

THE MARGINALIZED PARTICLE FILTER – ANALYSIS, APPLICATIONS AND GENERALIZATIONS

THOMAS B. SCHÖN, RICKARD KARLSSON AND FREDRIK GUSTAFSSON¹

Abstract. The marginalized particle filter is a powerful combination of the particle filter and the Kalman filter, which can be used when the underlying model contains a linear sub-structure, subject to Gaussian noise. This paper outlines the marginalized particle filter and very briefly hint at possible generalizations, giving rise to a larger family of marginalized nonlinear filters. Furthermore, we analyze several properties of the marginalized particle filter, including its ability to reduce variance and its computational complexity. Finally, we provide an introduction to various applications of the marginalized particle filter.

1. INTRODUCTION

Many problems in for instance positioning and target tracking can be cast as nonlinear state estimation problems, where the uncertainty in the process model and/or in the measurement model may be non-Gaussian. Such a general model can be formulated according to

$$x_{t+1} = f(x_t, u_t) + w_t, \quad (1a)$$

$$y_t = h(x_t) + e_t, \quad (1b)$$

with state variable $x_t \in \mathbb{R}^m$, input signal u_t and measurements $Y_t = \{y_i\}_{i=1}^t$, with known *probability density function* (PDF) for the process noise $p_w(w)$ and the measurement noise $p_e(e)$. Hence, traditional estimation methods based on the *Kalman filter* (KF) [19, 20], or linearized version thereof, do not always provide good performance. Over the past 40 years there has been several suggestions on how to tackle the problem of estimating the states in (1). An appealing solution is provided by the *particle filter* (PF) [9, 17, 30], which allows for a systematic treatment of both nonlinearities and non-Gaussian noise. However, due to the inherent computational complexity of the particle filter, real-time issues arise in many applications when the sampling rate is high. Furthermore, the particle filter only works for moderate state dimensions, when the state dimension is large, something else is required. If the model includes a sub-structure with linear equations, subject to Gaussian noise, it is possible to exploit this in the estimator. Here, this method is referred to as the *marginalized particle filter* (MPF), it is also known as the Rao-Blackwellized particle filter, see for instance [2, 3, 6, 9, 10, 33]. The MPF is a combination of the standard particle filter and the Kalman filter. It is a well known fact that in some cases it is possible to obtain better estimates, i.e., estimates with reduced variance, using the marginalized particle filter instead of using the standard particle filter [12].

¹ Division of Automatic Control
Department of Electrical Engineering
Linköping University, Sweden
SE-581 83 Linköping, Sweden

It is the linear, Gaussian sub-structure that opens up for the use of the marginalized particle filter. Hence, it is a structural property of the underlying model. It is interesting to note that this property can be exploited in combination with other nonlinear filters as well, resulting in a rather general class of filters, which we will refer to as *marginalized nonlinear filters* (MNLF).

The aim of this paper is to introduce the the marginalized particle filter and very briefly hint at possible extensions. We will also provide an overview of some of the applications where the marginalized particle filter has been successfully applied. Since we cannot cover all the details in this paper references to more detailed treatments are provided.

2. MARGINALIZED NONLINEAR FILTERS

The aim of recursively estimating the filtering density $p(x_t|Y_t)$ can be accomplished using the standard particle filter. However, if there is a linear sub-structure, subject to Gaussian noise, present in the model this can be exploited to obtain better estimates and possibly reduce the computational demand as well. This, together with the fact that the standard particle filter might be prohibited due to a too high state dimension, constitutes the motivation underlying the marginalized nonlinear filters.

2.1. Representation

The task of nonlinear filtering can be split into two parts: representation of the filtering probability density function and propagation of this density during the time and measurement update stages. Figure 1 illustrates different representations of the filtering density for a two-dimensional example (similar to the example used in Section 4). The *extended Kalman filter* (EKF) [1,19], can be interpreted as using one Gaussian distribution for representation and the propagation is performed according to a linearized model. The *Gaussian sum filter* [1,35] extends the EKF to be able to represent multi-modal distributions, still with an approximate propagation.

