subset A$. If $x \in A$ then

$$\|x\| \le \|\mathrm{A}x\|\|\mathrm{A}^{-1}\| \le \|\frac{1}{2N}\mathbf{1}\|\|\mathrm{A}^{-1}\| \le \frac{\sqrt{n}}{2N}\|\mathrm{A}^{-1}\|.$$

So $A \subset R_{\text{out}}$. □

Corollary 20 considers the center boxes ($\|i\|_\infty < N^2$) of BGMA (Def. 16) and shows that the covariances of the first d dimensions converge to zero when N approaches infinity.

Corollary 20: Covariances Σ_i are the same as in Def. 16. Now

$$\sum_{j=1}^{d} \Sigma_{ij,j} \underset{N\to\infty}{\longrightarrow} 0, \text{ when } \|i\|_\infty < N^2.$$

Proof: Because

$$\begin{aligned}
\sum_{j=1}^{d} \Sigma_{ij,j} &= \int_{A_i} \|\xi_{1:d} - \mu_{i\,1:d}\|^2 \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi\\
&\overset{\text{Lem. 19}}{\le} \int_{A_i} \frac{d}{N^2}\|\mathrm{A}_{11}^{-1}\|^2 \frac{p_x(\xi)}{\alpha_i}\mathrm{d}\xi\\
&= \frac{d}{N^2}\|\mathrm{A}_{11}^{-1}\|^2
\end{aligned}$$

then $\sum_{j=1}^{d} \Sigma_{ij,j} \underset{N\to\infty}{\longrightarrow} 0$, for all $\|i\|_\infty < N^2$. □

Theorem 26 (see Appendix B) shows that the BGMA converges weakly to the original distribution when the center boxes are bounded. Theorem 21 uses this result to show that BGMA converges weakly to the original distribution even if all boxes are unbounded.

Theorem 21 (BGMA convergence, Gaussian case): Let

$$x_N \sim \mathrm{M}(\alpha_i, \mu_i, \Sigma_i)_{(i,(2N^2+1)^d)}$$

be BGMA of $x \sim \mathrm{N}_n(\mu, \Sigma)$, where $\Sigma > 0$ (see Def. 16). Now

$$x_N \underset{N\to\infty}{\overset{\text{w}}{\longrightarrow}} x.$$

Proof: First we define new random variables

$$\begin{aligned}
&\bar{x} = \bar{\mathrm{A}}(x-\mu) \sim \mathrm{N}(0, \mathrm{I}) \quad \text{and}\\
&\bar{x}_N = \bar{\mathrm{A}}(x_N - \mu) \sim \mathrm{M}(\alpha_i, \bar{\mathrm{A}}(\mu_i - \mu), \bar{\mathrm{A}}\Sigma_i\bar{\mathrm{A}}^T)_{(i,(2N^2+1)^d)},
\end{aligned}$$

where $\bar{\mathrm{A}}$ is defined in Thm. 17. Note that $\bar{\mathrm{A}}\Sigma_i\bar{\mathrm{A}}^T$ are diagonal matrices. It is enough to show that (because of Slutsky's Theorem 10)

$$\bar{x}_N \underset{N\to\infty}{\overset{\text{w}}{\longrightarrow}} \bar{x}.$$

Let F_N and F be the cumulative density functions corresponding to the random variables $\bar{x}_N$ and $\bar{x}$. We have to show that

$$F_N(\bar{x}) \underset{N\to\infty}{\longrightarrow} F(\bar{x}), \quad \forall \bar{x} \in \mathbb{R}^n. \tag{20}$$

Because

$$\begin{aligned}
F_N(\bar{x}) &= \sum_i \alpha_i \int_{-\infty}^{\bar{x}_1} \mathrm{N}_{\mathrm{I}-\Lambda_i}^{\epsilon_i}(\eta_1)\mathrm{d}\eta_1 \int_{-\infty}^{\bar{x}_2} \mathrm{N}_{\mathrm{I}}^{0}(\eta_2)\mathrm{d}\eta_2\\
&= G_N(\bar{x}_1)\int_{-\infty}^{\bar{x}_2} \mathrm{N}_{\mathrm{I}}^{0}(\eta_2)\mathrm{d}\eta_2,\\
F(\bar{x}) &= \int_{-\infty}^{\bar{x}_1} \mathrm{N}_{\mathrm{I}}^{0}(\eta_1)\mathrm{d}\eta_1 \int_{-\infty}^{\bar{x}_2} \mathrm{N}_{\mathrm{I}}^{0}(\eta_2)\mathrm{d}\eta_2\\
&= G(\bar{x}_1)\int_{-\infty}^{\bar{x}_2} \mathrm{N}_{\mathrm{I}}^{0}(\eta_2)\mathrm{d}\eta_2,
\end{aligned}$$

where $\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix}$, it is enough to show that

$$\bar{x}_{N_{1:d}} \underset{N\to\infty}{\overset{\text{w}}{\longrightarrow}} \bar{x}_{1:d} \tag{21}$$

Based on Thm. 26 (see Appendix B), Eq. (21) is true, which implies the theorem. □

Corollary 22 (BGMA convergence, GM case): Let

$$\bar{x}_{j,N} \sim \mathrm{M}(\bar{\alpha}_i, \bar{\mu}_i, \bar{\Sigma}_i)_{(i,(2N^2+1)^d)}$$

be the BGMA of $x_j \sim \mathrm{N}_n(\mu_j, \Sigma_j)$, where $\Sigma_j > 0$ (see Def. 16). Let x be the GM whose density function is

$$p_x(\xi) = \sum_{j=i}^{N_x} \alpha_j \, \mathrm{N}_{\Sigma_j}^{\mu_j}(\xi).$$

and x_N be the GM whose density function is

$$p_{x_N}(\xi) = \sum_{j=i}^{N_x} \alpha_j p_{\bar{x}_{j,N}}(\xi).$$

Now

$$x_N \xrightarrow[N\to\infty]{\mathrm{w}} x.$$

Proof: Take arbitrary $\epsilon > 0$, then there are n_j, $j = 1, \ldots, N_x$ such that (Thm. 21)

