Raitoharju *et al. EURASIP Journal on Advances in Signal Processing* (2015) 2015:36
DOI 10.1186/s13634-015-0221-2

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Binomial Gaussian mixture filter

Matti Raitoharju[1*], Simo Ali-Löytty[2] and Robert Piché[1]

Abstract

In this work, we present a novel method for approximating a normal distribution with a weighted sum of normal distributions. The approximation is used for splitting normally distributed components in a Gaussian mixture filter, such that components have smaller covariances and cause smaller linearization errors when nonlinear measurements are used for the state update. Our splitting method uses weights from the binomial distribution as component weights. The method preserves the mean and covariance of the original normal distribution, and in addition, the resulting probability density and cumulative distribution functions converge to the original normal distribution when the number of components is increased. Furthermore, an algorithm is presented to do the splitting such as to keep the linearization error below a given threshold with a minimum number of components. The accuracy of the estimate provided by the proposed method is evaluated in four simulated single-update cases and one time series tracking case. In these tests, it is found that the proposed method is more accurate than other Gaussian mixture filters found in the literature when the same number of components is used and that the proposed method is faster and more accurate than particle filters.

Keywords: Gaussian mixture filter; Estimation; Nonlinear filtering

1 Introduction

Estimation of a state from noisy nonlinear measurements is a problem arising in many different technical applications including object tracking, navigation, economics, and computer vision. In this work, we focus on Bayesian update of a state with a measurement when the measurement function is nonlinear. The measurement value y is assumed to be a d-dimensional vector whose dependence on the n-dimensional state x is given by

$$\mathrm{y} = h(\mathrm{x}) + \varepsilon, \tag{1}$$

where $h(\mathrm{x})$ is a nonlinear measurement function and ε is the additive measurement error, assumed to be zero mean Gaussian and independent of the prior, with nonsingular covariance matrix R. When a Kalman filter extension that linearizes the measurement function is used for the update, the linearization error involved is dependent on the measurement function and also the covariance of the prior: generally, larger prior covariances give larger linearization errors. In some cases, e.g., when the posterior is multimodal, no Kalman filter extension that uses a single normal distribution as the state estimate can estimate the posterior well.

In this paper, we use Gaussian mixtures (GMs) [1] to handle situations where the measurement nonlinearity is high. A GM is a weighted sum of normal distributions

$$p(\mathrm{x}) = \sum_{k=1}^{m} w_k p_{\mathrm{N}}(\mathrm{x}|\mu_k, P_k), \tag{2}$$

where m is the number of components in the mixture, w_k is the component weight $\left(\sum w_k = 1, w_k \geq 0\right)$, and $p_{\mathrm{N}}(\mathrm{x}|\mu_k, P_k)$ is the probability density function (pdf) of a multivariate normal distribution with mean μ_k and covariance P_k. The mean of a GM is

$$\mu = \sum_{k=1}^{m} w_k \mu_k \tag{3}$$

and the covariance is

$$P = \sum_{k=1}^{m} w_k \left[P_k + (\mu_k - \mu)(\mu_k - \mu)^T\right]. \tag{4}$$

Gaussian mixture filters (GMFs) work in such a manner that the prior components are split, if necessary, into smaller components to reduce the linearization error within a component. In splitting, it is desirable that the

*Correspondence: matti.raitoharju@tut.fi
[1] Department of Automation Science and Engineering, Tampere University of Technology, P.O. Box 692, FI-33101 Tampere, Finland
Full list of author information is available at the end of the article

Springer

© 2015 Raitoharju et al.; licensee Springer. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

mixture generated is similar to the original prior. Usually, the mean and covariance of the generated mixture are matched to the mean and covariance of the original component. Convergence properties are more rarely discussed, but, for example, in [2] a GM splitting that converges weakly to the prior component is presented.

We propose in this paper a way of splitting a prior component called Binomial Gaussian mixture (BinoGM) and show that when the number of components is increased, the pdf and cumulative distribution function (cdf) of the resulting mixture converge to the pdf and cdf of the prior component. Furthermore, we propose the Binomial Gaussian mixture filter (BinoGMF) for time series filtering. BinoGMF uses BinoGM in component splitting and optimizes the component covariance so that the measurement nonlinearity is kept small, while minimizing the required number of components needed to have a good approximation of the prior.

In our GMF implementation, we use the unscented Kalman filter (UKF) [3,4] for computing the measurement update. The UKF is used because the proposed splitting algorithm uses measurement evaluations that can be reused in the UKF update. To reduce the number of components in the GMF, we use the algorithm proposed in [5].

The rest of the paper is organized as follows. In Section 2, related work is discussed. Binomial GM is presented in Section 3, and BinoGMF algorithms are given in Section 4. Tests and results are presented in Section 5, and Section 6 concludes the article.

2 Related work

In this section, we present first the UKF algorithm that is used for updating the GM components and then five different GM generating methods that we use also in our tests section for comparison.

2.1 UKF

The UKF update is based on the evaluation of the measurement function at the so-called sigma points. The computation of the sigma points requires the computation of a matrix square root of the covariance matrix P, that is, a matrix $L(P)$ such that

$$P = L(P)L(P)^T. \tag{5}$$

This can be done using, for example, the Cholesky decomposition. The extended symmetric sigma point set is

$$\mathcal{X}_i = \begin{cases} \mu & i = 0 \\ \mu + \mathbf{\Delta}_i & 1 \leq i \leq n \\ \mu - \mathbf{\Delta}_{i-n} & n < i \leq 2n \end{cases}, \tag{6}$$

where n is the dimension of the state, $\mathbf{\Delta}_i = \sqrt{n+\xi}L(P)_{[:,i]}$ ($L(P)_{[:,i]}$ is the ith column of $L(P)$), and ξ is a configuration parameter. The images of the sigma points transformed with the measurement function are

$$\mathcal{Y}_i = h(\mathcal{X}_i). \tag{7}$$

The prior mean μ and covariance P are updated using

$$\begin{aligned} \mathbf{z} &= \sum_{i=0}^{2n} \Omega_{i,m}\mathcal{Y}_i \\ S &= R + \sum_{i=0}^{2n} \Omega_{i,c}\left(\mathcal{Y}_i - \mathbf{z}\right)\left(\mathcal{Y}_i - \mathbf{z}\right)^T \\ C &= \sum_{i=0}^{2n} \Omega_{i,c}\left(\mathcal{X}_i - \mu\right)\left(\mathcal{Y}_i - \mathbf{z}\right)^T, \\ K &= CS^{-1} \\ \mu^+ &= \mu + K(\mathbf{y} - \mathbf{z}) \\ P^+ &= P - KSK^T \end{aligned} \tag{8}$$

where μ^+ is the posterior mean, P^+ is the posterior covariance, $\Omega_{0,m} = \frac{\xi}{n+\xi}$, $\Omega_{0,c} = \frac{\xi}{n+\xi} + (1 - \alpha_{\mathrm{UKF}}^2 + \beta_{\mathrm{UKF}})$, $\Omega_{i,c} = \Omega_{i,m} = \frac{1}{2n+2\xi}, (i > 0)$ and $\xi = \alpha_{\mathrm{UKF}}^2(n + \kappa_{\mathrm{UKF}}) - n$. The variables with subscript UKF are configuration parameters [3,4].

When UKF is used to update a GM, update (8) is computed separately for each component. Each component's weight is updated with the innovation likelihood

$$w_k^+ = w_k p_N(\mathbf{y}|\mathbf{z}_k, S_k), \tag{9}$$

where k is the index of kth mixture component. Weights are normalized so that $\sum_{k=1}^{m} w_k^+ = 1$.

2.2 Havlak and Campbell (H&C)

In H&C [6], a GM is used to improve the estimation in the case of the nonlinear state transition model, but the method can be applied also to the case of the nonlinear measurement model by using measurement function in place of state transition function.

In H&C, a linear model is fitted to the transformed sigma points using least squares fitting

$$\arg\min \sum \left(A\mathcal{X}_i + \mathbf{b} - \mathcal{Y}_i\right)^2. \tag{10}$$

For computing the direction of maximum nonlinearity, the sigma points $\mathcal{X}_i$ are weighted by the norm of their associated residual $||A\mathcal{X}_i + \mathbf{b} - \mathcal{Y}_i||$ and the direction of nonlinearity is the eigenvector corresponding to the largest eigenvalue of the second moment of this set of weighted sigma points.

In the prior splitting, the first dimension of a standard multivariate normal distribution is replaced with a precomputed mixture. An affine transformation is then

applied to this mixture. The affine transformation is chosen to be such that after transformation the split dimension is aligned with the direction of the maximum nonlinearity and such that the resulting mixture is a good approximation of the prior component.

2.3 Faubel and Klakow (F&K)

In [7], the nonlinearity corresponding to ith pair of sigma points is

$$\eta_i = \frac{1}{2}||\mathcal{Y}_i + \mathcal{Y}_{i+n} - 2\mathcal{Y}_0||. \tag{11}$$

If there is more than one component, the component next to be split is chosen to be the one that maximizes

$$\sqrt{w_i \eta_i}. \tag{12}$$

In the paper, there are two alternatives for choosing the direction of maximum nonlinearity. Better results with fewer components were obtained in [7] by using the eigenvector corresponding of the largest eigenvalue of matrix

$$\sum_{i=1}^{n} \eta_i \frac{(\mathcal{X}_i - \mathcal{X}_{i+n})(\mathcal{X}_i - \mathcal{X}_{i+n})^T}{||\mathcal{X}_i - \mathcal{X}_{i+n}||^2}. \tag{13}$$

The splitting direction is chosen to be the eigenvector of the prior covariance closest to this direction, and the prior components are split into two or three components that preserve the mean and covariance of the original component.

2.4 Split and merge (S&M)

In the split and merge unscented Gaussian mixture filter, the component to be split is chosen to be the one that would have the highest posterior probability (9) with nonlinearity

$$\eta = \sum \frac{||\mathcal{Y}_i + \mathcal{Y}_{i+n} - 2\mathcal{Y}_0||^2}{||\mathcal{Y}_i - \mathcal{Y}_{i+n}||^2} \tag{14}$$

higher than some predefined value [8]. The splitting direction is chosen to be the eigenvector corresponding to the maximum eigenvalue of the prior covariance. Components are split into two components with equal covariances.

2.5 Box GMF (BGMF)

The Box GMF [2] uses the nonlinearity criterion

$$\frac{\sqrt{\mathrm{tr} PH_i PH_i}}{\sqrt{R_{[i,i]}}} > 1, \text{for some } i, \tag{15}$$

where H_i is the Hessian of the ith component of the measurement equation. This criterion is only used to assess whether or not the splitting should be done.

If a measurement is considered highly nonlinear, the prior is split into a grid along all dimensions. Each grid cell is replaced with a normal distribution having the mean and covariance of the pdf inside the cell and having as weight the amount of probability inside the cell. It is shown that the resulting mixture converges weakly to the prior component.