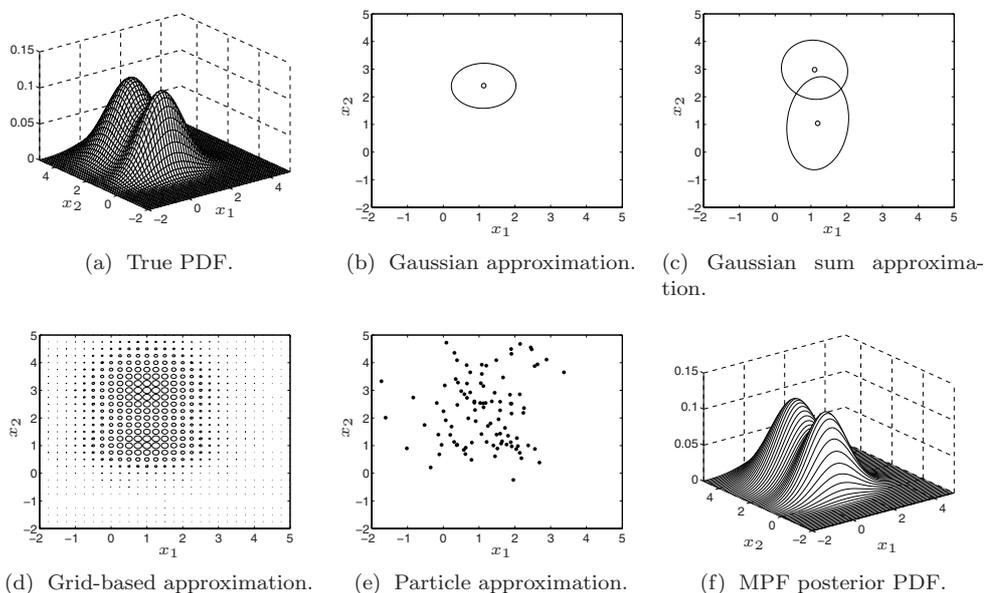


FIGURE 1. True probability density function and different approximate representations, in order of appearance, Gaussian, Gaussian sum, point-masses (grid-based approximation), particle samples and MPF PDF representation.

Figure 1(d)–(f) illustrates numerical approaches where the exact nonlinear relations present in the model are used for propagation. The *point-mass filter* (grid-based approximation) [4] employ a regular grid, where the grid weight is proportional to the posterior. The *particle filter* (PF), [17] represents the posterior by a stochastic grid in form of a set of samples, where all particles (samples) have the same weight. Finally, the *marginalized particle filter* (MPF) uses a stochastic grid for some of the states, and Gaussian distributions for the rest. That is, the MPF can be interpreted as a particle representation for a subspace of the state space, where each particle has an associated Gaussian distribution for the remaining state dimensions, Figure 1(f). It will be demonstrated that an exact nonlinear propagation is still possible if there is a linear sub-structure in the model. An important model class has the property that the (co-)variance is the same for all particles, which simplifies computations significantly.

2.2. Model

Consider a state vector x_t , which can be partitioned according to

$$x_t = \begin{pmatrix} x_t^n \\ x_t^l \end{pmatrix}, \quad (2)$$

where x_t^l denotes the linear states and x_t^n denotes the nonlinear states, in the dynamics and measurement relation. A rather general model with the properties discussed above is given by

$$x_{t+1}^n = f_t^n(x_t^n, k_t) + A_t^n(x_t^n, k_t)x_t^l + G_t^n(x_t^n, k_t)w_t^n, \quad (3a)$$

$$x_{t+1}^l = f_t^l(x_t^n, k_t) + A_t^l(x_t^n, k_t)x_t^l + G_t^l(x_t^n, k_t)w_t^l, \quad (3b)$$

$$y_t = h_t(x_t^n, k_t) + C_t(x_t^n, k_t)x_t^l + e_t, \quad (3c)$$

where k_t is used to denote a discrete mode parameter. Furthermore, the state noise is assumed white and Gaussian distributed with

$$w_t = \begin{pmatrix} w_t^n \\ w_t^l \end{pmatrix} \sim \mathcal{N}(0, Q_t), \quad Q_t = \begin{pmatrix} Q_t^n & Q_t^{ln} \\ (Q_t^{ln})^T & Q_t^l \end{pmatrix}. \quad (3d)$$

The measurement noise is assumed white and Gaussian distributed according to

$$e_t \sim \mathcal{N}(0, R_t). \quad (3e)$$

Furthermore, x_0^l is Gaussian,

$$x_0^l \sim \mathcal{N}(\bar{x}_0, \bar{P}_0). \quad (3f)$$

Finally, the density of x_0^n can be arbitrary, but it is assumed known. More specifically, conditioned on the nonlinear state variables and the discrete mode parameters there is a linear sub-structure, subject to Gaussian noise available in (3), given by (3b).

Bayesian estimation methods, such as the particle filter, provide estimates of the filtering density function $p(x_t, k_t | Y_t)$. By employing the fact

$$p(x_t^l, X_t^n, K_t | Y_t) = p(x_t^l | X_t^n, K_t, Y_t) p(X_t^n, K_t | Y_t) = p(x_t^l | X_t^n, K_t, Y_t) p(X_t^n | K_t, Y_t) p(K_t | Y_t), \quad (4)$$

the overall problem is decomposed into three sub-problems. Hence, a marginalized nonlinear filter for the general problem is characterised by

- A Kalman filter operating on the conditionally linear, Gaussian model (3) provides an estimate of $p(x_t^l | X_t^n, K_t, Y_t)$. Note that, conditioned on that the nonlinear state sequence X_t^n and discrete mode sequence K_t model (3) is a linear, Gaussian model.
- A marginalized nonlinear filter (e.g., PF, PMF, Gaussian sum filters, UKF) is designed for a fixed mode sequence.
- A pruning or merging scheme (IMM, AFMM) for the exponentially increasing number of mode sequences, see Chapter 10 in [18].

It is very important to note that the three sub-problems mentioned above are all coupled, for example, the result from the nonlinear filter at time t is used by the Kalman filters at time t . This is further explained in the subsequent section.