$$|F_{x_j}(\xi) - F_{\bar{x}_{j,N_j}}(\xi)| \leq \epsilon, \tag{22}$$

for all j and ξ, when $N_j > n_j$. Now

$$\begin{aligned} |F_x(\xi) - F_{x_N}(\xi)| &= \Big| \sum_{j=1}^{N_x} \alpha_j \left(F_{x_j}(\xi) - F_{x_{j,N}}(\xi) \right) \Big| \\ &\leq \sum_{j=1}^{N_x} \alpha_j |F_{x_j}(\xi) - F_{x_{j,N}}(\xi)| \\ &\overset{(22)}{\leq} \sum_{j=1}^{N_x} \alpha_j \epsilon = \epsilon, \end{aligned}$$

for all ξ, when $N > \max_j\{n_j\}$. □

VI. Box Gaussian Mixture Filter

The Box Gaussian Mixture Filter (BGMF) is a GMF (Sec. III) that approximates the prior x_k^- as a new GM (Step 2 in Algorithm 2) using BGMA (Sec. V) separately for each component of the prior. Section IV shows that BGMF converges weakly to the exact posterior distribution.

VII. Simulations

In the simulations we consider only the case of a single time step. Our state $x = \begin{bmatrix} r_\mathrm{u} \\ v_\mathrm{u} \end{bmatrix}$ consists of the 2D-position vector r_u and the 2D-velocity vector v_u of the user. The prior distribution is

$$x \sim \mathrm{N}\left(\begin{bmatrix} 100 \\ 10 \\ 10 \\ 10 \end{bmatrix}, \begin{bmatrix} 90000 & 0 & 0 & 0 \\ 0 & 7500 & 0 & 2500 \\ 0 & 0 & 1000 & 0 \\ 0 & 2500 & 0 & 7500 \end{bmatrix} \right),$$

and the current measurement (see Eq. (8)) is

$$\begin{bmatrix} 500 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \|r_\mathrm{u}\| \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x + v,$$

where v is independent of x and

$$v \sim \mathrm{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 10^4 & 0 & 0 & 0 \\ 0 & 10^3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \right).$$

So now $d = 2$ and $n = 4$ (see Def. 16). The current posterior of the 2D-position is shown in Fig. 2. We see that the posterior distribution is multimodal.

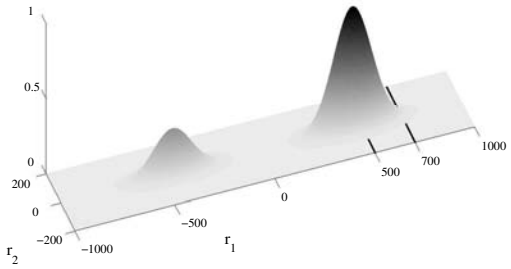

Fig. 2. The posterior of the position.

Now we compute the posterior approximations using BGMF (Algorithm 2 steps (2) and (3)) and a Particle Filter [18]. BGMF uses BGMA with parameters $N \in \{0, 1, 2, \ldots, 9\}$; the corresponding numbers of posterior mixture components are (Def. 16)

$$n_{\mathrm{BGMF}} \in \{1, 9, 81, \ldots, 26569\}.$$

The numbers of particles in the Particle Filter are

$$n_{\mathrm{PF}} \in \left\{2^2 \cdot 100, 2^3 \cdot 100, \ldots, 2^{17} \cdot 100\right\}.$$

We compute error statistics

$$|\mathrm{P}\left(x_{\text{true}} \in C\right) - \mathrm{P}\left(x_{\text{app.}} \in C\right)|,$$

where the set $C = \left\{ x \,\middle|\, |e_1^T x - 600| \leq 100 \right\}$ (see Fig. 2). We know that $\mathrm{P}\left(x_{\text{true}} \in C\right) \approx 0.239202$. These error statistics are shown as a function of CPU time in Fig. 3. Thm. 9 shows that these error statistics converge to zero when the posterior approximation converges weakly to the correct posterior.

Fig. 3 is consistent with the convergence results. It seems that the error statistics of both BGMF and PF converge to zero when the number of components or particles increase. We also see that in this case $2^{10} \cdot 100 \approx$ 1e5 particles in PF are definitely too few. However, BGMF gives promising results with only 81 components ($N = 2$) when CPU time is significantly less than one second, which is good considering real time implementations.

VIII. Conclusion

In this paper, we have presented the Box Gaussian Mixture Filter (BGMF), which is based on Box Gaussian Mixture Approximation (BGMA). We have presented the general form

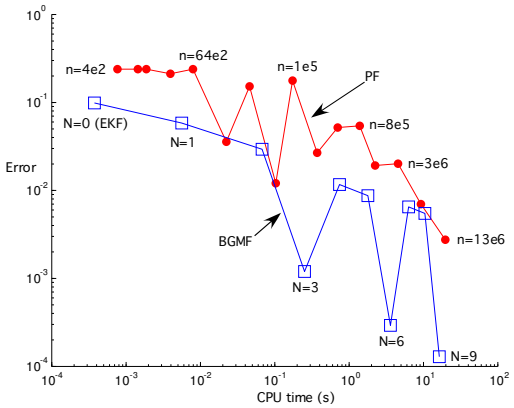

Fig. 3. Simulation results of BGMF and PF.

of Gaussian Mixture Filters (GMF) and we have shown that GMF converges weakly to the correct posterior at given time instant. BGMF is a GMF so we have shown that BGMF converges weakly to the correct posterior at given time instant. We have illustrated this convergence result with a tiny example in which BGMF outperforms a basic particle filter. The assumptions of BGMF fit very well in positioning and previous work [27] shows that BGMF is feasible for real time positioning implementations. BGMF also has smaller computational and memory requirements than conventional GMFs, because it splits only those dimensions where we get nonlinear measurements.

Appendix A
Product of two normal densities.

The aim of this appendix is to compute the product of two Gaussian densities (Theorem 25).

