2015 European Control Conference (ECC)
July 15-17, 2015. Linz, Austria

Monte Carlo Filter Particle Filter

Masaya Murata†, Hidehisa Nagano† and Kunio Kashino†‡

Abstract— **We propose a new realization method of the sequential importance sampling (SIS) algorithm to derive a new particle filter. The new filter constructs the importance distribution by the Monte Carlo filter (MCF) using sub-particles, therefore, its non-Gaussianity nature can be adequately considered while the other type of particle filter such as unscented Kalman filter particle filter (UKF-PF) assumes a Gaussianity on the importance distribution. Since the state estimation accuracy of the SIS algorithm theoretically improves as the estimated importance distribution becomes closer to the true posterior probability density function of state, the new filter is expected to outperform the existing, state-of-the-art particle filters. We call the new filter Monte Carlo filter particle filter (MCF-PF) and confirm its effectiveness through the numerical simulations.**

I. Introduction

A. Problem Definition

In this paper, we consider the sequential state estimation problem of the following nonlinear non-Gaussian state space model.

$$X_t = f(X_{t-1}) + \Phi_t,\ \Phi_t \sim p(\Phi_t) \quad (1)$$
$$Y_t = h(X_t) + \Omega_t,\ \Omega_t \sim p(\Omega_t) \quad (2)$$

Here, X_t is the L-dimensional random vector called state at time t and this is the estimation target using the past series of M-dimensional observations $\{\mathcal{Y}_{t-1}\} = \{Y_{t-1}, Y_{t-2}, \cdots, Y_1\}$ or $\{\mathcal{Y}_t\} = \{Y_t, Y_{t-1}, \cdots, Y_1\}$. The former is called the predicted state estimate $\bar{X}_t := E[X_t|\{\mathcal{Y}_{t-1}\}]$ and the latter is called the filtered state estimate $\hat{X}_t := E[X_t|\{\mathcal{Y}_t\}]$. Their approximation errors are calculated by $E[(X_t - \bar{X}_t)(X_t - \bar{X}_t)^T|\{\mathcal{Y}_{t-1}\}]$ and $E[(X_t-\hat{X}_t)(X_t-\hat{X}_t)^T|\{\mathcal{Y}_t\}]$, respectively. $f(\cdot)$ and $h(\cdot)$ are nonlinear functions and called state transition and state observation, respectively. Φ_t and Ω_t are independent white noises following arbitrary probability density functions (PDFs) and they are called system noise and observation noise, respectively. Equations (1) and (2) are referred to as system model and observation model, and these two equations constitute the state space model of interest. For this model, since both state and noises do not always follow Gaussian distributions, the well-studied, state-of-the-art Gaussian filters[1] such as extended Kalman filter (EKF)[2] and unscented Kalman filter (UKF)[3] can not be adequately employed. Therefore, designing the sequential state estimator for this model still remains as a challenging problem.

†Masaya Murata, Hidehisa Nagano and Kunio Kashino are research scientists at NTT Communication Science Laboratories, NTT Corporation, 3-1, Morinosato Wakamiya, Atsugi-Shi, Kanagawa 243-0198, Japan. murata.masaya at lab.ntt.co.jp

‡Kunio Kashino is also a visiting professor at National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430 Japan.

B. Sequential Importance Sampling (SIS)

To tackle the problem of interest, the sequential importance sampling (SIS) algorithm[4][5] is often used to obtain the filtered state estimate. The SIS algorithm approximates the $p(X_t|\{\mathcal{Y}_t\})$, that is, the posterior PDF of state at t given observations up to time t, as follows:

Sequential Importance Sampling (SIS)

$$p(X_t|\{\mathcal{Y}_t\}) \approx \sum_{i=1}^{N} w_t'^{(i)} \delta(X_t - X_t^{(i)})$$

where

$$X_t^{(i)} \sim p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t),\ i = 1, 2, \cdots, N$$

$$w_t^{(i)} = w_{t-1}'^{(i)} \frac{p(Y_t|X_t = X_t^{(i)}) p(X_t = X_t^{(i)}|X_{t-1} = X_{t-1}^{(i)})}{p(X_t = X_t^{(i)}|X_{t-1} = X_{t-1}^{(i)}, Y_t)}$$