3. MARGINALIZED PARTICLE FILTER

In the MPF the marginalized nonlinear filter is given by a particle filter. The present discussion assumes that there are no discrete modes k_t present. This is just to make the presentation more accessible.

3.1. Algorithm

Similarly to (4) the filtering density function $p(x_t | Y_t)$ is given by (assuming no discrete modes)

$$p(x_t^l, X_t^n | Y_t) = p(x_t^l | X_t^n, Y_t) p(X_t^n | Y_t). \quad (5)$$

Using this expression the problem can be put in a form that is suitable for the MPF framework, i.e., to analytically marginalize out the linear state variables from $p(x_t | Y_t)$. Note that $p(x_t^l | X_t^n, Y_t)$ is analytically tractable, since X_t^n is given by the particle filter. Hence, the underlying model is conditionally linear-Gaussian, and the PDF can be computed from the Kalman filter. Furthermore, an estimate of $p(x_t^n | Y_t)$ is provided by the particle filter, more specifically, it is given by

$$p(x_t^n | Y_t) = \sum_{i=1}^N \tilde{\omega}_t \delta(x_t^n - x_t^{n,(i)}). \quad (6)$$

Hence, the resulting PDF estimate is given by

$$p(x_t | Y_t) = \sum_{i=1}^N \tilde{\omega}_t \delta(x_t^n - x_t^{n,(i)}) \mathcal{N}(x_t^l; \hat{x}_{t|t}^{l,(i)}, P_{t|t}^{(i)}), \quad (7)$$

motivating the fact that the MPF provides an estimate of the PDF that is a mix of a parametric and an unparametric estimate. That is a mix of a parametric distribution from the Gaussian family and a nonparametric distribution represented by samples. Another name for the MPF is the Rao-Blackwellized particle filter, and it has been known for quite some time, see e.g., [2, 5, 6, 10, 12, 12, 29, 33]. If the same numbers of particles are used in the standard PF and the MPF, the latter will provide estimates of better or at least the same quality. Intuitively this makes sense, since the dimension of $p(x_t^n | Y_t)$ is smaller than the dimension of $p(x_t | Y_t)$, implying that the particles occupy a lower dimensional space. Furthermore, the optimal algorithm is used to estimate the linear state variables. For a detailed discussion regarding the improved accuracy of the estimates, see, e.g., [11, 12].

The MPF for estimating the states in a dynamic model in the form (3) is provided in Algorithm 1. The present paper will merely provide the intuition for the MPF. For a detailed derivation, see [33]. From this algorithm, it should be clear that the only difference from the standard PF is that the time update (prediction) stage has been changed. In the standard PF, the prediction stage is given solely by step 4(b) in Algorithm 1.

Algorithm 1 Marginalized particle filter

- (1) Initialization: For $i = 1, \dots, N$, initialize the particles, $x_{0|-1}^{n,(i)} \sim p_{x_0^n}(x_0^n)$ and set $\{x_{0|-1}^{l,(i)}, P_{0|-1}^{(i)}\} = \{\bar{x}_0^l, \bar{P}_0\}$. Set $t := 0$.
- (2) PF measurement update: For $i = 1, \dots, N$, evaluate the importance weights

$$\omega_t^{(i)} = p\left(y_t | X_t^{n,(i)}, Y_{t-1}\right), \quad (8)$$

and normalize $\tilde{\omega}_t^{(i)} = \omega_t^{(i)} / \sum_{j=1}^N \omega_t^{(j)}$.

- (3) Resample N particles, with replacement,

$$Pr\left(x_{t|t}^{n,(i)} = x_{t|t-1}^{n,(j)}\right) = \tilde{\omega}_t^{(j)}.$$

- (4) PF time update and KF:

- (a) Kalman filter measurement update:

$$\hat{x}_{t|t}^l = \hat{x}_{t|t-1}^l + K_t \left(y_t - h_t - C_t \hat{x}_{t|t-1}^l \right), \quad (9a)$$

$$P_{t|t} = P_{t|t-1} - K_t M_t K_t^T, \quad (9b)$$

$$M_t = C_t P_{t|t-1} C_t^T + R_t, \quad (9c)$$

$$K_t = P_{t|t-1} C_t^T M_t^{-1}. \quad (9d)$$

- (b) PF time update (prediction): For $i = 1, \dots, N$, predict new particles,

$$x_{t+1|t}^{n,(i)} \sim p\left(x_{t+1|t}^n | X_t^{n,(i)}, Y_t\right).$$

- (c) Kalman filter time update:

$$\begin{aligned} \hat{x}_{t+1|t}^l &= \bar{A}_t^l \hat{x}_{t|t}^l + G_t^l (Q_t^{ln})^T (G_t^n Q_t^n)^{-1} z_t + f_t^l \\ &\quad + L_t \left(z_t - A_t^n \hat{x}_{t|t}^l \right), \end{aligned} \quad (10a)$$

$$P_{t+1|t} = \bar{A}_t^l P_{t|t} (\bar{A}_t^l)^T + G_t^l \bar{Q}_t^l (G_t^l)^T - L_t N_t L_t^T, \quad (10b)$$

$$N_t = A_t^n P_{t|t} (A_t^n)^T + G_t^n Q_t^n (G_t^n)^T, \quad (10c)$$

$$L_t = \bar{A}_t^l P_{t|t} (A_t^n)^T N_t^{-1}, \quad (10d)$$

where

$$z_t = x_{t+1}^n - f_t^n, \quad (11a)$$

$$\bar{A}_t^l = A_t^l - G_t^l (Q_t^{ln})^T (G_t^n Q_t^n)^{-1} A_t^n, \quad (11b)$$

$$\bar{Q}_t^l = Q_t^l - (Q_t^{ln})^T (Q_t^n)^{-1} Q_t^{ln}. \quad (11c)$$

- (5) Set $t := t + 1$ and iterate from step 2.