2.6 Adaptive splitting (AS)

In [9], the splitting direction is computed by finding the direction of maximum nonlinearity of the following transformed version of criterion (15) for one-dimensional measurements:

$$\mathrm{tr}\ PHPH > R. \tag{16}$$

It is shown that the direction of the maximum nonlinearity is aligned with the eigenvector corresponding to the largest absolute eigenvalue of PH. In [9], a numerical method for approximating the splitting direction that is exact for second-order polynomial measurements is also presented.

3 Binomial Gaussian mixture

The BinoGM is based on splitting a normal distributed prior component into smaller ones using weights and transformed locations from the binomial distribution.

The binomial distribution is a discrete probability distribution for the number of successes in a sequence of independent Bernoulli trials with probability of success p. The probability mass function for having k_s successes in m_s trials is

$$p_{B,m_s}(k_s) = \binom{m_s}{k_s} p^{k_s}(1-p)^{m_s - k_s}. \tag{17}$$

A standardized binomial distribution is a binomial distribution scaled and translated so that it has zero mean and unit variance. The pdf of a standardized binomial distribution can be written using Dirac delta notation as

$$\begin{aligned} p_{B,m}(x) = &\sum_{k=1}^{m} \binom{m-1}{k-1} p^{k-1}(1-p)^{m-k} \delta \\ &\times \left(x - \frac{k - 1 - (m-1)p}{\sqrt{(m-1)p(1-p)}} \right), \end{aligned} \tag{18}$$

where, for later convenience we have used $m = m_s + 1$, the number of possible outcomes, and $k = k_s + 1$. By the Berry-Esseen theorem [10], the sequence of cdfs of (18) converges uniformly to the cdf of the standard normal distribution as m increases with fixed p. In other words, the sequence of standardized binomial distributions converges weakly to a standard normal distribution as the number of trials is increased. The error of (18) as an approximation of a standard normal distribution is

of order $\mathcal{O}\left(\frac{1}{\sqrt{m(1-p)p}}\right)$ [11]. This error is smallest when $p = 0.5$, in which case (18) simplifies to

$$p_{B,m}(x) = \sum_{k=1}^{m} \binom{m-1}{k-1} \left(\frac{1}{2}\right)^{m-1} \delta\left(x - \frac{2k-m-1}{\sqrt{m-1}}\right). \tag{19}$$

If a random variable having a standardized binomial distribution is scaled by σ, then its variance is σ^2.

The BinoGM is constructed using a mixture of standard normal distributions along the main axes, with mixture component means and weights selected using a scaled binomial distribution. The mixture product is then transformed with an affine transformation to have the desired mean and covariance.

To illustrate, the construction of a two-dimensional GM is shown in Figure 1. On the left, there are two binomial distributions having variances $\sigma_1^2 = 1$ and $\sigma_2^2 = 8$ with probability mass distributed in $m_1 = 5$ and $m_2 = 3$ points. One-dimensional GMs are generated by taking point mass locations and weights from the discrete distributions as the means and weights of standard normal distributions. The product of the two one-dimensional GMs is a two-dimensional GM, to which the affine transformation (multiplication with an invertible matrix T and addition of a mean μ) is applied.

A BinoGM consisting of $m_{\text{tot}} = \prod_{i=1}^{n} m_i$ components has a pdf of the form

$$p_{\text{BinoGM}}(\mathbf{x}) = \sum_{l=1}^{m_{\text{tot}}} w_l p_N(\mathbf{x}|\mu_l, P), \tag{20}$$

where

$$w_l = \prod_{i=1}^{n} \binom{m_i - 1}{C_{l,i} - 1} \left(\frac{1}{2}\right)^{m_i - 1} \tag{21a}$$

$$\mu_l = T \begin{bmatrix} \sigma_1 \frac{2C_{l,1} - m_1 - 1}{\sqrt{m_1 - 1}} \\ \sigma_2 \frac{2C_{l,2} - m_2 - 1}{\sqrt{m_2 - 1}} \\ \vdots \\ \sigma_n \frac{2C_{l,n} - m_n - 1}{\sqrt{m_n - 1}} \end{bmatrix} + \mu \tag{21b}$$

$$P = TT^T \tag{21c}$$

and C is the Cartesian product

$$C = \{1, \ldots, m_1\} \times \{1, \ldots, m_2\} \times \ldots \times \{1, \ldots, m_n\}, \tag{22}$$

which contains sets of indices to binomial distributions of all mixture components. Notation $C_{l,i}$ is used to denote the ith component of the lth combination. If $m_k = 1$, the term $\frac{2C_{l,k} - m_k - 1}{\sqrt{m_k - 1}}$ in (21b) is replaced with 0.

We use the notation

$$\mathbf{x}_{\text{BinoGM}} \sim \text{BinoGM}(\mu, T, \Sigma, m_1, \ldots, m_n), \tag{23}$$

where $\Sigma = \text{diag}\left(\sigma_1^2, \ldots, \sigma_n^2\right)$, to denote a random variable distributed according to a BinoGM. Parameters of the distribution are $\mu \in \mathbb{R}^n$, $T \in \mathbb{R}^{n \times n} \wedge \det T \neq 0$ and $\forall i; 1 \leq i \leq n : m_i \in \mathbb{N}^+ \wedge \sigma_i \in \mathbb{R}^+$.

Matrix T is a square root (5) of a component covariance P. We use notation T instead of $L(P)$ here, because the matrix square root $L(P)$ is not unique and the choice of T affects the covariance of the mixture (25). BinoGM could also be parameterized using prior covariance P_0 instead of T. In such case, T should be replaced with $T = L(P_0)(\Sigma + I)^{-\frac{1}{2}}$. In this case, the component covariance is affected by the choice of $L(P_0)$.

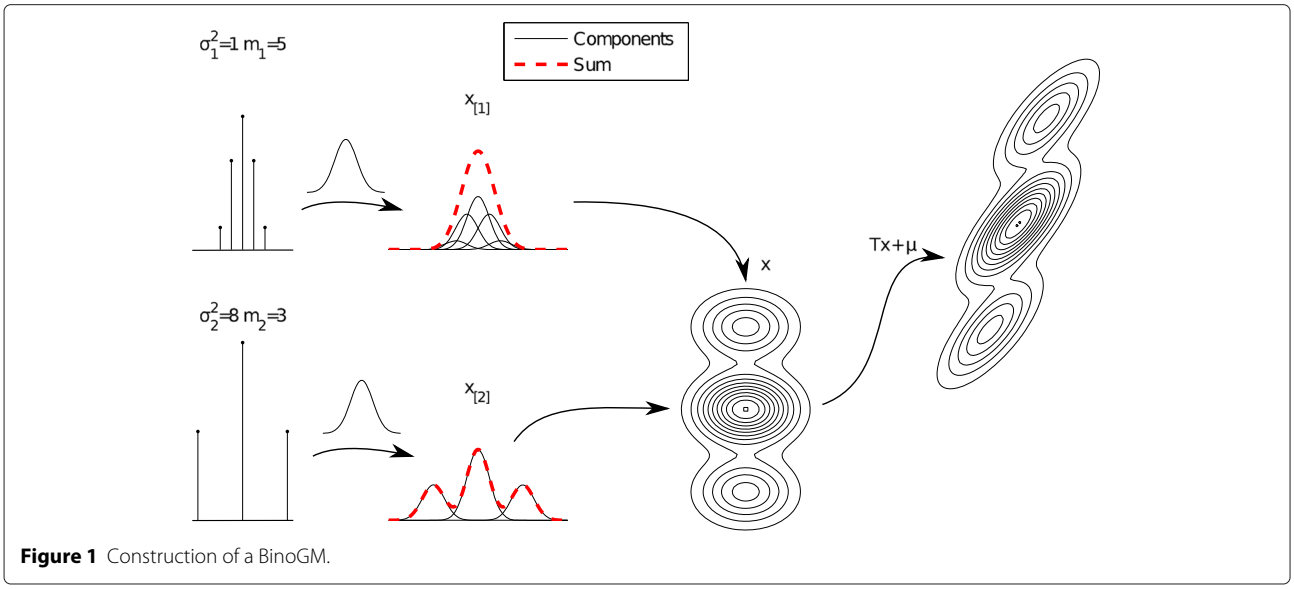

Figure 1 Construction of a BinoGM.

We will next show that

$$\mathrm{E}(\mathbf{x}_{\mathrm{BinoGM}}) = \mu \tag{24}$$

$$\mathrm{cov}(\mathbf{x}_{\mathrm{BinoGM}}) = P_0 = T(\Sigma + I)T^T \tag{25}$$

$$\begin{aligned}\lim_{m_1,\ldots,m_n\to\infty} &\mathrm{BinoGM}(\mu, T, \Sigma, m_1, \ldots, m_n) \\ &= \mathrm{N}\left(\mu, T(\Sigma + I)T^T\right).\end{aligned} \tag{26}$$

The limit (26) for BinoGM is in the sense of the weak convergence and convergence of the pdf.

First, we consider a sum of a scaled standardized binomial random variable $x_{B,m}$ and a standard normal random variable x_N that are independent,

$$\sigma x_{B,m} + x_N. \tag{27}$$

Because $x_{B,m}$ converges weakly to standard normal distribution, then by the continuous mapping theorem, $\sigma x_{B,m}$ converges weakly to normal distribution with variance σ^2 [12].

The pdf of a sum of independent random variables is the convolution of the pdfs, and the cdf is the convolution of the cdfs [13]. For (27), the pdf is

$$\begin{aligned} g_m(x) &= \int_{-\infty}^{\infty} p_{B,m}(y) p_N(x - y, 0, 1)\mathrm{d}y \\ &= \sum_{i=1}^{m} w_i p_N\left(x|\mu_i, 1\right), \end{aligned} \tag{28}$$

where $w_i = \binom{m-1}{i-1}$ and $\mu_i = \sigma\frac{2i-m-1}{\sqrt{m-1}}$. This is a pdf of a GM whose variance is $\sigma^2 + 1$. For the weak convergence, we use the following result (Theorem 105 of [13]):

Theorem 1. *If cdf $\Phi_m(x)$ converges weakly to $\Phi(x)$ and cdf $B_m(x)$ converges weakly to $B(x)$ and Φ_m and B_m are independent, then the convolution of $\Phi_m(x)$ and $B_m(x)$ converges to the convolution of $\Phi(x)$ and $B(x)$.*

Because the cdf of $\sigma x_{B,m}$ converges weakly to the cdf of a normal distribution with variance σ^2 and the sum of two independent normally distributed random variables is normally distributed, the cdf of the sum (27) converges to the cdf of a normal distribution.

Convergence of the pdf means that

$$\sup_{x\in\mathbb{R}} |g_m(x) - p_N\left(x|0, \sigma^2 + 1\right)| \to 0, \mathrm{as} m \to \infty. \tag{29}$$

For this, we consider the requirements given in [14] (Lemma 1).

1. The pdf g_m exists
2. Weak convergence
3. $\sup_m |g_m(x)| \leq M(x) < \infty$, for all $x \in \mathbb{R}$
4. For every x and $\varepsilon > 0$, there exist $\delta(\varepsilon)$ and $m(\varepsilon)$ such that $|x - y| < \delta(\varepsilon)$ implies that $|g_m(x) - g_m(y)| < \varepsilon$ for all $m \geq m(\varepsilon)$.