$$w_t'^{(i)} = \frac{w_t^{(i)}}{\sum_{j=1}^{N} w_t^{(j)}}$$

Here, $\{X_t^{(i)}, w_t'^{(i)}\},\ i = 1, 2, \cdots, N$ are weighted set of particles (particle values and their corresponding normalized importance weights) at time t and $\sum_{i=1}^{N} w_t'^{(i)} = 1$ holds. $\delta(\cdot)$ is the Dirac's delta function and this PDF approximation is called the Dirac's point-mass estimate. $p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t)$ is the optimal importance distribution specified by eqs. (1) and (2) and it proves to minimizes the variance of the importance weights. $X_t^{(i)} \sim p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t),\ i = 1, 2, \cdots, N$ denotes the random (i.i.d.) sampling from the optimal importance distribution. $p(Y_t|X_t = X_t^{(i)})$ and $p(X_t = X_t^{(i)}|X_{t-1} = X_{t-1}^{(i)})$ are likelihood and transition PDFs specified by eq. (2) and (1), respectively. The above importance weight equation also indicates that the current weight is calculated from the previous weight and together with the fact that the current particle value depends on the previous value, this algorithm can be sequentially performed after the appropriate importance distribution selection. Then the filtered state estimate at time t (denoted as $\hat{X}_t$) can be obtained as follows:

$$\hat{X}_t = \sum_{i=1}^{N} w_t'^{(j)} X_t^{(i)} \quad (3)$$

Note that the SIS algorithm is not designed to obtain the predicted state estimate at time t (denoted as $\bar{X}_t$). This algorithm proves to easily encounter the degeneracy problem that almost all the particles have zero or nearly zero weights as the time proceeds and therefore, the particle resampling procedure is often performed after obtaining $\{X_t^{(i)}, w_t'^{(i)}\},\ i = 1, 2, \cdots, N$. The resampling makes the importance weights for all of the new particles become $1/N$. The new particles are randomly drawn from the discrete set $\{X_t^{(i)}\}, i = 1, 2, \cdots, N$ with probabilities $\{w_t'^{(i)}\}, i = 1, 2, \cdots, N$. Note that after the resampling procedure, the sequential weight equation becomes

$$w_t^{(i)} \propto \frac{p(Y_t|X_t = X_t^{(i)})p(X_t = X_t^{(i)}|X_{t-1} = X_{t-1}^{(i)})}{p(X_t = X_t^{(i)}|X_{t-1} = X_{t-1}^{(i)}, Y_t)} \tag{4}$$

since the resampling makes $\{w_t'^{(i)}\}, i = 1, 2, \cdots, N$ become $1/N$.

C. Importance Distribution Selection

The selection of the importance distribution affects the entire state estimation performance of the SIS algorithm and generally speaking, the smaller the difference between the assumed importance distribution and true posterior PDF of state, the better the state estimation accuracy becomes. The simplest choice is $p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t) \approx p(X_t|X_{t-1} = X_{t-1}^{(i)})$ and it derives the bootstrap filter (BF)[6]. Here, the $p(X_t|X_{t-1} = X_{t-1}^{(i)})$ is the state transition PDF specified by eq. (1) and the BF is one of the most popular particle filters when tackling the problem of interest. However, one obvious problem is that the possible large deviation between the assumed importance distribution and the true one is likely to arise. To alleviate this gap, we may need the large number of particles and it sometimes results in the severe computational burden.

The other often-used selection is to assume $p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t) \approx p(X_t|X_{(i),t-1}, Y_t)$ where $X_{(i),t-1}$ is supposed to follow eqs. (1) and (2) and $E[X_{(i),t-1}|\{Y_{t-1}\}]$ is set to $X_{t-1}^{(i)}$. We call $X_{(i),t-1}$ ith particle state. Moreover the $p(X_t|X_{(i),t-1}, Y_t)$ is assumed as Gaussian distributed. Then, using N particles which specify the expectations of the N particle states, N different $p(X_t|X_{(i),t-1}, Y_t)$ are estimated by using the Gaussian filter such as EKF or UKF. The former filter is called extended Kalman filter particle filter (EKF-PF)[7] and the latter one is called unscented Kalman filter particle filter (UKF-PF)[7][8]. Since the latest observation Y_t is incorporated when constructing the importance distribution, these filters theoretically are expected to be more accurate than the BF that does not take Y_t into account. The UKF-PF is also theoretically better than the EKF-PF since the unscented transformation (UT)[3] in the UKF algorithm handles the nonlinear state transformation with higher accuracy than the truncated Taylor series based linearization approach in the EKF algorithm.

D. Contribution

In this paper, we propose a method to construct the $p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t)$ by using Monte Carlo filter (MCF)[9] with sub-particles. Therefore the non-Gaussianity of the importance distribution is taken into account, contributing to the improvement in the state estimation accuracy. Although the computational burden increases due to the MCF execution for all of the particles under consideration, the new filter is expected to be superior over the existing filters. We call the new filter Monte Carlo filter particle filter (MCF-PF) and confirm its effectiveness using the numerical examples.