Let us now briefly discuss step 4 in Algorithm 1. Step 4(a) is a standard Kalman filter measurement update using the information available in the measurement y_t . Once this has been performed the new estimates of the linear states can be used to obtain a prediction of the nonlinear state $x_{t+1|t}^n$. This is performed in Step 4(b). Now, consider model (3) conditioned on the nonlinear state variable. The conditioning implies that (3a) can be thought of as a measurement equation. This is used in step 4(c) together with a time update of the linear state estimates.

The estimates, as expected means, of the state variables and their covariances are given below.

$$\hat{x}_{t|t}^n = \sum_{i=1}^N \tilde{\omega}_t^{(i)} \hat{x}_{t|t}^{n,(i)}, \quad (12a)$$

$$\hat{P}_{t|t}^n = \sum_{i=1}^N \tilde{\omega}_t^{(i)} \left(\left(\hat{x}_{t|t}^{n,(i)} - \hat{x}_{t|t}^n \right) \left(\hat{x}_{t|t}^{n,(i)} - \hat{x}_{t|t}^n \right)^T \right), \quad (12b)$$

$$\hat{x}_{t|t}^l = \sum_{i=1}^N \tilde{\omega}_t^{(i)} \hat{x}_{t|t}^{l,(i)}, \quad (12c)$$

$$\hat{P}_{t|t}^l = \sum_{i=1}^N \tilde{\omega}_t^{(i)} \left(P_{t|t}^{(i)} + \left(\hat{x}_{t|t}^{l,(i)} - \hat{x}_{t|t}^l \right) \left(\hat{x}_{t|t}^{l,(i)} - \hat{x}_{t|t}^l \right)^T \right), \quad (12d)$$

where $\{\tilde{\omega}_t^{(i)}\}_{i=1}^N$ are the normalized importance weights, provided by step 2 in Algorithm 1.

3.2. Analysis

In this section, several properties important in the practical application of the MPF are analyzed. First, the variance reduction inherent using the Rao-Blackwellization idea is explained. Second, the computational burden of the MPF is analyzed in detail. Note that this analysis is not generally applicable, it has to be performed on a case by case basis. However, the approach briefly outlined in Section 3.2.2 below can always be used.

3.2.1. Variance Reduction

The variance of a function or estimator $g(U, V)$, depending on two random variables, U and V can be written as

$$\text{Var}\{g(U, V)\} = \text{Var}\{\text{E}\{g(U, V)|V\}\} + \text{E}\{\text{Var}\{g(U, V)|V\}\}. \quad (13)$$

Hence, in principle, the conditional inequality

$$\text{Var}\{\text{E}\{g(x_t^l, X_t^n)|X_t^n\}\} \leq \text{Var}\{g(x_t^l, X_t^n)\}, \quad (14)$$

can be employed. In the current MPF setup, U and V are represented by the linear and nonlinear states, respectively. This is sometimes referred to as Rao-Blackwellization, see, e.g., [31] and it is the basic part that improves performance using the marginalization idea. Note that for the variance reduction to be significant, the left hand side in (14) has to be significantly smaller than the right hand side. In other words, the term

$$\text{E}\{\text{Var}\{g(x_t^l, X_t^n)|X_t^n\}\} \quad (15)$$

has to be large. That is, the expectation of the conditional variance of the corresponding Kalman filter has to be large. In order to make this a bit clearer, let $g(x_t^l, X_t^n) = x_t^l$, implying that (15) reads

$$\text{Var}\{\text{E}\{x_t^l|X_t^n\}\} \leq \text{Var}\{x_t^l\} \quad (16)$$

This shows that the variance of the linear part is always smaller for the MPF than for the PF. The difference is the expected variance term,

$$\text{E}\{\text{Var}\{g(U, V)|V\}\} = \text{E}\{\text{Var}\{x_t^l|X_t^n\}\} = \text{E}\{P_{t|t}^i\}. \quad (17)$$

This states that the improvement in the quality of the estimate is given by the term $E\{P_{t|t}^i\}$. That is, the Kalman filter covariance $P_{t|t}$ is a good indicator of how much that has been gained in using the MPF instead of the PF.