The pdf (28) exists and the weak convergence was already shown. The third item is satisfied because the maximum value of the pdf is bounded:

$$\begin{aligned} g_m(x) &= \sum_{i=1}^{m} w_i p_N(x, \mu_i, 1) \leq \max(p_N(x|0, 1)) \\ &= \frac{1}{\sqrt{2\pi}} < \infty. \end{aligned} \tag{30}$$

Then from

$$\begin{aligned} |g_m(x) - g_m(y)| &\leq \sum_{i=1}^{m} w_i |p_{N,i}(x) - p_{N,i}(y)| \\ &\overset{\exists c\in\mathbb{R}}{\leq} \sum_{i=1}^{m} w_i |p'_{N,i}(c)||x - y| \\ &< \delta \max\left(|p'_N(c)| + 1\right), \end{aligned} \tag{31}$$

we see that by choosing $\delta = \frac{\varepsilon}{\max|p'_N(c)|+1}$, the requirements of the fourth item are fulfilled and the pdf converges.

For convergence of a multidimensional mixture, consider a random vector $\mathbf{x}$ whose ith component is given by

$$\mathbf{x}_{[i]} = \sigma_i x_{B,m_i} + x_{N,i}, \tag{32}$$

where x_{B,m_i} has the standardized binomial distribution (19) and $x_{N,i}$ has the standard normal distribution. Let the components $\mathbf{x}_{[i]}$ be independent. The variance of $\mathbf{x}$ is

$$\mathrm{var}\,\mathbf{x} = \Sigma + I, \tag{33}$$

where $\Sigma_{[i,i]} = \sigma_i^2$, and the multidimensional cdf and pdf are products of cdfs and pdfs of each component. As we showed, the approximation error in the one-dimensional case approaches 0 when the number of components is increased. Now the multidimensional pdf and cdf converge also because

$$\prod_{i=1}^{n} \lim_{\varepsilon_i\to 0} f_i(x) + \varepsilon_i = \prod_{i=1}^{n} f_i(x), \tag{34}$$

where ε_i is an error term.

Let random variable $\mathbf{x}$ be multiplied with a nonsingular matrix T and be added to a constant μ

$$\mathbf{x}_{\mathrm{BinoGM}} = T\mathbf{x} + \mu. \tag{35}$$

As a consequence of continuous mapping theorem [12], the cdf of $T\mathbf{x} + \mu$ approaches the cdf of a normal distribution with mean μ and covariance $T(\Sigma + I)T^T$. The transformed variable has pdf

$$p_{\mathbf{x}_{\mathrm{BinoGM}}}(\mathbf{x}_{\mathrm{BinoGM}}) = p_{\mathbf{x}}\left(T^{-1}\left(\mathbf{x}_{\mathrm{BinoGM}} - \mu\right)\right)|\det T^{-1}|. \tag{36}$$

Because $p_{\mathbf{x}}$ converges to a normal distribution, $p_{\mathbf{x}\mathrm{BinoGM}}$ converges also and as such the pdf of the multidimensional mixture (35), which is BinoGM (21), converges to a normal distribution.

4 Binomial Gaussian mixture filter

In this section, we present BinoGMF, which uses BinoGM for splitting prior components when nonlinear measurements occur. In Section 4.1, we propose algorithms for choosing parameters for BinoGM in the case of one-dimensional measurements. After the treatment of one-dimensional measurements, we extend the proposed method for multidimensional possibly dependent measurements in Section 4.2, and finally, we present the algorithm for time series filtering in Section 4.3.

4.1 Choosing the parameters for a BinoGM

The BinoGM can be constructed using (21), when numbers of components $m_1, \ldots, m_n$, binomial variances $\sigma_1, \ldots, \sigma_n$, and parameters of the affine transformation T and μ are given. Now the goal is to find parameters for the BinoGM such that it is a good approximation of the prior, the nonlinearity is below a desired threshold η_{limit}, and the number of components is minimized. In this subsection, we concentrate on the case of a one-dimensional measurement. Treatment of multidimensional measurements is presented in Section 4.2.

In splitting a prior component, the parameters are chosen such that the mean and covariance are preserved. The mean is preserved by choosing μ in (21b) to be the mean of the prior component. If the prior covariance matrix is P_0 and it is split into smaller components that have covariance P (i.e., $P_0 - P$ is positive semi-definite), matrices T and Σ have to be chosen so that

$$TT^T = P \tag{37a}$$

$$T\Sigma T^T + P = P_0. \tag{37b}$$

It was shown in Section 3 that the BinoGM converges to a normal distribution when the number of components is increased. In practical use cases, the number of components has to be specified. We propose a heuristic rule to choose the number of components such that two equally weighted components that are next to each other produce a unimodal pdf. In Appendix 1, it is shown that unimodality is preserved if

$$m \geq \sigma^2 + 1. \tag{38}$$

Because we want to minimize the number of components and we can choose σ so that σ^2 is an integer, we will use the following relationship

$$m = \sigma^2 + 1. \tag{39}$$

In Figure 2, this rule is illustrated in the situation where a mixture consisting of unit variance components is used to approximate a Gaussian prior having variance $\sigma_0 = 3$. In this case, (27) gives $\sigma^2 = 2$. According to (39), the proposed number of components is three. The figure shows how the mixture with two components has clearly distinct peaks and that the approximation improvement when using four components instead of three is insignificant.

For the multidimensional case, the component number rule generalizes in a straightforward way to

$$m_i = \Sigma_{[i,i]} + 1. \tag{40}$$

Using this relationship, matrix Σ in (67) is linked to the total number of components by the formula $m_{\text{tot}} = \prod m_i = \prod \left(\Sigma_{[i,i]} + 1\right)$.

To approximate the amount of linearization error in the update, we use

$$\eta = \frac{\operatorname{tr} PHPH}{R} \tag{41}$$

as the nonlinearity measure, which is similar to the ones used in [2,9]. The Hessian H of the measurement function is evaluated at the mean of a component and treated as a constant. Analytical evaluation of this measure requires that the second derivative of the

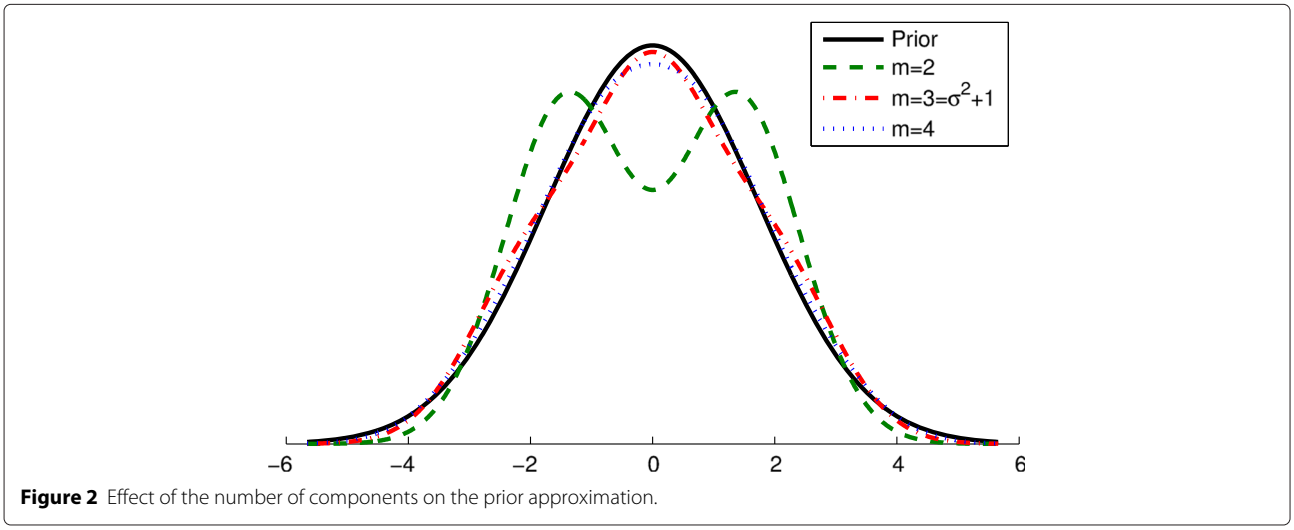

Figure 2 Effect of the number of components on the prior approximation.

measurement function (1) exists. The optimal component size is such that the total number of components is minimized while (40) is satisfied and the nonlinearity (41) is below η_{limit}. Nonlinearity measure (41) is defined for one-dimensional measurements only. We present a method for handling multidimensional measurements in Section 4.2.

We show in Appendix 2 that if we neglect the integer nature of m_i, the optimal values for m_i are 1 or satisfy the equation

$$\frac{\lambda_i^2}{m_i^2} = \frac{\lambda_j^2}{m_j^2} \tag{42}$$

with the conditions that $m_i \geq 1$ and $\frac{1}{R}\sum \frac{\lambda_i^2}{m_i^2} = \eta_{\text{limit}}$, where λ_i is the ith eigenvalue in the eigendecomposition

$$V \Lambda V^T = L(P_0)^T H L(P_0). \tag{43}$$

The optimal T matrix is

$$T = L(P_0) V \text{diag}\left(\frac{1}{\sqrt{m_1}}, \ldots, \frac{1}{\sqrt{m_n}}\right) \tag{44}$$

and the component covariance is

$$P = L(P_0) V \text{diag}\left(\frac{1}{m_1}, \ldots, \frac{1}{m_n}\right) V^T L(P_0)^T. \tag{45}$$

Using (44) and (40), the component mean becomes

$$\mu_l = L(P_0) V \text{diag}\left(\frac{1}{\sqrt{m_1}}, \ldots, \frac{1}{\sqrt{m_n}}\right) \begin{bmatrix} 2C_{l,1} - m_1 - 1 \\ 2C_{l,2} - m_2 - 1 \\ \vdots \\ 2C_{l,n} - m_n - 1 \end{bmatrix} + \mu. \tag{46}$$

In (43), $L(P_0)^T H L(P_0)$ can be replaced with $\frac{1}{\gamma^2} Q$, where

$$Q_{[i,j]} = \begin{cases} h(\mathbf{x} + \mathbf{\Delta}_i) + h(\mathbf{x} - \mathbf{\Delta}_i) - 2h(\mathbf{x}) & , i = j \\ \frac{1}{2}[h(\mathbf{x} + \mathbf{\Delta}_i + \mathbf{\Delta}_j) + h(\mathbf{x} - \mathbf{\Delta}_i - \mathbf{\Delta}_j) - \\ 2h(\mathbf{x}) - Q_{[i,i]} - Q_{[j,j]}] & , i \neq j \end{cases} \tag{47}$$

and $\mathbf{\Delta}_i = \gamma L(P_0)_{[:,i]}$. If γ is chosen as $\gamma = \sqrt{n + \xi}$, then the computed values of the measurement function in (47) may also be used in the UKF [9]. The Q matrix can also be used for computing the amount of nonlinearity (41), because $\text{tr}PHPH \approx \frac{QQ}{\gamma^4}$ [9]. Using the Q matrix instead of the analytically computed H takes measurement function values from a larger area into account, analytical differentiation of the measurement function is not needed, and the numerical approximation can be done for any measurement function (1) that is defined everywhere. Because the approximation is based on second-order polynomials, it is possible that when the measurement function is not differentiable, the computed nonlinearity value does not represent the nonlinearity well.