Lemma 23: If $\mathrm{A} > 0$ (s.p.d.) then

$$\|a \pm \mathrm{A}^{-1}b\|^2_{\mathrm{A}} = a^T\mathrm{A}a \pm 2b^Ta + b^T\mathrm{A}^{-1}b.$$

Proof:

$$\begin{aligned}\|a \pm \mathrm{A}^{-1}b\|^2_{\mathrm{A}} &= \left(a \pm \mathrm{A}^{-1}b\right)^T \mathrm{A}\left(a \pm \mathrm{A}^{-1}b\right)\\ &= a^T\mathrm{A}a \pm a^Tb \pm b^Ta + b^T\mathrm{A}^{-1}b\\ &= a^T\mathrm{A}a \pm 2b^Ta + b^T\mathrm{A}^{-1}b\end{aligned}$$

□

Lemma 24: If $\Sigma_1, \Sigma_2 > 0$ then

$$\|x-\mu\|^2_{\Sigma_1^{-1}} + \|y-\mathrm{H}x\|^2_{\Sigma_2^{-1}} = \|x-\bar{\mu}\|^2_{\Sigma_3^{-1}} + \|y-\mathrm{H}\mu\|^2_{\Sigma_4^{-1}},$$

where

$$\begin{aligned}\bar{\mu} &= \mu + \mathrm{K}(y-\mathrm{H}\mu),\\ \Sigma_3 &= (\mathrm{I}-\mathrm{K}\mathrm{H})\Sigma_1,\\ \mathrm{K} &= \Sigma_1\mathrm{H}^T\Sigma_4^{-1}, \text{ and}\\ \Sigma_4 &= \mathrm{H}\Sigma_1\mathrm{H}^T + \Sigma_2.\end{aligned}$$

Proof:

$$\begin{aligned}&\|x-\mu\|^2_{\Sigma_1^{-1}} + \|y-\mathrm{H}x\|^2_{\Sigma_2^{-1}}\\ &\overset{\text{Lem. 23}}{=} x^T(\Sigma_1^{-1} + \mathrm{H}^T\Sigma_2^{-1}\mathrm{H})x - 2(\mu^T\Sigma_1^{-1} + y^T\Sigma_2^{-1}\mathrm{H})x\\ &\quad + \mu^T\Sigma_1^{-1}\mu + y^T\Sigma_2^{-1}y\\ &\overset{\text{Lem. 23}}{=} \|x-\mathrm{C}c\|^2_{\mathrm{C}^{-1}} - c^T\mathrm{C}c + \mu^T\Sigma_1^{-1}\mu + y^T\Sigma_2^{-1}y\\ &= \|x-\bar{\mu}\|^2_{\Sigma_3^{-1}} + \|y-\mathrm{H}y\|^2_{\Sigma_4^{-1}},\end{aligned}$$

where $c = \Sigma_1^{-1}\mu + \mathrm{H}^T\Sigma_2^{-1}y$,

$$\begin{aligned}\mathrm{C} &= (\Sigma_1^{-1} + \mathrm{H}^T\Sigma_2^{-1}\mathrm{H})^{-1}\\ &\overset{\star_1}{=} \Sigma_1 - \Sigma_1\mathrm{H}^T(\mathrm{H}\Sigma_1\mathrm{H}^T + \Sigma_2)^{-1}\mathrm{H}\Sigma_1\\ &= (\mathrm{I}-\mathrm{K}\mathrm{H})\Sigma_1 = \Sigma_3,\end{aligned}$$

in step $\star_1$ we use the matrix inversion lemma [34, p.729].

$$\begin{aligned}\mathrm{C}c &= (\mathrm{I}-\mathrm{K}\mathrm{H})\Sigma_1(\Sigma_1^{-1}\mu + \mathrm{H}^T\Sigma_2^{-1}y)\\ &= (\mathrm{I}-\mathrm{K}\mathrm{H})\mu + (\mathrm{I}-\mathrm{K}\mathrm{H})\Sigma_1\mathrm{H}^T\Sigma_2^{-1}y\\ &= (\mathrm{I}-\mathrm{K}\mathrm{H})\mu + \Sigma_1\mathrm{H}^T\Sigma_2^{-1}y - \mathrm{K}(\Sigma_4-\Sigma_2)\Sigma_2^{-1}y\\ &= (\mathrm{I}-\mathrm{K}\mathrm{H})\mu + \Sigma_1\mathrm{H}^T\Sigma_2^{-1}y - \Sigma_1\mathrm{H}^T\Sigma_2^{-1}y + \mathrm{K}y\\ &= \mu + \mathrm{K}(y-\mathrm{H}\mu) = \bar{\mu}\end{aligned}$$

and

$$\begin{aligned}c^T\mathrm{C}c &= (\Sigma_1^{-1}\mu + \mathrm{H}^T\Sigma_2^{-1}y)^T((\mathrm{I}-\mathrm{K}\mathrm{H})\mu + \mathrm{K}y)\\ &= (\mu^T\Sigma_1^{-1} + y^T\Sigma_2^{-1}\mathrm{H})((\mathrm{I}-\mathrm{K}\mathrm{H})\mu + \mathrm{K}y)\\ &= \mu^T(\Sigma_1^{-1} - \mathrm{H}^T\Sigma_4^{-1}\mathrm{H})\mu + \mu^T\mathrm{H}^T\Sigma_4^{-1}y\ldots\\ &\quad + y^T(\Sigma_2^{-1}\mathrm{H} - \Sigma_2^{-1}\mathrm{H}\mathrm{K}\mathrm{H})\mu + y^T\Sigma_2^{-1}\mathrm{H}\mathrm{K}y\\ &\overset{(23)}{=} \mu^T(\Sigma_1^{-1} - \mathrm{H}^T\Sigma_4^{-1}\mathrm{H})\mu + \mu^T\mathrm{H}^T\Sigma_4^{-1}y\ldots\\ &\quad + y^T\Sigma_4^{-1}\mathrm{H}\mu + y^T(\Sigma_2^{-1} - \Sigma_4^{-1})y\\ &= -(y^T\Sigma_4^{-1}y - 2\mu^T\mathrm{H}^T\Sigma_4^{-1}y + \mu^T\mathrm{H}^T\Sigma_4^{-1}\mathrm{H}\mu)\ldots\\ &\quad + y^T\Sigma_2^{-1}y + \mu^T\Sigma_1^{-1}\mu\\ &= -\|y-\mathrm{H}y\|^2_{\Sigma_4^{-1}} + y^T\Sigma_2^{-1}y + \mu^T\Sigma_1^{-1}\mu.\end{aligned}$$