3.2.2. Computational Complexity

In discussing the use of the MPF it is sometimes better to partition the state vector into one part that is estimated using the PF $x_t^p \in \mathbb{R}^p$ and one part that is estimated using the KF $x_t^k \in \mathbb{R}^k$. Obviously all the nonlinear states x_t^n are included in x_t^p . However, we could also choose to include some of the linear states there as well. Under the assumption of linear dynamics, this notation allows us to write (3) according to

$$x_{t+1}^p = A_t^p x_t^p + A_t^k x_t^k + w_t^p, \quad w_t^p \sim \mathcal{N}(0, Q_t^p), \quad (18a)$$

$$x_{t+1}^k = F_t^p x_t^p + F_t^k x_t^k + w_t^k, \quad w_t^k \sim \mathcal{N}(0, Q_t^k), \quad (18b)$$

$$y_t = h_t(x_t^p) + C_t x_t^k + e_t, \quad e_t \sim \mathcal{N}(0, R_t). \quad (18c)$$

First, the case $C_t = 0$ is discussed. For instance, the first instruction $P_{t|t}(A_t^k)^T$ corresponds to multiplying $P_{t|t} \in \mathbb{R}^{k \times k}$ with $(A_t^k)^T \in \mathbb{R}^{k \times p}$, which requires pk^2 multiplications and $(k-1)kp$ additions [16]. The total *equivalent flop* (EF)¹ complexity is derived by [24],

$$\mathcal{C}(p, k, N) \approx 4pk^2 + 8kp^2 + \frac{4}{3}p^3 + 5k^3 - 5kp + 2p^2 + (6kp + 4p^2 + 2k^2 + p - k + pc_3 + c_1 + c_2)N. \quad (19)$$

Here, the coefficient c_1 has been used for the calculation of the Gaussian likelihood, c_2 for the resampling and c_3 for the random number complexity. Note that, when $C_t = 0$ the same covariance matrix is used for all Kalman filters, which significantly reduce the computational complexity.

By requiring $\mathcal{C}(p+k, 0, N_{\text{PF}}) = \mathcal{C}(p, k, N(k))$, where N_{PF} corresponds to the number of particles used in the standard PF we can solve for $N(k)$. This gives the number of particles $N(k)$ that can be used by the MPF in order to obtain the same computational complexity as if the standard PF had been used for all states. In Figure 2 the ratio $N(k)/N_{\text{PF}}$ is plotted for systems with $m = 3, \dots, 9$ states.

Hence, using Figure 2 it is possible to directly find out how much there is to gain in using the MPF from a computational complexity point of view. The figure also shows that the computational complexity is always reduced when the MPF can be used instead of the standard PF. Furthermore, as previously mentioned, the quality of the estimates will improve or remain the same when the MPF is used [12].

Second, if $C_t \neq 0$, the Riccati recursions have to be evaluated separately for each particle. This results in a significantly increased computational complexity. Hence, different covariance matrices have to be used for each Kalman filter, implying that (19) has to be modified. Approximately the complexity is given by [24],

$$\mathcal{C}(p, k, N) \approx (6kp + 4p^2 + 2k^2 + p - k + pc_3 + c_1 + c_2 + 4pk^2 + 8kp^2 + \frac{4}{3}p^3 + 5k^3 - 5kp + 2p^2 + k^3)N. \quad (20)$$

In Figure 3 the ratio $N(k)/N_{\text{PF}}$ is plotted for systems with $m = 3, \dots, 9$ states. For systems with few states the MPF is more efficient than the standard PF. However, for systems with more states, where most of the states are marginalized the standard PF becomes more efficient than the MPF. This is due to the Riccati recursions mentioned above.

¹The EF complexity for an operation is defined as the number of flops that result in the same computational time as the operation.