Figure 3 shows three different situations where second-order polynomials are fitted to a function either using Taylor expansion or sigma points. The analytical nonlinearity measure (43) can be interpreted as the use of the Taylor expansion for nonlinearity computation and (47) as a second-order polynomial interpolation at sigma points for nonlinearity computation. The figure shows how the analytical fit is more accurate close to the mean point, but the second-order approximation gives a more uniform fit in the whole region covered by sigma points. The two first functions in Figure 3 are smooth, but the third is piecewise linear, i.e., its Hessian is zero everywhere except at points

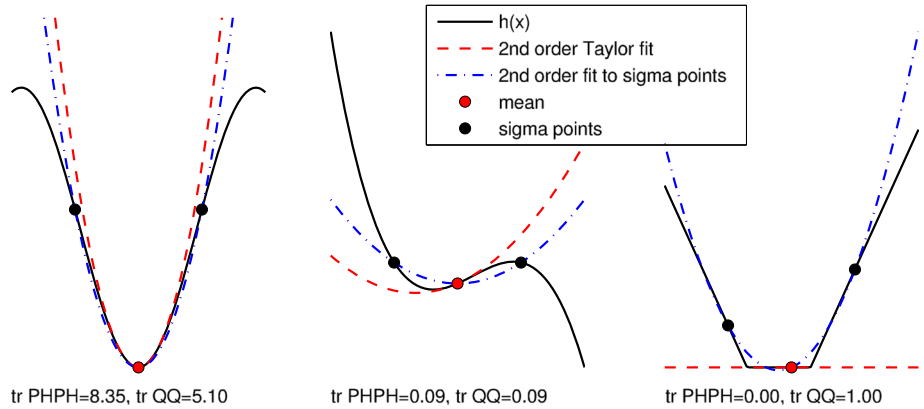

Figure 3 Second-order polynomials fitted to three nonlinear measurement functions using Taylor expansion and sigma points.

where the slope changes, where it is undefined. The analytical fit of this function is linear, but the fit to the sigma points detects nonlinearity.

Our proposed algorithm for choosing the integer number of components is presented in Algorithm 1. At the start of the algorithm, the nonlinearity is reduced to η_{limit}, but if this reduction produces more than m_{limit} components, the component covariance is chosen such that nonlinearity is minimized while having at most m_{limit} components. The algorithm for splitting a Gaussian with component weight w_0 is summarized in Algorithm 2.

Algorithm 1: Algorithm for choosing the number of components for a scalar measurement

$\eta \leftarrow R\eta_{\text{limit}}$ // Remaining nonlinearity scaled with scalar measurement variance R
$i \leftarrow 1$
Compute V and Λ of eigendecomposition $V\Lambda V^T = 1/\gamma^2 Q$ so that eigenvalues are sorted according to their absolute values. // Q is a $n \times n$ matrix defined in (47)
while $i \leq n$ **do**
  $m_i \leftarrow \max\left(\left\lceil \sqrt{\frac{(n+1-i)}{\eta}}|\lambda_i| \right\rceil, 1\right)$ // Dimensions share of nonlinearity $\frac{\lambda_i^2}{m_i^2} \leq \frac{\eta}{n+1-i}$
  $\eta \leftarrow \eta - \frac{\lambda_i^2}{m_i^2}$ // Update remaining nonlinearity
  $i \leftarrow i+1$
end
// If number of components exceeds the maximum
if $\prod_{i=1}^{n} m_i > m_{\text{limit}}$ **then**
  $i \leftarrow 1$
  while $\lambda_i = 0 \;\&\; i \leq n$ **do**
    $m_i \leftarrow 1$ // For linear dimension use 1 component
    $i \leftarrow i+1$
  end
  while $i \leq n$ **do**
    // Number of components proportional to $|\lambda_i|$
    $m_i \leftarrow \max\left(\left\lfloor |\lambda_i| \left(\frac{m_{\text{limit}}}{\prod_{j=1}^{i-1} m_j \prod_{j=i}^{n} |\lambda_j|}\right)^{\frac{1}{n-i+1}} \right\rfloor, 1\right)$
    $i \leftarrow i+1$
  end
end

Algorithm 2: Algorithm for splitting a Gaussian

Compute $n \times n$ matrix square root $L(P_0)$ // Use the same $L(P_0)$ in every step
Use $L(P_0)$ to compute Q // (47)
$V\Lambda V^T = 1/\gamma^2 Q$ // Compute eigendecomposition (43)
Compute $m_1, \ldots, m_n$ using Algorithm 1
$T \leftarrow L(P_0) V \operatorname{diag}\left(\frac{1}{\sqrt{m_1}}, \ldots, \frac{1}{\sqrt{m_n}}\right)$ // $n \times n$ transformation matrix (44)
$X \leftarrow \mathbb{R}^{n \times 1}$ // Initialize X for orthogonal means
$w \leftarrow w_0$ // Prior weight
$m_X \leftarrow 1$ // Current size of X
// Go through all combinations
for $k = 1 : n$ **do**
  $X \leftarrow \left[\overbrace{X, \ldots, X}^{m_k}\right]$ // Catenate matrix X m_k times
  $w \leftarrow \left[\overbrace{w, \ldots, w}^{m_k}\right]$ // Catenate w m_k times
  for $j = 1 : m_k$ **do**
    for $i = 1 : m_X$ **do**
      $X_{[k,mj-i+1]} \leftarrow 2j - m_k - 1$ // $2C - m - 1$ part of (46)
      $w_{[mj-i+1]} \leftarrow \binom{m_k-1}{j-1} \left(\frac{1}{2}\right)^{m_k-1} w_{[m_{<k}j-i+1]}$ // binomial weight
    end
    $m_X \leftarrow m_X m_k$ // Number of components in the mixture so far
  end
end
// Parameters for components of the BinoGM ($1 \leq l \leq m_X = \prod m_i$)
$\mu_l \leftarrow TX_{[:,l]} + \mu$ // Mean (46)
$w_l \leftarrow w_{[l]}$ // Weight
$P_l \leftarrow TT^T$ // Covariance

Figure 4 shows the result of splitting a prior in a highly nonlinear case. To verify the relationship between m and σ^2 (40) and determine which η_{limit} should be used, we tested the splitting with different parameters ($\cdot\times$ tells how much was the number of components increased). In the figure captions, m is the total number of components and KL is the Kullback-Leibler divergence [15]

$$\int \ln\left(\frac{p(\mathbf{x})}{q(\mathbf{x})}\right) p(\mathbf{x}) \mathrm{d}\mathbf{x}, \tag{48}$$

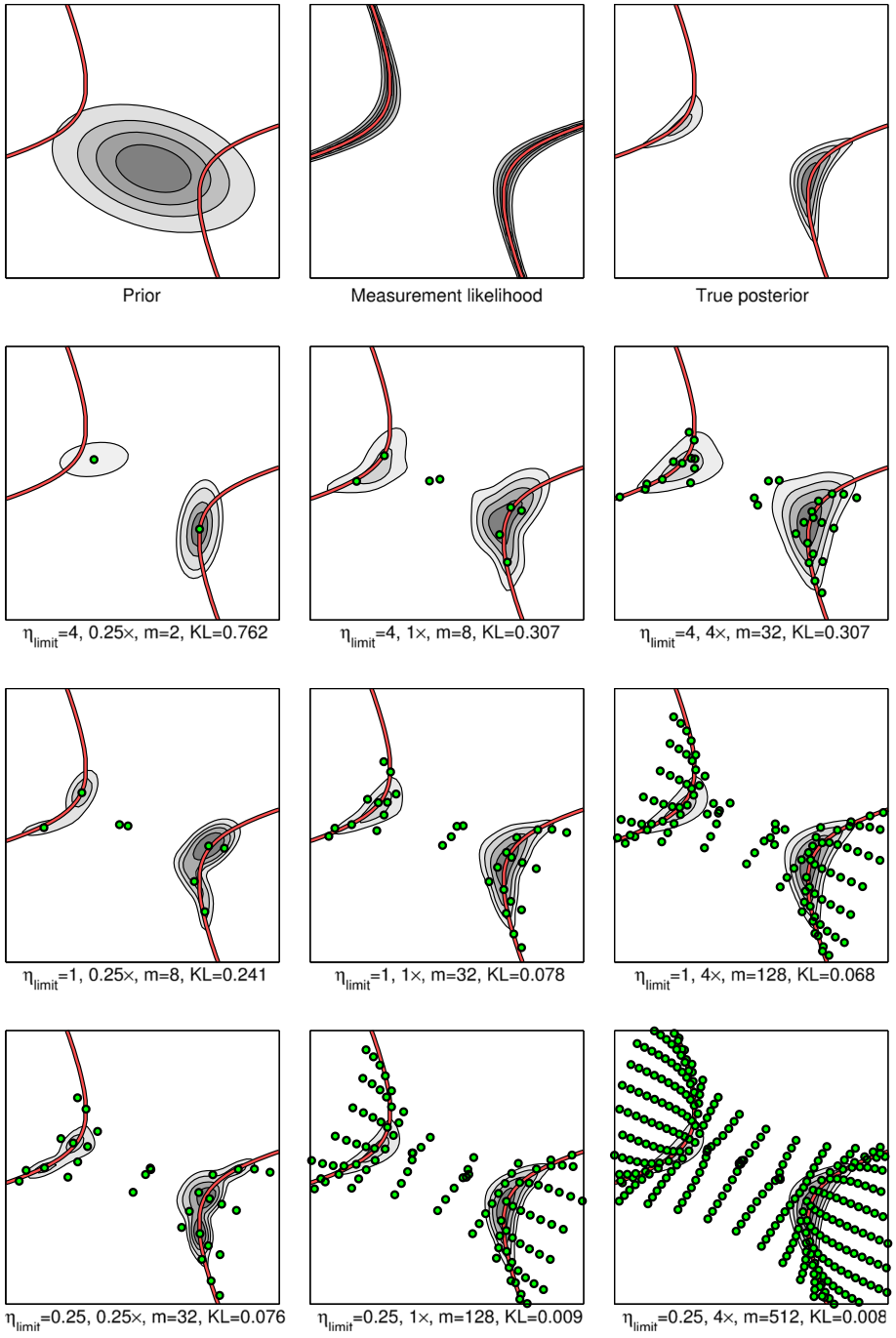

Figure 4 Effect of parameters on posterior approximation. Green dots are located at posterior component means, and red lines show where the measurement likelihood is highest. In subfigure captions, η_{limit} is the nonlinearity threshold for each component, $\cdot\times$ is the number of components compared to proposed, m is the total number of components, and KL is the Kullback-Leibler divergence of the GM posterior approximation with respect to the true posterior.

where $p(\mathbf{x})$ is the true posterior and $q(\mathbf{x})$ is the approximated posterior estimate.