The rest of this paper is organized as follow. Section I I describes the MCF algorithm and its relationship to the BF algorithm. In section III, we explain the proposed filter formulation and summarize the sequential state estimation algorithm. Numerical examples are shown in section IV and section V summarizes the basic findings and conclusion of this paper.

II. Monte Carlo Filter (MCF)

A. Algorithm

As well as the SIS algorithm, the Monte Carlo filter (MCF) is also a promising solution for the aforementioned sequential state estimation problem. Unlike the SIS, the MCF approximates the Bayesian filter algorithm using randomly drawn samples called particles. And, the MCF not only provides the filtered state estimate, but also provides the predicted state estimate. The algorithm of MCF can be summarized as follows:

Monte Carlo Filter (MCF)

$\hat{X}_0^{(i)} \sim \mathcal{N}(\bar{X}_0, \bar{P}_0),\ \hat{w}_0^{(i)} = 1/N,\ i = 1, 2, \cdots, N$

For each $t = 1, 2, \cdots$

Predicted state estimate

$$\Phi_t^{(i)} \sim p(\Phi_t),\ i = 1, 2, \cdots, N$$
$$\bar{X}_t^{(i)} = f(\hat{X}_{t-1}^{(i)}) + \Phi_t^{(i)}$$
$$\bar{w}_t'^{(i)} = \hat{w}_{t-1}'^{(i)}$$
$$\bar{X}_t = \sum_{j=1}^{N} \bar{w}_t'^{(j)} \bar{X}_t^{(j)}$$

Filtered state estimate

$$\hat{X}_t^{(i)} = \bar{X}_t^{(i)}$$
$$\hat{w}_t^{(i)} = \bar{w}_t'^{(i)} p(Y_t|X_t = \hat{X}_t^{(i)})$$
$$\hat{w}_t'^{(i)} = \frac{\hat{w}_t^{(i)}}{\sum_{j=1}^{N} \hat{w}_t^{(j)}}$$
$$\hat{X}_t = \sum_{j=1}^{N} w_t'^{(j)} \hat{X}_t^{(j)}$$

Here, $\hat{X}_0^{(i)} \sim p(X_0)$ and $\Phi_t^{(i)} \sim p(\Phi_t)$ are the ith particles randomly drawn from the initial state PDF and the system noise PDF, respectively. $p(Y_t|X_t = \bar{X}_t^{(i)})$ is the likelihood PDF specified by eq.(2). $\bar{X}_t$ and $\hat{X}_t$ are the predicted and the filtered state estimates, and they are calculated by the weighted sum of the particle values as shown in the algorithm. As well as the SIS algorithm, the particle resampling procedure is performed after the filtered state estimate to randomly drawn particles from the estimated posterior state PDF. Then $\{w_t'^{(i)}\}, i = 1, 2, \cdots, N$ all become $1/N$ and the weight equation in the filtered state estimate can be simply expressed as follows:

$$\hat{w}_t^{(i)} \propto p(Y_t|X_t = \hat{X}_t^{(i)}) \tag{5}$$

Such particle resampling is also effective to avoid the particle impoverishment problem that almost all of the particles have the same values after the time proceeds to some extent.

B. Relationship to the Bootstrap Filter (BF)

The difference between the MCF and BF mentioned in the previous section is that while the MCF is based on the Bayesian filter, the BF is formulated from the SIS algorithm. The MCF uses $\hat{X}_t^{(i)} = \bar{X}_t^{(i)} = f(\hat{X}_{t-1}^{(i)}) + \Phi_t^{(i)}$ to approximate the posterior PDF of state, while the BF uses $X_t^{(i)} \sim p(X_t|X_{t-1} = X_{t-1}^{(i)})$. Then, since their weight equations are the same (eq. (4) in section I-B becomes eq. (5) in section II-A by selecting the transition prior as the importance distribution), the MCF is very similar to the BF. Therefore the same problem occurring in the BF algorithm also exists in the MCF algorithm such that the large number of particles are required to guarantee the state estimation accuracy due to the deviation between the simple importance distribution and the true posterior PDF of state. To address this issue, we propose the new particle filter formulation that employs the MCF algorithm to accurately estimate the importance distribution. The resulting estimated importance distribution is expected to be much closer to the true posterior PDF of state than that for the BF.