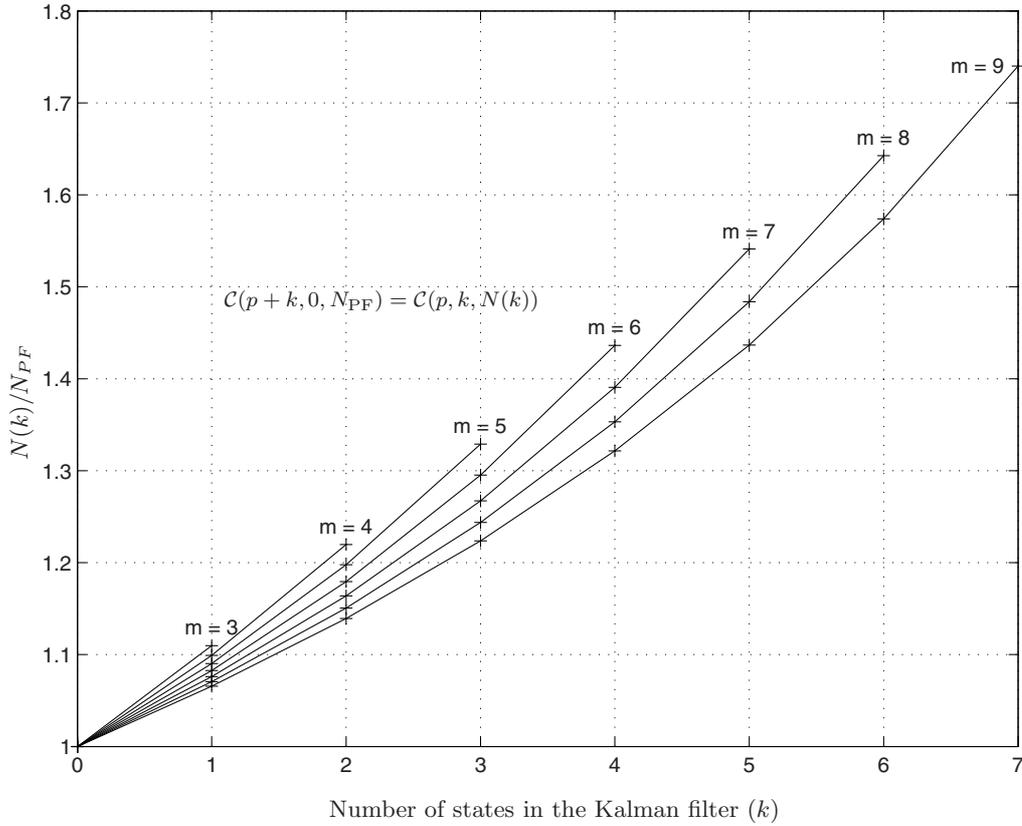


FIGURE 2. Ratio $N(k)/N_{PF}$ for systems with $m = 3, \dots, 9$ states and $C_t = 0$, $n = 2$ is shown. It is apparent the MPF can use more particles for a given computational complexity, when compared to the standard PF.

4. ILLUSTRATING EXAMPLE

The aim here is to illustrate how the MPF works using the following nonlinear stochastic system.

$$x_{t+1}^n = x_t^l x_t^n + w_t^n, \quad (21a)$$

$$x_{t+1}^l = x_t^l + w_t^l, \quad (21b)$$

$$y_t = 0.2(x_t^n)^2 + e_t, \quad (21c)$$

where the noise is assumed white and Gaussian according to

$$w_t = \begin{pmatrix} w_t^n \\ w_t^l \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0.25 & 0 \\ 0 & 10^{-4} \end{pmatrix} \right), \quad (21d)$$

$$e_t \sim \mathcal{N}(0, 1). \quad (21e)$$

The initial state x_0 is given by

$$x_0 \sim \mathcal{N} \left(\begin{pmatrix} 0.1 \\ 0.99 \end{pmatrix}, \begin{pmatrix} 16 & 0 \\ 0 & 10^{-3} \end{pmatrix} \right) \quad (21f)$$

This system was used in [38], where it illustrated grid-based (point-mass) filters. Obviously, the states can be estimated by applying the standard particle filter to the entire state vector. However, a better solution is to exploit the conditionally linear, Gaussian sub-structure that is present in (21). The nonlinear process x_t^n is a first-order AR process, where the linear process x_t^l is the time-varying parameter. The linear, Gaussian sub-structure is used by the MPF and the resulting filtering density function at time 10, $p(x_{10}|Y_{10})$ is given in Figure 4 (for a particular realization). In this example 2000 particles were used, but only 100 of them are plotted in Figure 4 in order to obtain a clearer illustration of the result. The figure illustrates the fact that the MPF is a combination of the KF and the PF. The density functions for the linear states are provided by the Kalman filters, which is evident from the fact that the marginal densities $p(x_t^{l,(i)}|Y_t)$ are given by Gaussian densities. Furthermore, the nonlinear state estimates are provided by the PF. Hence, the linear states are given by a parametric estimator (KF), whereas the nonlinear states are given by a nonparametric estimator (PF). In this context the MPF can be viewed as a combination of a parametric and an unparametric estimator.

5. APPLICATION OVERVIEW

Two important areas where the MPF has been successfully applied are GPS-free *positioning*, where the aim is to estimate the own platform's position and *target tracking*, where the state of an unknown, observed target