From the figure, we can see that if the number of components is quadrupled from (40), the KL divergence does not improve significantly. If the subfigures with the same numbers of components are compared, it is evident that subfigures with ($\eta_{\text{limit}} = 4$) have clearly the worst KL divergence. Subfigures with $\eta_{\text{limit}} = 0.25$ have slightly better KL divergence than whose with $\eta_{\text{limit}} = 1$ but are by visual inspection more peaky. The KL divergence

reduces towards the bottom right corner, which indicates convergence.

4.2 Multidimensional measurements

To be able to use BinoGMF with possibly correlated multidimensional measurements, the nonlinearity measure (41) for scalar measurements cannot be applied directly. However, because we assume that the measurement noise is additive and nondegenerate Gaussian, the measurement can be decorrelated with a linear invertible transformation

$$\hat{\mathbf{y}} = L(R)^{-1}\mathbf{y} = L(R)^{-1}h(\mathbf{x}) + L(R)^{-1}\varepsilon. \tag{49}$$

The covariance of the transformed measurements is $L(R)^{-1}RL(R)^{-I} = I$. This kind of transformation does not change the posterior, which is shown for the UKF in Appendix 3.

The UKF update is an approximation of a Bayesian update of a prior, which can be written as

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{x})p(\mathbf{y}|\mathbf{x}), \tag{50}$$

where $p(x|y)$ is the posterior, $p(x)$ is the prior, and $p(y|x)$ is the measurement likelihood. If two measurements are conditionally independent, the combined measurement likelihood is the product of likelihoods

$$p(\mathbf{y}_1, \mathbf{y}_2|\mathbf{x}) = p(\mathbf{y}_1|\mathbf{x})p(\mathbf{y}_2|\mathbf{x}). \tag{51}$$

Thus, the update of prior with two conditionally independent measurements is $p(\mathbf{x}|\mathbf{y}_1, \mathbf{y}_2) \propto p(\mathbf{x})p(\mathbf{y}_1|\mathbf{x})p(\mathbf{y}_2|\mathbf{x})$, which can be done with two separate updates, first $p(\mathbf{x}|\mathbf{y}_1) \propto p(\mathbf{x})p(y_1|\mathbf{x})$ and then, using this as the prior for the next update, $p(\mathbf{x}|\mathbf{y}_1, \mathbf{y}_2) \propto p(\mathbf{x}|\mathbf{y}_1)p(\mathbf{y}_2|\mathbf{x})$. Thus, the measurements can be applied one at a time. Of course, when using an approximate update method, such as the UKF, the result is not exactly the same because there are approximation errors involved. Processing measurements separately can improve the accuracy, e.g., when $\mathbf{y}_1$ is a linear measurement, then the prior for the second measurement becomes smaller and thus the effect of its nonlinearity is reduced.

4.3 Time series filtering

So far, we have discussed the update of a prior component with a measurement at a single time step. For the time series estimation, the filter requires in addition to Algorithms 1 and 2 a prediction step and a component reduction step.

If the state model is linear, the predicted mean of a component is $\mu_{(t)} = F\mu^{+}_{(t-1)}$, where F is the state transition matrix and $\mu^{+}_{(t-1)}$ is the posterior mean of the previous time step. The predicted covariance is $P_{(t)} = FP^{+}_{(t-1)}F^T + W$, where W is the covariance of the state transition error. The weights of components do not change in the prediction step. If the state model is nonlinear, a sigma point approach can be also used for the prediction [3].

For component reduction, we use the measure proposed in [5]

$$B_{i,j} = \frac{1}{2}\left[(w_i + w_j)\log\det P_{i,j} - w_i \log\det P_i - w_j \log\det P_j\right], \tag{52}$$

where $P_{i,j} = \frac{w_i P_i + w_j P_j}{w_i + w_j} + (\mu_i - \mu_j)(\mu_i - \mu_j)^T$. Whenever the number of components is larger than m_{reduce} or $B_{i,j} < B_{\text{limit}}$, the component pair that has the smallest $B_{i,j}$ is merged so that the mean and covariance of the mixture is preserved. The test for B_{limit} is our own addition to the algorithm to allow the number of components to be reduced below m_{reduce} if there are very similar components.

When the prior consists of multiple components, the value of m_{limit} for each component is chosen proportional to the prior weight so that they sum up to the total limit of components. Algorithm 3 shows the BinoGMF algorithm. A MATLAB implementation of BinoGMF is also available [see Additional file 1].

Algorithm 3: Binomial Gaussian mixture filter

for k=1,...,m **do**
  Predict components
end
Compute $m_{k,\text{limit}}$ for each component
$\hat{\mathbf{y}} \leftarrow L(R)^{-1}y$ `// Make measurement error covariance unit diagonal by multiplying` d`-dimensional measurement` $\mathbf{y}$ `with` $L(R)^{-1}$
$\hat{h}(\mathbf{x}) \leftarrow L(R)^{-1}h(\mathbf{x})$ `// Define decorrelated measurement functions`
for i=1,...,d **do**
  for k=1,...,m **do**
    Compute $L(P_0)$ for kth prior component
    Use $L(P_0)$ and $\hat{h}_{[i]}(x)$ to compute $\hat{Q}_i$ `// (47)`
    if $tr\hat{Q}_i\hat{Q}_i \geq \eta_{limit}$ **then**
      Use Algorithm 2 to split the component
      Update resulting components with UKF, Section 2.1
      Update component weights (9)
    else
      Update component with UKF, Section 2.1
      Update component weight (9)
    end
  end
  Normalize weights
  Reduce components
end

5 Tests

We evaluate the performance of the proposed BinoGMF method in two simulation settings. First, the splitting method is compared with other GM splitting algorithms in single prior component single measurement cases, and then BinoGMF is compared with a particle filter in a time series scenario. In all cases, a single-component UKF is used as a reference.

5.1 Comparison of GM splitting methods

To compare the accuracy of the proposed BinoGMF with other GM splitting methods presented in Section 2, we update a one-component prior with a nonlinear measurement. The accuracy of the estimate is evaluated with two measures: mean residual and the Kullback-Leibler divergence (48). The mean residual is the absolute value of the difference of the mean of the posterior estimate and the mean of the true posterior. The integrals involved in computing results were computed using Monte Carlo integration. Each simulation case was repeated 100 times. Cases are as follows:

- 2D range - a range measurement in two-dimensional positioning
- 2D second order - the measurement consists of a second-order polynomial term aligned with a random direction and a linear measurement term aligned with another random direction
- 4D speed - a speedometer measurement with a highly uncertain prior
- 10D third order - a third-order polynomial measurement along a random direction in a ten-dimensional space

For computation of sigma points and UKF updates, we used the parameter values $\alpha_{\mathrm{UKF}} = 0.5$, $\kappa_{\mathrm{UKF}} = 0$, and $\beta_{\mathrm{UKF}} = 2$. Detailed parameters of different cases are presented in Appendix 4.

For comparison, we use a UKF that estimates the posterior with only one component and the GMFs presented in Section 2. In our implementation, there are some minor differences with the original methods:

- For H&C, the one-dimensional mixture is optimized with a different method. This should not have a significant effect on results.
- For F&K, [7] gives two alternatives for choosing the number of components and for computing the direction of nonlinearity. We chose to use split into three components, and the direction of nonlinearity was computed using the eigendecomposition.
- In AS, splitting is not done recursively; instead, every split is done to the component with highest nonlinearity. This is done to get the desired number of components. This probably makes the estimate more accurate than with the original method.

Our implementation of UKF, H&C, F&K, and S&M might have some other inadvertent differences from the original implementations.

The nonlinearity-based stopping criterion is not tested in this paper. Splitting is done with a fixed upper limit on the number of components. We chose to test the methods with at most 3, 9, and 81 components for consistency with the number of components in BGMF. The number of components in BGMF is $(2N^2 + 1)^n$, where $N \in \mathbb{N}$ is a parameter that adjusts the number of components. Since BinoGMF does the splitting into multiple directions at once, the number of components of BinoGMF is usually less than the maximum limit.

Figure 5 shows the 25%, 50%, and 75% quantiles for each tested method in different cases. The figure shows how the proposed method has the best posterior approximation accuracy with a given number of components, except in the 4D speed case where AS has slightly better accuracy. This is due to large variations of the Hessian H within the region where the prior pdf is significant. In the 2D range case, the posterior accuracy of AS and S&M does not improve when the number of components is increased from 9 to 81. This is caused by prior approximation error as the AS and S&M splitting does not converge to the prior.

In Table 1, the estimation of direction nonlinearity is tested in a two-dimensional case where the prior has identity covariance matrix and the measurement function is

$$\mathbf{y} = e^{\mathbf{a}^T \mathbf{x}}. \tag{53}$$

The true direction of nonlinearity is $\mathbf{a}$. AS and BinoGMF use the same algorithm for computing the direction of maximum nonlinearity. S&M uses the maximum eigenvalue of the prior component as the split direction, and in this case, it gives the same results as choosing the direction randomly. In the table, $\Delta\theta$ is the average error of the direction of maximum nonlinearity.

In two dimensions, choosing a random direction for splitting has a mean error of 45 degrees. The good performances of the proposed method and AS in Figure 5 are partly due to the algorithm for estimating the direction of maximum nonlinearity, because they have clearly the most accurate nonlinearity direction estimates. The 0.8 degree error of AS and BinoGMF could be reduced by choosing sigma points closer to the mean, but this would make the estimate more local. The BinoGMF was also tested with $\alpha_{\mathrm{UKF}} = 10^{-3}$. This resulted in splitting direction estimates closer than 10^{-5} degrees, but there was significant accuracy drop in the 4D speed case. With a