$$\Sigma_2^{-1}\mathrm{H}\mathrm{K} = \Sigma_2^{-1} - \Sigma_4^{-1} \tag{23}$$

□

Theorem 25 (Product of two Gaussians): If $\Sigma_1, \Sigma_2 > 0$ then

$$\mathrm{N}^{\mu}_{\Sigma_1}(x)\,\mathrm{N}^{\mathrm{H}x}_{\Sigma_2}(y) = \mathrm{N}^{\bar{\mu}}_{\Sigma_3}(x)\,\mathrm{N}^{\mathrm{H}\mu}_{\Sigma_4}(y),$$

where

$$\begin{aligned}\bar{\mu} &= \mu + \mathrm{K}(y-\mathrm{H}\mu),\\ \Sigma_3 &= (\mathrm{I}-\mathrm{K}\mathrm{H})\Sigma_1,\\ \mathrm{K} &= \Sigma_1\mathrm{H}^T\Sigma_4^{-1}, \text{ and}\\ \Sigma_4 &= \mathrm{H}\Sigma_1\mathrm{H}^T + \Sigma_2.\end{aligned}$$

Proof:

$$
\begin{aligned}
&\mathrm{N}^{\mu}_{\Sigma_1}(x)\,\mathrm{N}^{\mathrm{H}x}_{\Sigma_2}(y)\\
&= \frac{\exp\left(-\frac{1}{2}\|x-\mu\|^2_{\Sigma_1^{-1}}\right)}{(2\pi)^{\frac{n_x}{2}}\sqrt{\det(\Sigma_1)}}\frac{\exp\left(-\frac{1}{2}\|y-\mathrm{H}x\|^2_{\Sigma_2^{-1}}\right)}{(2\pi)^{\frac{n_y}{2}}\sqrt{\det(\Sigma_2)}}\\
&\overset{\text{Lem. 24}}{=} \frac{\exp\left(-\frac{1}{2}\|x-\bar{\mu}\|^2_{\Sigma_3^{-1}}\right)\exp\left(-\frac{1}{2}\|y-\mathrm{H}\mu\|^2_{\Sigma_4^{-1}}\right)}{(2\pi)^{\frac{n_x}{2}}(2\pi)^{\frac{n_y}{2}}\sqrt{\det(\Sigma_1)\det(\Sigma_2)}}\\
&\overset{(24)}{=} \frac{\exp\left(-\frac{1}{2}\|x-\bar{\mu}\|^2_{\Sigma_3^{-1}}\right)}{(2\pi)^{\frac{n_x}{2}}\sqrt{\det(\Sigma_3)}}\frac{\exp\left(-\frac{1}{2}\|y-\mathrm{H}\mu\|^2_{\Sigma_4^{-1}}\right)}{(2\pi)^{\frac{n_y}{2}}\sqrt{\det(\Sigma_4)}}\\
&= \mathrm{N}^{\bar{\mu}}_{\Sigma_3}(x)\,\mathrm{N}^{\mathrm{H}\mu}_{\Sigma_4}(y)
\end{aligned}
$$

where n_x and n_y are dimension of x and y, respectively.

$$
\begin{aligned}
&\det(\Sigma_1)\det(\Sigma_2)\\
&= \det\left(\begin{bmatrix}\Sigma_1 & 0\\ 0 & \Sigma_2\end{bmatrix}\right)\\
&= \det\left(\begin{bmatrix}\mathrm{I} & 0\\ \mathrm{H} & \mathrm{I}\end{bmatrix}\begin{bmatrix}\Sigma_1 & 0\\ 0 & \Sigma_2\end{bmatrix}\begin{bmatrix}\mathrm{I} & \mathrm{H}^T\\ 0 & \mathrm{I}\end{bmatrix}\right)\\
&= \det\left(\begin{bmatrix}\Sigma_1 & \Sigma_1\mathrm{H}^T\\ \mathrm{H}\Sigma_1 & \mathrm{H}\Sigma_1\mathrm{H}^T+\Sigma_2\end{bmatrix}\right)\\
&= \det\left(\begin{bmatrix}\mathrm{I} & \mathrm{K}\\ 0 & \mathrm{I}\end{bmatrix}\begin{bmatrix}\Sigma_3 & 0\\ \mathrm{H}\Sigma_1 & \Sigma_4\end{bmatrix}\right)\\
&= \det(\Sigma_3)\det(\Sigma_4)
\end{aligned}
\tag{24}
$$

□

Appendix B
BGMA convergence when $d = n$

Theorem 26 (BGMA convergence when $d = n$): Let

$$x_N \sim \mathrm{M}(\alpha_i, \mu_i, \Sigma_i)_{(i,(2N^2+1)^d)}$$

be the BGMA of $x \sim \mathrm{N}_n(\mu, \Sigma)$, where $\Sigma > 0$ (see Def. 16). We assume that $d = n$. Now

$$x_N \xrightarrow[N\to\infty]{\mathrm{w}} x.$$

Proof: Let F_N and F be the cumulative density functions corresponding to the random variables x_N and x. We have to show that

$$F_N(x) \underset{N\to\infty}{\longrightarrow} F(x), \quad \forall x \in \mathbb{R}^n. \tag{25}$$