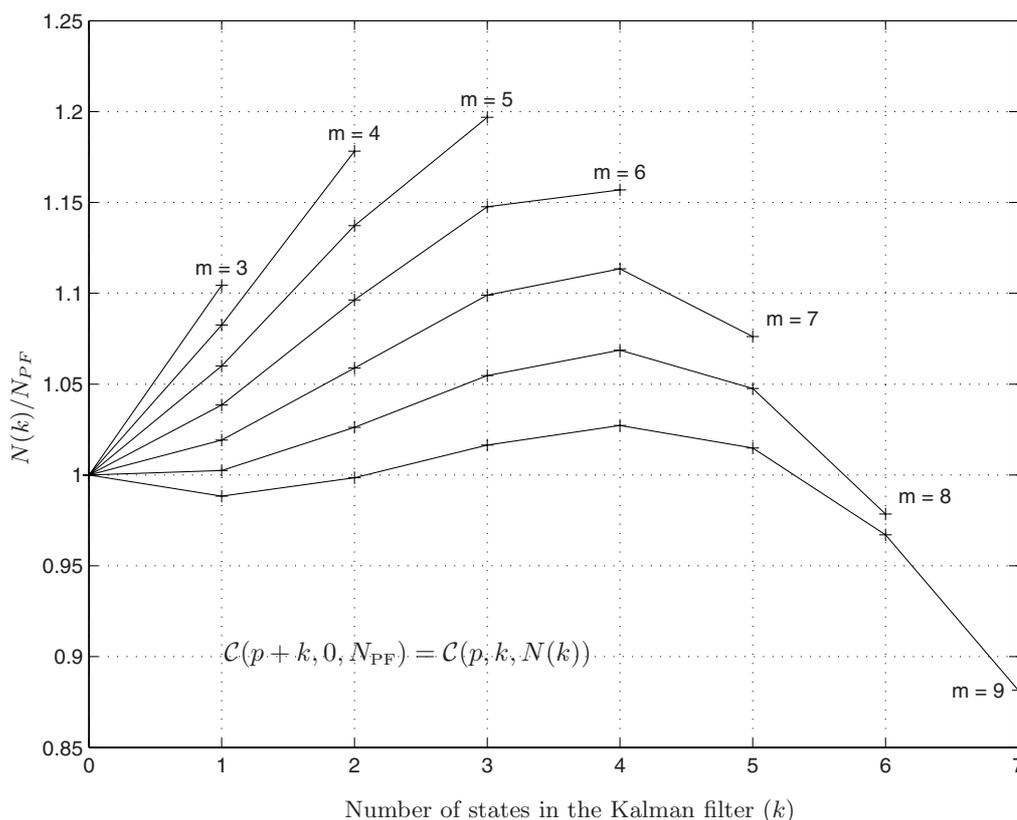


FIGURE 3. Ratio $N(k)/N_{PF}$ for systems with $m = 3, \dots, 9$ states and $C_t \neq 0$, $n = 2$ is shown. For systems with high state dimension and many marginalized states the standard PF can use more particles than the MPF.

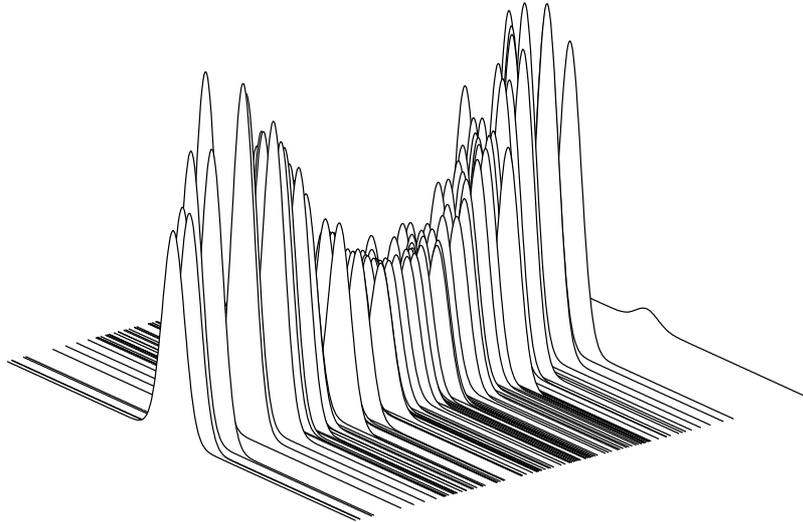


FIGURE 4. The estimated filter PDF for system (21) at time 10, $p(x_{10}|Y_{10})$ using the MPF. It is instructive to see that the linear state x_{10}^l is estimated by Gaussian densities (from the Kalman filter) and position along the nonlinear state x_{10}^n is given by a particles (from the particle filter). Note that only a subset of particles have been plotted, in order to make the illustration clear.

is estimated from measurements. These applications also represent typical examples where sensor fusion techniques are important. The MPF provides an efficient way to incorporate both linear and nonlinear measurement relations. A more detailed overview of some of our MPF applications is provided [34]. In *simultaneous localization and mapping* (SLAM) applications, where the map information is built up from measurement data, the MPF is one of the key techniques for fast real-time applications, see for instance [37]. In SLAM applications the resulting algorithm is sometimes referred to as FastSLAM.

Here is a list of some MPF applications:

Positioning and map applications:

- *Underwater terrain-aided positioning* [21, 23]
- *Aircraft terrain-aided positioning* [33]
- *Automotive map-aided positioning* [36]
- *GPS navigation* [15]
- *Simultaneous Localization and Mapping (SLAM)* [13, 27, 28, 37]

Target tracking applications:

- *Automotive target tracking* [14]
- *Bearings-only target tracking* [22]

- *Radar target tracking* [34]

Other applications:

- *Communication applications* [7, 39]
- *System identification* [8, 25, 26, 32]
- *Audio applications* [3].

ACKNOWLEDGMENT

This work was supported by VINNOVA's Center of Excellence ISIS (Information Systems for Industrial Control and Supervision), and by the Swedish Research Council (VR).