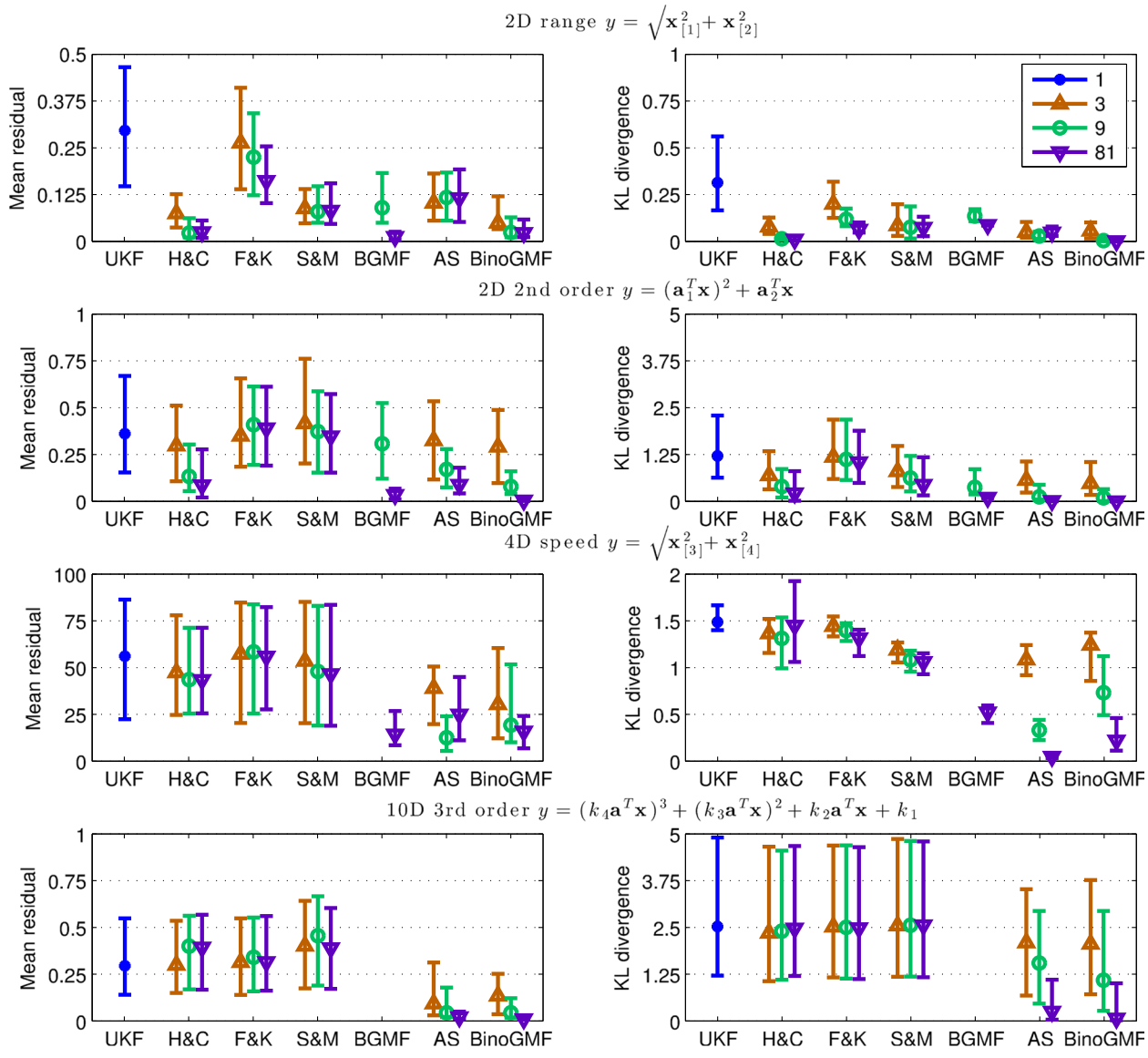

Figure 5 Mean residuals and estimated mean and corresponding Kullback-Leibler divergences.

small value of α_{UKF}, the computation of Q does not take variations of the Hessian into account. This causes fewer problems with AS, because AS splits only in the direction of maximum nonlinearity and then re-evaluates the nonlinearity for the resulting components.

Table 2 shows the number of measurement function evaluations of each method assuming that the update is done with UKF using a symmetric set of sigma points.

Table 1 Average error on estimation of the splitting direction in degrees

	Random	F&K	H&C	BinoGMF
$\Delta\theta$	45	22	23	0.8

BinoGMF does fewer measurement function evaluations than AS; it does significantly more than other methods only if $n \gg m$. Because BinoGMF evaluates the nonlinearity and does the splitting all at once, the computationally

Table 2 Number of measurement function evaluations

Method	Evaluations	Order
H&C	$(m+1)(2n-1)$	$\mathcal{O}(mn)$
F&K	$\frac{(m+1)}{2}(2n+1)$	$\mathcal{O}mn)$
S&M	$(2m-1)(2n-1)$	$\mathcal{O}(mn)$
BGMF	$m(2n+1)$	$\mathcal{O}(mn)$
AS	$(m-1)(n^2+n)+2n+1$	$\mathcal{O}(mn^2)$
BinoGMF	$\frac{n^2+n}{2}+m(2n+1)$	$\mathcal{O}(n^2+mn)$

most expensive part of the algorithm is done only once and the update of the mixture can be done with UKF. Also, because the resulting components of a split have equal covariance matrix $P = TT^T$, $T(44)$ can be used as the matrix square root for UKF for all components.

5.2 Time series filtering

In the time series evaluation, we simulate the navigation of an aircraft. The route does circular loops with a radius of 90 m if seen from above. In the vertical dimension, the aircraft first ascends until it reaches 100 m, flies at constant altitude for a while, and finally descends. In the simulation, there are five ground stations that emit signals that the aircraft receives.

There are two kinds of measurements available:

- Time of arrival (TOA) [16]

$$\mathbf{y} = \|\mathbf{r}_{\text{receiver}} - \mathbf{r}_{\text{emitter}}\| + \delta_{\text{receiver}} + \varepsilon_{\text{receiver}} + \varepsilon, \tag{54}$$

  where $\mathbf{r}_{\text{receiver}}$ is the aircraft location, $\mathbf{r}_{\text{emitter}}$ is the location of a ground station, δ_{receiver} is the receiver clock bias, $\varepsilon_{\text{receiver}}$ is an error caused by receiver clock jitter that is the same for all measurements, and ε is the independent error. The covariance of a set of TOA measurements is

$$R_{\text{TOA}} = I + 5^2\mathbf{1}, \tag{55}$$

  where $\mathbf{1}$ is a matrix of ones.
- Doppler [17]

$$\mathbf{y} = \frac{\mathbf{v}^T_{\text{receiver}}(\mathbf{r}_{\text{receiver}} - \mathbf{r}_{\text{emitter}})}{\|\mathbf{r}_{\text{receiver}} - \mathbf{r}_{\text{emitter}}\|} + \gamma_{\text{receiver}} + \varepsilon_{\text{receiver}} + \varepsilon, \tag{56}$$

  where $\mathbf{v}_{\text{receiver}}$ is the velocity of the aircraft and γ_{receiver} is the clock drift. The Doppler measurement error covariance used is

$$R_{\text{Doppler}} = I + \mathbf{1}. \tag{57}$$

The probability of receiving a measurement is dependent on the distance from a ground station, and a TOA measurement is available with 50% probability when a Doppler measurement is available. The estimated state contains eight variables, three for position, three for velocity, and one for each clock bias and drift. The state model used with all filters is linear. State model, true track, and ground station parameters can be found in Appendix 4.

We compare BinoGMF with three different parameter sets with UKF, a bootstrap particle filter (PF) that uses systematic resampling [18,19] with different numbers of particles, and a Rao-Blackwellized particle filter (RBPF) that is implemented according to [20]. The Rao-Blackwellized particle filter exploits the fact that when conditioned with position variables, the measurement functions become linear-Gaussian with respect to the remaining state variables. Consequently, position variables are estimated with particles, and for the remaining variables, there is a normal distribution attached to every particle. BinoGMF parameters are presented in Table 3.

Figure 6 shows 5%, 25%, 50%, 75%, and 95% quantiles of mean position errors of 1,000 runs. Because some errors are large, on the left, there are plots that show a maximum error of 200 m, and on the right, the maximum error shown is 20 m. In some test cases, all particles end up having zero weights due to finite precision in computation. In these cases, the particle filters are considered to have failed. This causes lines in the plot to end at some point, e.g., with 100 particles, the RBPF fails in more than in 50% of cases.

The UKF and BinoGMF results are located according to their time usage compared to the time usage of PFs. The figure shows how BinoGMF achieves better positioning accuracy with smaller time usage than the PFs. The 95% quantile of UKF and BinoGMF4 is more than 150 m, which is caused by multimodality of the posterior. In these cases, UKF and BinoGMF4 follow only wrong modes. The RBPF has better accuracy than the PF with a similar number of particles, but is slower. In our implementation, updating of one Rao-Blackwellized particle is 6 to 10 times faster than the UKF update depending on how many particles are updated. The RBPF requires much more particles than BinoGMF requires components. The bootstrap PF is faster than the RBPF, because our MATLAB implementation of the bootstrap PF is highly optimized.

The locations of the ground stations are almost coplanar, which causes the geometry of the test situation to be such that most of the position error is in the altitude dimension. Figure 7 shows an example of the altitude estimates given by BinoGMF16, bootstrap PF with 10^4 particles, RBPF with 100 particles, and UKF. Both PFs are slower than the BinoGMF16.

The figure shows how the PF and BinoGMF16 estimates are multimodal at several time steps, but most of the time, BinoGMF16 has more weight on the correct mode. The RBPF starts in the beginning to follow a wrong mode and does not recover during the whole test. The UKF estimate starts out somewhere between the modes, and it takes a while to converge to the correct mode. UKF could also converge to a wrong node. In multimodal situations as in

Table 3 Parameters used in filtering test for BinoGMF

	m_{total}	m_{reduce}	η_{limit}	B_{limit}
BinoGMF4	4	2	4	0.1
BinoGMF16	16	4	1	0.01
BinoGMF64	64	16	0.25	0.001

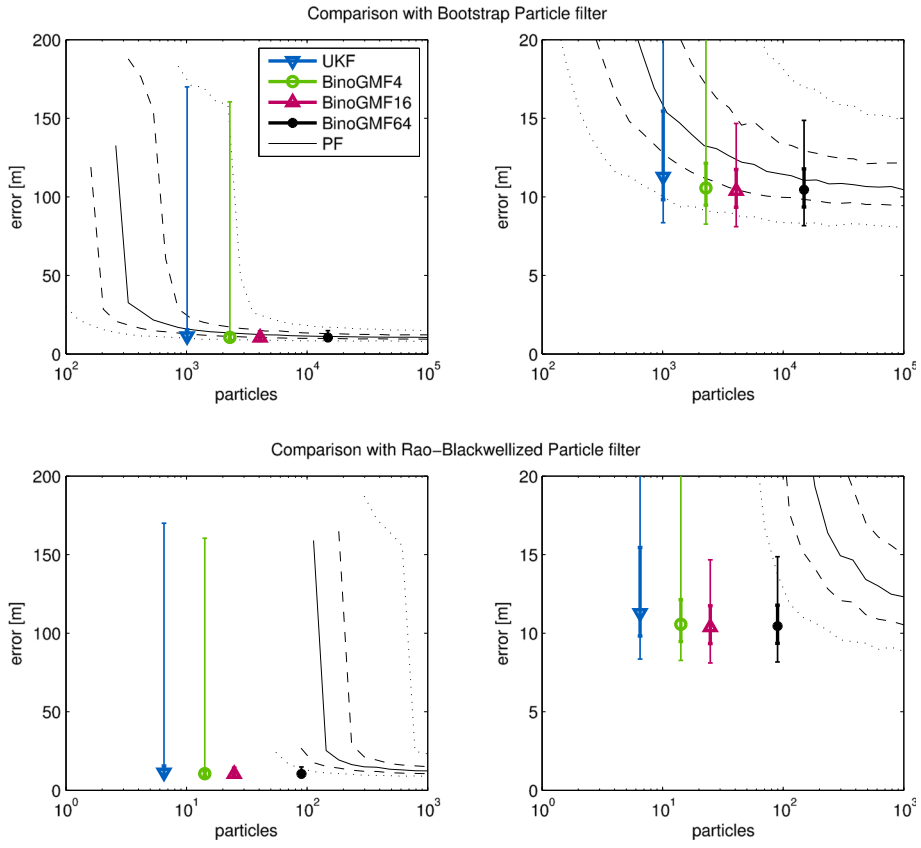

Figure 6 Position estimate error comparison between BinoGMF, PF, RBPF, and UKF.