Let $x \in \mathbb{R}^n$ be an arbitrary vector whose components are x_j, and define the index sets

$$
\begin{aligned}
IN(x) &= \{i | \mu_i \le x \text{ and } \|\mathrm{i}\|_\infty < \mathrm{N}^2\},\\
OUT(x) &= \{i | \mu_i \not\le x \text{ and } \|\mathrm{i}\|_\infty < \mathrm{N}^2\},\\
OUT_j(x) &= \{i | (\mu_i)_j > x_j \text{ and } \|\mathrm{i}\|_\infty < \mathrm{N}^2\} \text{ and}\\
OUT^l_j(x) &=\\
&\{i | l r_{\text{in}} < (\mu_i)_j - x_j \le (l+1) r_{\text{in}} \text{ and } \|\mathrm{i}\|_\infty < \mathrm{N}^2\},
\end{aligned}
$$

where $r_{\text{in}} = \frac{1}{2N\|\mathrm{A}\|}$ and less than or equal sign $''\le''$ is interpreted elementwise. First we show that

$$\sum_{i\in IN(x)} \alpha_i \to F(x). \tag{26}$$

Now

$$F(x - 2r_{\text{out}}\mathbf{1}) - \epsilon_{\text{edge}} \le \sum_{j\in IN(x)} \alpha_j \le F(x + 2r_{\text{out}}\mathbf{1}),$$

where $r_{\text{out}} = \frac{\sqrt{n}}{2N}\|\mathrm{A}^{-1}\|$ and

$$\epsilon_{\text{edge}} = \mathrm{P}\left(x \in \bigcup_{\|i\|_\infty = N^2} A_i\right).$$

We see that $r_{\text{out}} \underset{N\to\infty}{\longrightarrow} 0$ and $\epsilon_{\text{edge}} \underset{N\to\infty}{\longrightarrow} 0$. Using these results and the continuity of the cumulative density function $F(x)$ we see that equation (26) holds. So equation (25) holds if

$$\epsilon_N(x) = F_N(x) - \sum_{i\in IN(x)} \alpha_i \underset{N\to\infty}{\longrightarrow} 0, \quad \forall x \in \mathbb{R}^n. \tag{27}$$

Now we show that this equation (27) holds. We find upper and lower bounds of $\epsilon_N(x)$. The upper bound of $\epsilon_N(x)$ is

$$
\begin{aligned}
\epsilon_N(x) &= \int_{\xi\le x} p_{x_N}(\xi)\mathrm{d}\xi - \sum_{i\in IN(x)} \alpha_i\\
&\le \epsilon_{\text{edge}} + \int_{\xi\le x} \sum_{i\in OUT(x)} \alpha_i\, \mathrm{N}^{\mu_i}_{\Sigma_i}(\xi)\mathrm{d}\xi\\
&\le \epsilon_{\text{edge}} + \sum_{j=1}^{n}\int_{\xi_j\le x_j} \sum_{i\in OUT_j(x)} \alpha_i\, \mathrm{N}^{\mu_i}_{\Sigma_i}(\xi)\mathrm{d}\xi\\
&= \epsilon_{\text{edge}} + n\sum_{l=0}^{\infty}\int_{\xi_j\le x_j} \sum_{i\in OUT^l_j(x)} \alpha_i\, \mathrm{N}^{\mu_i}_{\Sigma_i}(\xi)\mathrm{d}\xi\\
&\le \epsilon_{\text{edge}} + n\sum_{l=0}^{\infty}\int_{\xi_j\le x_j} \sum_{i\in OUT^l_j(x)} \alpha_i\, \mathrm{N}^{x_j + l r_{\text{in}}}_{r^2_{\text{out}}}(\xi_j)\mathrm{d}\xi_j\\
&\le \epsilon_{\text{edge}} + n\sum_{l=0}^{\infty}\int_{y\le 0} \mathrm{N}_1^{l\frac{r_{\text{in}}}{r_{\text{out}}}}(y)\mathrm{d}y\alpha_{\max}\\
&\overset{(29)}{\le} \epsilon_{\text{edge}} + n\left(\frac{\kappa(\mathrm{A})\sqrt{2\pi n}+2}{4}\right)\alpha_{\max},
\end{aligned}
\tag{28}
$$

where

$$
\begin{aligned}
\alpha_{\max} &= \sup_{j,l}\left(\sum_{i\in OUT^l_j(x)} \alpha_i\right)\\
&\le \sup_{j,c\in\mathbb{R}} \left(\mathrm{P}\left(|x_j - c| \le 4r_{\text{out}}\right)\right).
\end{aligned}
$$

and

$$\begin{aligned}
&\sum_{l=0}^{\infty}\int_{y\le 0}\mathrm{N}_1^{l\frac{r_{\text{in}}}{r_{\text{out}}}}(y)\mathrm{d}y\\
&=\sum_{l=0}^{\infty}\int_{y\le 0}\frac{\exp\left(-\frac{y^2}{2}+l\frac{r_{\text{in}}}{r_{\text{out}}}y-\frac{1}{2}\left(l\frac{r_{\text{in}}}{r_{\text{out}}}\right)^2\right)}{\sqrt{2\pi}}\mathrm{d}y\\
&\le\sum_{l=0}^{\infty}\exp\left(-\frac{1}{2}\left(l\frac{r_{\text{in}}}{r_{\text{out}}}\right)^2\right)\int_{y\le 0}\frac{\exp\left(-\frac{y^2}{2}\right)}{\sqrt{2\pi}}\mathrm{d}y\\
&=\frac{1}{2}\sum_{l=0}^{\infty}\exp\left(-\frac{1}{2}\left(l\frac{r_{\text{in}}}{r_{\text{out}}}\right)^2\right)\\
&\le\frac{1}{2}\left(1+\int_0^{\infty}\exp\left(-\frac{1}{2}\left(l\frac{r_{\text{in}}}{r_{\text{out}}}\right)^2\right)\mathrm{d}l\right)\\
&\overset{\text{Lem. 19}}{=}\left(\frac{\kappa(\mathrm{A})\sqrt{2\pi n}+2}{4}\right).
\end{aligned}\tag{29}$$

The lower bound of $\epsilon_N(x)$ is

$$\begin{aligned}
\epsilon_N(x)&=\int_{\xi\le x}p_{x_k}(\xi)\mathrm{d}\xi-\sum_{i\in IN(x)}\alpha_i\\
&\ge-\int_{\xi\not\le x}\sum_{i\in IN(x)}\alpha_i\,\mathrm{N}_{\Sigma_i}^{\mu_i}(\xi)\mathrm{d}\xi\\
&\ge-\sum_{j=1}^{n}\int_{\xi_j>x_j}\sum_{i\in IN(x)}\alpha_i\,\mathrm{N}_{\Sigma_i}^{\mu_i}(\xi)\mathrm{d}\xi\\
&\ge-n\sum_{l=-\infty}^{-1}\int_{\xi_j>x_j}\mathrm{N}_{r_{\text{out}}^2}^{x_j+(l+1)r_{\text{in}}}(\xi_j)\mathrm{d}\xi_j\alpha_{\max}\\
&\overset{(29)}{\ge}-n\left(\frac{\kappa(\mathrm{A})\sqrt{2\pi n}+2}{4}\right)\alpha_{\max}
\end{aligned}\tag{30}$$