REFERENCES

- [1] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Information and system science series. Prentice Hall, Englewood Cliffs, NJ, USA, 1979.
- [2] C. Andrieu and A. Doucet. Particle filtering for partially observed Gaussian state space models. *Journal of the Royal Statistical Society*, 64(4):827–836, 2002.
- [3] C. Andrieu and S. J. Godsill. A particle filter for model based audio source separation. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation (ICA 2000)*, Helsinki, Finland, June 2000.
- [4] N. Bergman. *Recursive Bayesian Estimation: Navigation and Tracking Applications*. Phd thesis No 579, Linköping Studies in Science and Technology, SE-581 83 Linköping, Sweden, May 1999.
- [5] G. Casella and C. P. Robert. Rao-Blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.
- [6] R. Chen and J. S. Liu. Mixture Kalman filters. *Journal of the Royal Statistical Society*, 62(3):493–508, 2000.
- [7] R. Chen, X. Wang, and J. S. Liu. Adaptive joint detection in flat-fading channels via mixture Kalman filtering. *IEEE Transactions on Information Theory*, 46(6):2079–2094, 2000.
- [8] M. J. Daly, J. P. Reilly, and M. R. Morelande. Rao-Blackwellised particle filtering for blind system identification. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, PA, USA, Mar. 2005.
- [9] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York, USA, 2001.
- [10] A. Doucet, S. J. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.
- [11] A. Doucet, N. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. Technical Report CUED/F-INFENG/TR 359, Signal Processing Group, Department of Engineering, University of Cambridge, Trumpington street, CB2 1PZ Cambridge, 1999.
- [12] A. Doucet, N. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624, 2001.
- [13] E. Eade and T. Drummond. Scalable monocular SLAM. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 469–476, New York, NY, USA, June 2006.
- [14] A. Eidehall, T. B. Schön, and F. Gustafsson. The marginalized particle filter for automotive tracking applications. In *Proceedings of the IEEE Intelligent Vehicle Symposium*, pages 369–374, Las Vegas, USA, June 2005.
- [15] A. Giremus and J. Y. Tourneret. Joint detection/estimation of multipath effects for the global positioning system. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 17–20, Philadelphia, PA, USA, Mar. 2005.
- [16] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins University Press, Baltimore, third edition, 1996.
- [17] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEEE Proceedings on Radar and Signal Processing*, volume 140, pages 107–113, 1993.
- [18] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons, New York, USA, 2000.
- [19] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Information and System Sciences Series. Prentice Hall, Upper Saddle River, NJ, USA, 2000.
- [20] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME, Journal of Basic Engineering*, 82:35–45, 1960.
- [21] R. Karlsson and F. Gustafsson. Particle filter for underwater navigation. In *Proceedings of the Statistical Signal Processing Workshop*, pages 509–512, St. Louis, MO, USA, Sept. 2003.
- [22] R. Karlsson and F. Gustafsson. Recursive Bayesian estimation – bearings-only applications. *IEE Proceedings on Radar, Sonar, and Navigation. Special issue on target tracking: Algorithms and Applications*, 152(5):305–313, Oct. 2005.

- [23] R. Karlsson and F. Gustafsson. Bayesian surface and underwater navigation. *IEEE Transactions on Signal Processing*, 2006. Accepted for publication.
- [24] R. Karlsson, T. Schön, and F. Gustafsson. Complexity analysis of the marginalized particle filter. *IEEE Transactions on Signal Processing*, 53(11):4408–4411, Nov. 2005.
- [25] P. Li, R. Goodall, and V. Kadiramanathan. Parameter estimation of railway vehicle dynamic model using Rao-Blackwellised particle filter. In *Proceedings of the European Control Conference*, Cambridge, UK, Sept. 2003.
- [26] P. Li, R. Goodall, and V. Kadiramanathan. Estimation of parameters in a linear state space model using Rao-Blackwellised particle filter. *IEE Proceedings - Control Theory and Applications*, 151(6):727–738, Nov. 2004.
- [27] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM a factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002.
- [28] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003.
- [29] P.-J. Nordlund. *Sequential Monte Carlo Filters and Integrated Navigation*. Licentiate Thesis No 945, Department of Electrical Engineering, Linköping University, Sweden, 2002.
- [30] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: particle filters for tracking applications*. Artech House, London, UK, 2004.
- [31] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer texts in statistics. Springer, New York, 1999.
- [32] T. Schön and F. Gustafsson. Particle filters for system identification of state-space models linear in either parameters or states. In *Proceedings of the 13th IFAC Symposium on System Identification*, pages 1287–1292, Rotterdam, The Netherlands, Sept. 2003.
- [33] T. Schön, F. Gustafsson, and P.-J. Nordlund. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279–2289, July 2005.
- [34] T. B. Schön, R. Karlsson, and F. Gustafsson. The marginalized particle filter in practice. In *Proceedings of IEEE Aerospace Conference*, Big Sky, MT, USA, Mar. 2006.
- [35] H. W. Sorenson and D. L. Alspach. Recursive Bayesian estimation using Gaussian sum. *Automatica*, 7:465–479, 1971.
- [36] N. Svenzén. Real time implementation of map aided positioning using a Bayesian approach. Master's Thesis No LiTH-ISY-EX-3297-2002, Department of Electrical Engineering, Linköping University, Sweden, Dec. 2002.
- [37] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents. The MIT Press, Cambridge, MA, USA, 2005.
- [38] M. Šimandl, J. Královec, and Söderström. Advanced point-mass method for nonlinear state estimation. *Automatica*, 42(7):1133–1145, July 2006.
- [39] X. Wang, R. R. Chen, and D. Guo. Delayed-pilot sampling for mixture Kalman filter with application in fading channels. *IEEE Transactions on Signal Processing*, 50(2):241–254, Feb. 2002.