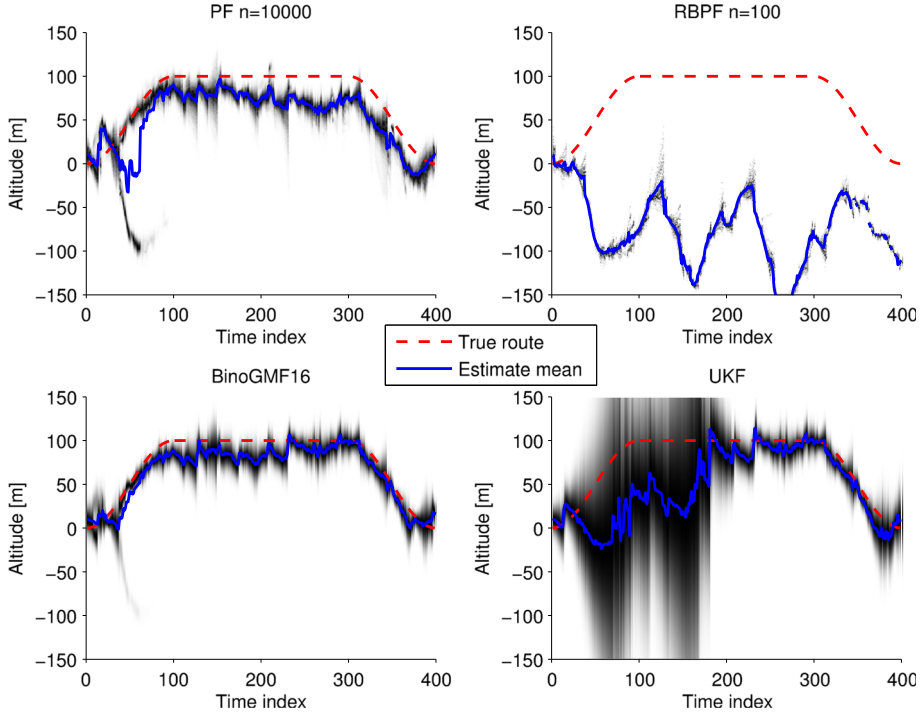

Figure 7 Altitude estimates of BinoGMF16, PF, RBPF, and UKF.

Figure 7, the comparison of the accuracy of the mean to the true route is not necessarily so relevant, e.g., in PF at time step 70, the mean is located in a low probability area of the posterior.

This simulated test showed that there are situations where the BinoGMF can outperform PFs. We found also that if the state transition model noise (87) was made smaller without changing the true track, then the number of required particles increased fast, while the effect on BinoGMF was small.

6 Conclusions

In this paper, we have presented the BinoGMF. BinoGMF uses a binomial distribution in the generation of a GM from a normal distribution. It was shown that the pdf and cdf of the resulting mixture converge to the prior when the number of components is increased. Furthermore, we presented an algorithm for choosing the component size so that the nonlinearity is not too high, the resulting mixture is a good approximation of the prior, and the number of required components is minimized.

We compared the proposed method with UKF and five different GMFs in several single-step estimation simulation cases and with UKF and PF in a time series estimation scenario. In these tests, the proposed method outperforms other GM-based methods in accuracy, while using a similar amount or fewer measurement function evaluations. In filtering, BinoGMF provided more accurate estimates faster than bootstrap PF or Rao-Blackwellized PF. BinoGMF can be used in suitable situations instead of PFs to get better estimation accuracy, if the measurement error can be modeled as additive and normally distributed.

Because BinoGMF performed well in all tests, we recommend it to be used instead of other GM splitting methods to get better estimation accuracy. It performs especially well in situations where there is more than a few dimensions and in cases where it is essential to have an accurate estimate of the posterior pdf.

Appendices

Appendix 1: Determining the component distance

Consider two equally weighted unit variance one-dimensional components located at an equal distance $\Delta\mu$ from the origin, with pdf

$$f(x) = \frac{w}{\sqrt{2\pi}} e^{-\frac{(x-\Delta\mu)^2}{2}} + \frac{w}{\sqrt{2\pi}} e^{-\frac{(x+\Delta\mu)^2}{2}}. \tag{58}$$

Due to symmetry, this function has zero slope at the origin and so there is either a local minimum or maximum at the origin. Function (58) is unimodal when the origin is a maximum and bimodal if it is a local minimum. The function has a local minimum at the origin if $f''(0) > 0$. The second derivative of (58) is

$$f''(x) = \frac{w}{\sqrt{2\pi}} \left[\left((\Delta\mu - x)^2 - 1 \right) e^{-\frac{(x-\Delta\mu)^2}{2}} + \left((\Delta\mu + x)^2 - 1 \right) e^{-\frac{(x+\Delta\mu)^2}{2}} \right]. \tag{59}$$

Evaluating this, we get the rule for having a local maximum at origin as

$$\begin{aligned} f''(0) \leq 0 \Leftrightarrow \\ \frac{w}{\sqrt{2\pi}} \left[(\Delta\mu^2 - 1) e^{-\frac{\Delta\mu^2}{2}} + (\Delta\mu^2 - 1) e^{-\frac{\Delta\mu^2}{2}} \right] \leq 0 \Leftrightarrow \\ \Delta\mu^2 - 1 \leq 0 \Leftrightarrow \\ \Delta\mu \leq 1 \end{aligned} \tag{60}$$

Consider the mean of the kth component of a mixture generated using the standardized binomial distribution scaled with σ

$$\mu_{\mathbf{x}_k} = \sigma \frac{2k - m - 1}{\sqrt{m-1}} \tag{61}$$

The distance between two components is

$$2\Delta\mu = \sigma \frac{2(k+1) - m - 1}{\sqrt{m-1}} - \sigma \frac{2k - m - 1}{\sqrt{m-1}} = \frac{2\sigma}{\sqrt{m-1}}. \tag{62}$$

Using (60), we get

$$m \geq \sigma^2 + 1. \tag{63}$$

Appendix 2: Optimization of mixture parameters

In [9], it was shown that if a component covariance P is computed as

$$P = P_0 - \beta L(P_0) V e_i e_i^T V^T L(P_0)^T, \tag{64}$$

where e_i is the ith column of the identity matrix and V and Λ are computed from the eigendecomposition

$$V \Lambda V^T = L(P_0)^T H L(P_0)^T, \tag{65}$$

where H is the Hessian of the measurement function, then the nonlinearity associated with direction $L(P_0)Ve_i$ changes from λ_i^2 to $(1-\beta)^2\lambda_i^2$.

It was also shown that the eigenvectors of (65) do not change in this reduction. Due to this, we can also do multiple reductions simultaneously

$$\begin{aligned} P &= P_0 - L(P_0) V B V^T L(P_0)^T \\ &= L(P_0) V (I - B) V^T L(P_0)^T, \end{aligned} \tag{66}$$

where B is a diagonal matrix having $\beta_1, \ldots, \beta_n$ on its diagonal.

A solution to (37) is to compute an eigendecomposition

$$U^T \Sigma U = L(P)^{-1} (P_0 - P) L(P)^{-T} \tag{67}$$

and choose

$$T = L(P) U. \tag{68}$$

We use the following matrix square root

$$L(P) = L(P_0)V\text{diag}\left(\sqrt{1-\beta_1},\ldots,\sqrt{1-\beta_n}\right). \quad (69)$$

Using (66) and (69) in (67) produces

$$\begin{aligned} &L(P)^{-1}(P_0-P)L(P)^{-T} \\ &= \sqrt{I-B}^{-1}V^T L(P_0)^{-1}L(P_0)VBV^T L(P_0)^T \\ &\quad L(P_0)^{-T}V\sqrt{I-B}^{-1} \\ &= \sqrt{I-B}^{-1}B\sqrt{I-B}^{-1} = (I-B)^{-1}B = I\Sigma I. \end{aligned} \quad (70)$$

From this, we get

$$\sigma_i^2 = \frac{\beta_i}{1-\beta_i} \quad (71)$$

and using relationship (40)

$$m_i = \sigma_i^2 + 1 = \frac{1}{1-\beta_i} \Leftrightarrow 1-\beta_i = \frac{1}{m_i}. \quad (72)$$

Now T (37) can be written using (69) and (72) as

$$T = L(P_0)V\text{diag}\left(\frac{1}{\sqrt{m_1}},\ldots,\frac{1}{\sqrt{m_n}}\right). \quad (73)$$

The nonlinearity (41) after multiple reductions can be written as

$$\eta = \frac{\text{tr}\,PHPH}{R} = \frac{\sum(1-\beta_i)^2\lambda_i^2}{R} \quad (74)$$

The problem is to find B so that the nonlinearity decreases below a given threshold, i.e.,

$$\frac{\sum(1-\beta_i)^2\lambda_i^2}{R} \leq \eta_{\text{limit}} \quad (75)$$

while the total number of components is as low as possible. Substituting (72) into (75), the nonlinearity becomes

$$\eta = \sum\frac{(1-\beta_i)^2\lambda_i^2}{R} = \frac{1}{R}\sum\frac{\lambda_i^2}{m_i^2}. \quad (76)$$

The optimization problem is to minimize

$$\prod_{i=1}^{n} m_i \quad (77)$$

with constraints

$$\sum_{i=1}^{n}\frac{\lambda_i^2}{m_i^2} \leq R\eta_{\text{limit}} \quad (78a)$$

$$m_i \geq 1, \forall i : 1 \leq i \leq n. \quad (78b)$$

The Karush-Kuhn-Tucker conditions [21] for this problem are

$$\begin{aligned} 0 &= \nabla\prod m_i + \xi_0\nabla\left(\sum\frac{\lambda_i^2}{m_i^2} - R\eta_{\text{limit}}\right) \\ &\quad + \sum\nabla\xi_i(1-m_i). \end{aligned} \quad (79)$$

The jth partial derivative is

$$0 = \prod_{i\neq j} m_i - 2\xi_0\frac{\lambda_j^2}{m_j^3} - \xi_j \Leftrightarrow \prod m_i = 2\xi_0\frac{\lambda_j^2}{m_j^2} + \xi_j m_j. \quad (80)$$

Due to the complementary slackness requirement, either $m_j = 1$ or $\xi_j = 0$. For all $m_j \neq 1 \neq m_i$

$$\frac{\lambda_i^2}{m_i^2} = \frac{\lambda_j^2}{m_j^2}. \quad (81)$$

This means that if the integer nature of m_i is neglected, the optimum m_i is either 1 or proportional to $|\lambda_i|$.