So from equations (28) and (30) we get that

$$|\epsilon_N(x)|\le\epsilon_{\text{edge}}+n\left(\frac{\kappa(\mathrm{A})\sqrt{2\pi n}+2}{4}\right)\alpha_{\max},$$

because $\epsilon_{\text{edge}}\underset{N\to\infty}{\longrightarrow}0$ and $\alpha_{\max}\underset{N\to\infty}{\longrightarrow}0$ then $\epsilon_N(x)\underset{N\to\infty}{\longrightarrow}0$. Now using Eq. (26) we get

$$F_N(x)\to F(x),\quad\forall x\in\mathbb{R}^n.$$

□

Appendix C
Lemma for update step

Lemma 27: Let $\epsilon_{i,j}=\int\star\,\mathrm{d}x$, where[8]

$$\star=\mathrm{N}_{\Sigma_i}^{\mu_i}(x)\left|\exp\left(-\frac{1}{2}\|z\|^2_{\mathrm{R}_j^{-1}}\right)-\exp\left(-\frac{1}{2}\|\tilde{z}_i\|^2_{\mathrm{R}_j^{-1}}\right)\right|,$$

$z=y-h(x)$, $\tilde{z}_i=y-h(\mu_i)-h'(\mu_i)(x-\mu_i)$, $i\in I_1$ (see Thm. 13). Now

$$\epsilon_{i,j}\underset{N\to\infty}{\longrightarrow}0.$$

[8]Here we simplify a little bit our notation.

Proof: First we define sets $C_{i,k}\subset\bar{C}_i$ which become smaller when N becomes larger.

$$C_{i,k}=\left\{x\,\middle|\,\left\|\left[\begin{array}{cc}\mathrm{I}_{d\times d} & 0\end{array}\right](x-\mu_i)\right\|\le\frac{1}{k}\right\},$$

where $i\in I_1$, $k>k_{\min}$ and

$$k_{\min}=\max_j\left(\sqrt{\frac{n_y c_{\mathrm{H}}(3\sqrt{\|\mathrm{R}_j\|}\|\mathrm{R}_j^{-1}\|+2)}{2}},\sqrt{\|\mathrm{R}_j^{-1}\|}\right).\tag{31}$$

Because $C_{i,k}\subset\bar{C}_i$ the Hessian matrices $h_1{}_j''(x)$ are bounded when $x\in C_{i,k}$. So there is a constant c_{H} such that (17)

$$\left|\xi^T h_j''(x)\xi\right|\le c_{\mathrm{H}}\xi^T\left[\begin{array}{cc}\mathrm{I}_{d\times d} & 0\\ 0 & 0\end{array}\right]\xi,\tag{32}$$

where $j\in\{1,\dots,n_y\}$.

Now

$$\epsilon_{i,j}=\int_{C_{i,k}}\star\,\mathrm{d}x+\int_{\complement C_{i,k}}\star\,\mathrm{d}x.\tag{33}$$

We show that $\int_{C_{i,k}}\star\,\mathrm{d}x\underset{k\to\infty}{\longrightarrow}0$ and $\int_{\complement C_{i,k}}\star\,\mathrm{d}x\underset{N\to\infty}{\longrightarrow}0$ for all k. We start to approximate integral $\int_{C_{i,k}}\star\,\mathrm{d}x$, and our goal is to show that $\int_{C_{i,k}}\star\,\mathrm{d}x\underset{k\to\infty}{\longrightarrow}0$. Now

$$\int_{C_{i,k}}\star\,\mathrm{d}x=\int_{C_{i,k}}\mathrm{N}_{\Sigma_i}^{\mu_i}(x)|f_{i,j}(x)|\mathrm{d}x,\tag{34}$$

where

$$\begin{aligned}
f_{i,j}(x)&=\exp\left(-\frac{1}{2}\|z\|^2_{\mathrm{R}_j^{-1}}\right)-\exp\left(-\frac{1}{2}\|z+\zeta_i\|^2_{\mathrm{R}_j^{-1}}\right),\\
\zeta_i&=h(x)-h(\mu_i)-h'(\mu_i)(x-\mu_i).
\end{aligned}\tag{35}$$

Using Taylor's theorem we get

$$\zeta_i=\sum_{j=1}^{n_y}e_j\frac{1}{2}(x-\mu_i)^T h_j''(\bar{x}_j)(x-\mu_i),$$

where $\bar{x}_j\in C_{i,k}$ for all $j\in\{1,\dots,n_y\}$ and e_j is the jth column of the identity matrix $\mathrm{I}_{n_y\times n_y}$. Now

$$\begin{aligned}
\|\zeta_i\|&\le\sum_{j=1}^{n_y}\left|\frac{1}{2}(x-\mu_i)^T h_j''(\bar{x}_j)(x-\mu_i)\right|\\
&\overset{(32)}{\le}\sum_{j=1}^{n_y}\frac{c_{\mathrm{H}}}{2}\left\|\left[\begin{array}{cc}\mathrm{I}_{d\times d} & 0\end{array}\right](x-\mu_i)\right\|^2\\
&\overset{x\in C_{i,k}}{\le}\frac{n_y c_{\mathrm{H}}}{2k^2},
\end{aligned}\tag{36}$$

where $k>k_{\min}$. So $\|\zeta_i\|\underset{k\to\infty}{\longrightarrow}0$, for all $i\in I_1$ when $x\in C_{i,k}$. Now we start to approximate $f_{i,j}(x)$, our goal being Eq. (44). We divide the problem into two parts, namely, $\|z\|^2_{\mathrm{R}_j^{-1}}\ge k^2$ and $\|z\|^2_{\mathrm{R}_j^{-1}}<k^2$. First we assume that $\|z\|^2_{\mathrm{R}_j^{-1}}\ge k^2$. Now

$$\|z\|^2\ge\frac{\|z\|^2_{\mathrm{R}_j^{-1}}}{\|\mathrm{R}_j^{-1}\|}\ge\frac{k^2}{\|\mathrm{R}_j^{-1}\|}\overset{(31)}{>}1\tag{37}$$

and using Eq. (31), Eq. (36) and Eq. (37) we get

$$\|\zeta_i\| \leq \min_j \left(1, \|z\|, \frac{1}{3\sqrt{\|\mathrm{R}_j\|}\|\mathrm{R}_j^{-1}\|}\right). \tag{38}$$

Now

$$\begin{aligned}
&\left|\|z+\zeta_i\|^2_{\mathrm{R}_j^{-1}} - \|z\|^2_{\mathrm{R}_j^{-1}}\right| \\
&= \left|2z^T\mathrm{R}_j^{-1}\zeta_i + \|\zeta_i\|^2_{\mathrm{R}_j^{-1}}\right| \\
&\leq (2\|z\| + \|\zeta_i\|)\,\|\mathrm{R}_j^{-1}\|\|\zeta_i\| \\
&\overset{(38)}{\leq} 3\|z\|\|\mathrm{R}_j^{-1}\|\|\zeta_i\| \\
&\overset{(40)}{\leq} 3\sqrt{\|\mathrm{R}_j\|}\sqrt{\|z\|^2_{\mathrm{R}_j^{-1}}}\|\mathrm{R}_j^{-1}\|\|\zeta_i\| \\
&\overset{(38)}{\leq} \sqrt{\|z\|^2_{\mathrm{R}_j^{-1}}} \overset{(\|z\|^2_{\mathrm{R}_j^{-1}} \geq k^2 > 1)}{\leq} \frac{1}{k}\|z\|^2_{\mathrm{R}_j^{-1}}.
\end{aligned} \tag{39}$$

Here we used the inequality

$$\begin{aligned}
\|z\|^2 &= z^T\mathrm{R}_j^{-\frac{1}{2}}\mathrm{R}_j\mathrm{R}_j^{-\frac{1}{2}}z \\
&\leq \|\mathrm{R}_j\|\|\mathrm{R}_j^{-\frac{1}{2}}z\|^2 = \|\mathrm{R}_j\|\|z\|^2_{\mathrm{R}_j^{-1}}.
\end{aligned} \tag{40}$$

So when $\|z\|^2_{\mathrm{R}_j^{-1}} \geq k^2$ we can approximate Eq. (35) as follows

$$\begin{aligned}
&f_{i,j}(x) \\
&\leq \exp\left(-\frac{1}{2}\|z\|^2_{\mathrm{R}_j^{-1}}\right) + \exp\left(-\frac{1}{2}\|z+\zeta_i\|^2_{\mathrm{R}_j^{-1}}\right), \\
&\overset{(39)}{\leq} \exp\left(-\frac{1}{2}\|z\|^2_{\mathrm{R}_j^{-1}}\right) + \exp\left(-\frac{k-1}{2k}\|z\|^2_{\mathrm{R}_j^{-1}}\right) \\
&\overset{\|z\|^2_{\mathrm{R}_j^{-1}} \geq k^2}{\leq} 2\exp\left(-\frac{k^2-k}{2}\right).
\end{aligned} \tag{41}$$

Now we assume that $\|z\|^2_{\mathrm{R}_j^{-1}} < k^2$, then

$$\begin{aligned}
&\left|\|z+\zeta_i\|^2_{\mathrm{R}_j^{-1}} - \|z\|^2_{\mathrm{R}_j^{-1}}\right| \\
&= \left|2z^T\mathrm{R}_j^{-1}\zeta_i + \|\zeta_i\|^2_{\mathrm{R}_j^{-1}}\right| \\
&\overset{(40)}{\leq} \left(2\sqrt{\|\mathrm{R}_j\|}k + \|\zeta_i\|\right)\|\mathrm{R}_j^{-1}\|\|\zeta_i\| \\
&\overset{(36)}{\leq} \left(2\sqrt{\|\mathrm{R}_j\|}k + \|\zeta_i\|\right)\|\mathrm{R}_j^{-1}\|\frac{n_y c_{\mathrm{H}}}{2k^2} \\
&\overset{(38)}{\leq} \left(2\sqrt{\|\mathrm{R}_j\|}k + 1\right)\|\mathrm{R}_j^{-1}\|\frac{n_y c_{\mathrm{H}}}{2k^2} \\
&\leq 2c_{\max}\frac{1}{k},
\end{aligned} \tag{42}$$

where $c_{\max} = \max_j \left(2\sqrt{\|\mathrm{R}_j\|}+1\right)\|\mathrm{R}_j^{-1}\|\frac{n_y c_{\mathrm{H}}}{4}$. So when $\|z\|^2_{\mathrm{R}_j^{-1}} < k^2$, we can approximate Eq. (35) as follows

$$\begin{aligned}
f_{\mathrm{i,j}}(x) &\leq \left|1 - \exp\left(\frac{1}{2}\|z\|^2_{\mathrm{R}_j^{-1}} - \frac{1}{2}\|z+\zeta_i\|^2_{\mathrm{R}_j^{-1}}\right)\right| \\
&\overset{(42)}{\leq} \exp\left(c_{\max}\frac{1}{k}\right) - 1.
\end{aligned} \tag{43}$$

Combining these results we get that if $k > k_{\min}$, then

$$\begin{aligned}
f_{i,j}(x) \leq \max\Bigg(&2\exp\left(-\frac{k^2-k}{2}\right), \dots \\
&\exp\left(c_{\max}\frac{1}{k}\right) - 1\Bigg)
\end{aligned} \tag{44}$$

Using this result we get (Eq. (34))

$$\begin{aligned}
&\int_{C_{i,k}} \star\,\mathrm{d}x = \int_{C_{i,k}} \mathrm{N}^{\mu_i}_{\Sigma_i}(x)|f_{i,j}(x)|\mathrm{d}x, \\
&\leq \max\left(2\exp\left(-\frac{k^2-k}{2}\right), \exp\left(c_{\max}\frac{1}{k}\right) - 1\right)
\end{aligned} \tag{45}$$

and then $\int_{C_{i,k}} \star\,\mathrm{d}x \underset{k\to\infty}{\longrightarrow} 0$.[9]