Appendix 3: UKF update after a linear transformation is applied to measurement function

Let A be an invertible matrix and $\mathcal{X}_i$ UKF sigma points computed as explained in Section 2.1. The measurement transformed with A is

$$\hat{\mathbf{y}} = A\mathbf{y} = Ah(\mathbf{x}) + A\varepsilon. \quad (82)$$

If the original measurement error covariance is R, then the transformed measurement error covariance is $\hat{R} = ARA^T$.

The transformed sigma points are:

$$\hat{\mathcal{Y}}_i = Ah(\mathcal{X}_i). \quad (83)$$

The update becomes

$$\begin{aligned} \hat{\mathbf{z}} &= \sum_{i=0}^{2n}\Omega_{i,m}A\mathcal{Y}_i = A\sum_{i=0}^{2n}\Omega_{i,m}\mathcal{Y}_i = A\mathbf{z} \\ \hat{S} &= ARA^T + \sum_{i=0}^{2n}\Omega_{i,c}\,(A\mathcal{Y}_i - A\mathbf{z})\,(A\mathcal{Y}_i - A\mathbf{z})^T = ASA^T \\ \hat{C} &= \sum_{i=0}^{2n}\Omega_{i,c}\,(\mathcal{X}_i - \mu)\,(A\mathcal{Y}_i - A\mathbf{z})^T = CA^T, \\ \hat{K} &= \hat{C}\hat{S}^{-1} = CA^TA^{-T}S^{-1}A^{-1} = KA^{-1} \\ \hat{\mu}^+ &= \mu + \hat{K}(A\mathbf{y} - A\mathbf{z}) = \mu + K(\mathbf{y} - \mathbf{z}) = \mu^+ \\ \hat{P}^+ &= P - \hat{K}\hat{S}\hat{K}^T = P - KA^{-1}ASA^{-T}K^T = P - KSK^T = P^+ \end{aligned} \quad (84)$$

and the posterior estimate is identical to the estimate computed with nontransformed measurement function.

Appendix 4: Simulation parameters

GMF comparison

The simulation parameters used to evaluate different methods for Figure 5 are below. The measurement variance was 1 in every case.

- 2D range
  - Prior mean: $[5\ 0]^T$

  - Prior covariance: $\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$, where $\theta \sim U(0, 2\pi)$
  - Measurement function: $\sqrt{\mathbf{x}_{[1]}^2 + \mathbf{x}_{[2]}^2}$

- 2D second order
  - Prior mean: $\sim N(0, I)$
  - Prior covariance: $\begin{bmatrix} 5 & 2 \\ 2 & 5 \end{bmatrix}$
  - Measurement function: $\left(\mathbf{a}_1^T \mathbf{x}\right)^2 + \mathbf{a}_2^T \mathbf{x}$, where $\mathbf{a}_1 \sim N(0, I)$ and $\mathbf{a}_2 \sim N(0, I)$

- 4D speed
  - Prior mean: $\sim N\left(0, \begin{bmatrix} \frac{100^3}{3} I & \frac{100^2}{2} I \\ \frac{100^2}{2} I & \frac{100^2}{2} I \end{bmatrix}\right)$
  - Prior covariance: $\begin{bmatrix} \frac{100^3}{3} I & \frac{100^2}{2} I \\ \frac{100^2}{2} I & \frac{100^2}{2} I \end{bmatrix}$
  - Measurement function: $\sqrt{\mathbf{x}_{[3]}^2 + \mathbf{x}_{[4]}^2}$

- 10D third order
  - Prior mean: $\sim N(0, I)$
  - Prior covariance: I
  - Measurement function: $\left(k_4 \mathbf{a}^T \mathbf{x}\right)^3 + \left(k_3 \mathbf{a}^T \mathbf{x}\right)^2 + k_2 \mathbf{a}^T \mathbf{x} + k_1$, where $k_i \sim N(0, 1)$ $(i = 1, \ldots, 4)$ and $\mathbf{a} \sim N(0, I)$

Filtering example

State model used in filters:

$$\mathbf{x}_{t+1} = F\mathbf{x}_t + \varepsilon_W, \tag{85}$$

where

$$F = \begin{bmatrix} I_{3\times3} & I_{3\times3} & 0_{3\times2} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times2} \\ 0_{2\times3} & 0_{2\times3} & \begin{matrix} 1 & 1 \\ 0 & 1 \end{matrix} \end{bmatrix} \tag{86}$$

and

$$\varepsilon_W = \begin{bmatrix} I_{3\times3} & \frac{1}{2} I_{3\times3} & 0_{3\times2} \\ \frac{1}{2} I_{3\times3} & \frac{1}{3} I_{3\times3} & 0_{3\times2} \\ 0_{2\times3} & 0_{2\times3} & \begin{matrix} \frac{1}{100} & \frac{1}{200} \\ \frac{1}{200} & \frac{1}{300} \end{matrix} \end{bmatrix}. \tag{87}$$

The track is simulated on time indices from 1 to 400 and the true track is:

- Position:

$$\mathbf{r}_i^{\text{true}} = \begin{bmatrix} 90\sin\theta_i \\ 90\cos\theta_i \\ h_i \end{bmatrix}, \tag{88}$$

where $\theta_i = \frac{2i\pi}{100}$ and

$$h_i = \begin{cases} 50 + 50\cos\frac{\theta_i + 2\pi}{2} & , i \leq 100 \\ 100 & , 100 < i \leq 300 \\ 50 + 50\cos\frac{\theta_i - 2\pi}{2} & , 300 < i \end{cases} \tag{89}$$

- Velocity: $\mathbf{v}_i^{\text{true}} = r_{i+1}^{\text{true}} - r_i^{\text{true}}$
- Bias: $\delta_i = 10\cos\frac{\theta_i}{3}$
- Drift: $\gamma_i = \delta_{i+1} - \delta_i$

Ground station locations:

$$\begin{bmatrix} -100 \\ 0 \\ -10 \end{bmatrix}, \begin{bmatrix} 100 \\ 0 \\ 20 \end{bmatrix}, \begin{bmatrix} 0 \\ -100 \\ -20 \end{bmatrix}, \begin{bmatrix} 0 \\ 100 \\ 10 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{90}$$

Additional file

Additional file 1: binogmf.m–a MATLAB implementation of the BinoGMF algorithm.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work was supported by Tampere Doctoral Programme in Information Science and Engineering, Nokia Corporation, Nokia Foundation, and Jenny and Antti Wihuri Foundation. The funding sources were not involved in the preparation of this article. The simulations were carried out using the computing resources of CSC - IT Center for Science.

Author details
[1]Department of Automation Science and Engineering, Tampere University of Technology, P.O. Box 692, FI-33101 Tampere, Finland. [2]Department of Mathematics, Tampere University of Technology, P.O. Box 553, FI-33101 Tampere, Finland.

Received: 16 March 2014 Accepted: 20 March 2015
Published online: 11 April 2015

References

1. HW Sorenson, DL Alspach, Recursive Bayesian estimation using Gaussian sums. Automatica. **7**(4), 465–479 (1971)
2. S Ali-Löytty, Box Gaussian mixture filter. IEEE Trans. Autom. Control. **55**(9), 2165–2169 (2010). doi:10.1109/TAC.2010.2051486
3. SJ Julier, JK Uhlmann, HF Durrant-Whyte, in *American Control Conference.* A new approach for filtering nonlinear systems, vol. 3 (Seattle, WA, USA, 21–23 June 1995), pp. 1628–1632
4. EA Wan, R Van Der Merwe, in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC.* The unscented Kalman filter for nonlinear estimation (Lake Louise, AB, Canada, 1–4 October 2000), pp. 153–158. doi:10.1109/ASSPCC.2000.882463
5. AR Runnalls, Kullback-Leibler approach to Gaussian mixture reduction. IEEE Trans. Aerospace Electron. Syst. **43**(3), 989–999 (2007). doi:10.1109/TAES.2007.4383588
6. F Havlak, M Campbell, Discrete and continuous, probabilistic anticipation for autonomous robots in urban environments. IEEE Trans. Rob. **PP**(99), 1–14 (2013). doi:10.1109/TRO.2013.2291620
7. F Faubel, D Klakow, in *Proc. Europ. Sig. Process. Conf. (EUSIPCO).* Further improvement of the adaptive level of detail transform: splitting in direction of the nonlinearity (Aalborg, Denmark, 23)
8. F Faubel, J McDonough, D Klakow, The split and merge unscented Gaussian mixture filter. Signal Process. Lett., IEEE. **16**(9), 786–789 (2009). doi:10.1109/LSP.2009.2024859
9. M Raitoharju, S Ali-Löytty, An adaptive derivative free method for Bayesian posterior approximation. Signal Process. Lett., IEEE. **19**(2), 87–90 (2012). doi:10.1109/LSP.2011.2179800

10. AC Berry, The accuracy of the Gaussian approximation to the sum of independent variates. Trans. Am. Math. Soc. **49**(1), 122–136 (1941)
11. W Feller, On the normal approximation to the binomial distribution. Ann. Math. Stat. **16**(4), 319–329 (1945). doi:10.1214/aoms/1177731058
12. P Billingsley, *Convergence of Probability Measures. Wiley Series in Probability and Statistics*. (Wiley, New York, 2009)
13. D McLeish, *STAT 901: Probability. Lecture Notes*. (University of Waterloo, Waterloo, ON, Canada, 2005)
14. DD Boos, A converse to Scheffe's theorem. Ann. Stat, 423–427 (1985)
15. S Kullback, RA Leibler, On information and sufficiency. Ann. Math. Stat. **22**(1), 79–86 (1951)
16. F Gustafsson, F Gunnarsson, Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements. Signal Process. Mag., IEEE. **22**(4), 41–53 (2005). doi:10.1109/MSP.2005.1458284
17. D Borio, N Sokolova, G Lachapelle, in *Proc., ION/GNSS*. Doppler measurements and velocity estimation: a theoretical framework with software receiver implementation, vol. 9 (Savannah, GA, USA, 22), pp. 304–316
18. NJ Gordon, DJ Salmond, AFM Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation. Radar Signal Process. IEE Proc. F. **140**(2), 107–113 (1993)
19. B Ristic, S Arulampalam, N Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. (Artech House, Boston, 2004)
20. G Hendeby, R Karlsson, F Gustafsson, The Rao-Blackwellized particle filter: a filter bank implementation. EURASIP J Adv. Signal Process. **2010**, 724087 (2010). doi:10.1155/2010/724087
21. HW Kuhn, AW Tucker, in *Second Berkeley Symposium on Mathematical Statistics and Probability*. Nonlinear programming, vol. 1 (Berkeley, CA, USA, 12 August 1951 31 July), pp. 481–492

Submit your manuscript to a SpringerOpen journal and benefit from:

► Convenient online submission
► Rigorous peer review
► Immediate publication on acceptance
► Open access: articles freely available online
► High visibility within the field
► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com