Finally we approximate the second integral $\int_{\complement C_{i,k}} \star\,\mathrm{d}x$ of Eq. (33) and our goal is to show that $\int_{\complement C_{i,k}} \star\,\mathrm{d}x \underset{N\to\infty}{\longrightarrow} 0$, for all k. Now

$$\begin{aligned}
\int_{\complement C_{i,k}} \star\,\mathrm{d}x &\leq \int_{\complement C_{i,k}} \mathrm{N}^{\mu_i}_{\Sigma_i}(x)\mathrm{d}x \\
&= \mathrm{P}(x \in \complement C_{i,k}) \\
&= \mathrm{P}\left(\left\|\left[\begin{array}{cc} \mathrm{I}_{d\times d} & 0 \end{array}\right](x-\mu_i)\right\| > \frac{1}{k}\right).
\end{aligned}$$

We know that (see Sec. IV-B, Sec. III-B (conventional approximation) and Corollary 20 (BGMA))

$$\sum_{j=1}^{d}(\Sigma_i)_{j,j} \underset{N\to\infty}{\longrightarrow} 0,\ \forall i \in I_2. \tag{46}$$

Using this information, Chebyshev's inequality and

$$\left[\begin{array}{cc} \mathrm{I}_{d\times d} & 0 \end{array}\right](x-\mu_i) \sim \mathrm{N}\left(0, (\Sigma_i)_{(1:d,1:d)}\right),$$

we see that $\int_{\complement C_{i,k}} \star\,\mathrm{d}x \underset{N\to\infty}{\longrightarrow} 0$ for all k and $i \in I_1$. Collecting these results we get Eq. (33) $\epsilon_{i,j} \underset{N\to\infty}{\longrightarrow} 0$.

□

ACKNOWLEDGMENT

The author would like to thank Niilo Sirola, Tommi Perälä and Robert Piché for their comments and suggestions. This study was partly funded by Nokia Corporation. The author acknowledges the financial support of the Nokia Foundation and the Tampere Graduate School in Information Science and Engineering.

[9] Actually it is straightforward to see that this integral converges to zero because $f_{i,j}(\mu_i) = 0$ and function $f_{i,j}(x)$ is continuous. However based on Eq. (44) we have some idea of the speed of convergence.

REFERENCES

[1] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*, ser. Statistics for Engineering and Information Science. Springer, 2001.

[2] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter, Particle Filters for Tracking Applications*. Boston, London: Artech House, 2004.

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

[4] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Mathematics in Science and Engineering. Academic Press, 1970, vol. 64.

[5] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, ser. Mathematics in Science and Engineering. Academic Press, 1982, vol. 141-2.

[6] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, ser. Prentice-Hall information and system sciences. Prentice-Hall, 1979.

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

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

[9] M. S. Grewal and A. P. Andrews, *Kalman Filtering Theory and Practice*, ser. Information and system sciences series. Prentice-Hall, 1993.

[10] D. Simon, *Optimal State Estimation Kalman, H_∞ and Nonlinear Approaches*. John Wiley & Sons, 2006.

[11] S. Ali-Löytty, N. Sirola, and R. Piché, "Consistency of three Kalman filter extensions in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, Jul. 2005.

[12] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *American Control Conference*, vol. 3, 1995, pp. 1628–1632.

[13] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, March 2004.

[14] R. S. Bucy and K. D. Senne, "Digital synthesis of non-linear filters," *Automatica*, vol. 7, no. 3, pp. 287–298, May 1971.

[15] S. C. Kramer and H. W. Sorenson, "Recursive Bayesian estimation using piece-wise constant approximations," *Automatica*, vol. 24, no. 6, pp. 789–801, 1988.

[16] N. Sirola and S. Ali-Löytty, "Local positioning with parallelepiped moving grid," in *Proceedings of 3rd Workshop on Positioning, Navigation and Communication 2006 (WPNC'06)*, Hannover, March 16th 2006, pp. 179–188.

[17] N. Sirola, "Nonlinear filtering with piecewise probability densities," Tampere University of Technology, Research report 87, 2007.

[18] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[19] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 736–746, March 2002.

[20] D. L. Alspach and H. W. Sorenson, "Nonlinear bayesian estimation using gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, Aug 1972.

[21] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, July 1971.

[22] T. Lefebvre, H. Bruyninckx, and J. De Schutter, "Kalman filters for nonlinear systems: a comparison of performance," *International Journal of Control*, vol. 77, no. 7, May 2004.

[23] N. Sirola, S. Ali-Löytty, and R. Piché, "Benchmarking nonlinear filters," in *Nonlinear Statistical Signal Processing Workshop*, Cambridge, September 2006.

[24] W. Cheney and W. Light, *A Course in Approximation Theory*, ser. The Brooks/Cole series in advanced mathematics. Brooks/Cole Publishing Company, 2000.

[25] W. Rudin, *Real and Complex Analysis*, 3rd ed., ser. Mathematics Series. McGraw-Hill Book Company, 1987.

[26] J. T.-H. Lo, "Finite-dimensional sensor orbits and optimal nonlinear filtering," *IEEE Transactions on Information Theory*, vol. 18, no. 5, pp. 583–588, September 1972.

[27] S. Ali-Löytty, "Efficient Gaussian mixture filter for hybrid positioning," in *Proceedings of PLANS 2008*, May 2008, (to be published).

[28] A. N. Shiryayev, *Probability*. Springer-Verlag, 1984.

[29] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis*, ser. Probability and mathematical statistics. London Academic Press, 1989.

[30] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2007*, May 2007, pp. 831–837.

[31] J. Kotecha and P. Djuric, "Gaussian sum particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2602–2612, October 2003.

[32] D. J. Salmond, "Mixture reduction algorithms for target tracking," *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, pp. 7/1–7/4, 1989.

[33] T. Ferguson, *A Course in Large Sample Theory*. Chapman & Hall, 1996.

[34] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, ser. Prentice-Hall information and system sciences. Prentice-Hall, 2000.

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 CANVAS 9.