III. Monte Carlo Filter Particle Filter (MCF-PF)

A. Importance Distribution Estimation using Sub-particles

The new filter called Monte Carlo filter particle filter (MCF-PF) constructs the importance distribution in the SIS algorithm by using MCF algorithm. The importance distribution becomes as follows:

$$\begin{aligned} p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t) \\ &= \frac{p(Y_t|X_t)p(X_t|X_{t-1} = X_{t-1}^{(i)})}{\int p(Y_t|X_t)p(X_t|X_{t-1} = X_{t-1}^{(i)})dX_t} \\ &= \frac{1}{C_t} p(Y_t|X_t)p(X_t|X_{t-1} = X_{t-1}^{(i)}) \end{aligned} \tag{6}$$

Here, $C_t \equiv \int p(Y_t|X_t)p(X_t|X_{t-1} = X_{t-1}^{(i)})dX_t$. By random sampling from $p(X_t|X_{t-1} = X_{t-1}^{(i)})$ specified by eq. (1), the following Dirac's point-mass approximation holds:

$$p(X_t|X_{t-1} = X_{t-1}^{(i)}) \approx \frac{1}{M} \sum_{j=1}^{M} \delta(X_t - \bar{X}_t^{(i),(j)}) \tag{7}$$

Here, $\bar{X}_t^{(i),(j)} \sim p(X_t|X_{t-1} = X_{t-1}^{(i)}),\ j = 1, 2, \cdots, M$ and the sub-particle number M is not necessarily equal to N (particle number in the SIS algorithm). We can then obtain the following equations:

$$\begin{aligned} p(Y_t|X_t)p(X_t|X_{t-1} = X_{t-1}^{(i)}) \\ &\approx \frac{1}{M} p(Y_t|X_t) \sum_{j=1}^{M} \delta(X_t - \bar{X}_t^{(i),(j)}) \end{aligned} \tag{8}$$

$$C_t = \frac{1}{M} \sum_{k=1}^{M} p(Y_t|X_t = \bar{X}_t^{(i),(k)}) \tag{9}$$

Substituting eqs. (8) and (9) into eq. (6) yields,

$$\begin{aligned} p(X_t|X_{t-1} = X_{t-1}^{(i)}, Y_t) \\ &\approx \frac{p(Y_t|X_t)}{\sum_{k=1}^{M} p(Y_t|X_t = \bar{X}_t^{(i),(k)})} \sum_{j=1}^{M} \delta(X_t - \bar{X}_t^{(i),(j)}) \\ &= \sum_{j=1}^{M} \hat{w}_t^{\prime(i),(j)} \delta(X_t - \hat{X}_t^{(i),(j)}) \end{aligned} \tag{10}$$

$$\begin{aligned} &\text{where } \hat{X}_t^{(i),(j)} = \bar{X}_t^{(i),(j)} \\ &\hat{w}_t^{(i),(j)} = p(Y_t|X_t = \bar{X}_t^{(i),(j)}) \\ &\hat{w}_t^{\prime(i),(j)} = \frac{\hat{w}_t^{(i),(j)}}{\sum_{k=1}^{M} \hat{w}_t^{(i),(k)}} \end{aligned}$$

Equation (10) means that the importance distribution for the ith particle is approximated by the MCF algorithm using the M sub-particles. Therefore, random sampling from the estimated importance distribution can be replaced with the uniform sampling from the discrete set $\{\hat{X}_t^{(i),(j)}\},\ i = 1, 2, \cdots, M$ with probabilities $\{\hat{w}_t^{\prime(i),(j)}\},\ i = 1, 2, \cdots, M$. Such sampling corresponds to the particle resampling procedure mentioned in the previous section. In short, we construct the importance distribution for each particle in conjunction with the MCF algorithm using the sub-particles and obtain the new particle by performing the resampling procedure to the constructed importance distribution.

B. Importance Weight Calculation

As explained in the previous section, by using eq. (10), the new particle sampled from the estimated importance distribution can be expressed as follows:

$$\begin{aligned} \hat{X}_t^{(i)} &\sim p(X_t|X_{t-1} = X_{t-1}^{(i)}|Y_t) \\ &\sim \sum_{j=1}^{M} \hat{w}_t^{\prime(i),(j)} \delta(X_t - \hat{X}_t^{(i),(j)}) \end{aligned}$$

Since the $\hat{X}_t^{(i)}$ is either $\hat{X}_t^{(i),(1)}$, $\hat{X}_t^{(i),(2)}$, $\cdots$, or $\hat{X}_t^{(i),(M)}$, we can also denote $\hat{X}_t^{(i)}$ as $\hat{X}_t^{(i),(l)}$, where l is either 1, 2, $\cdots$, or M. Therefore, the sequential importance weight calculation and the normalized weight become

$$\begin{aligned} \hat{w}_t^{(i)} &= \hat{w}_{t-1}^{\prime(i)} \frac{p(Y_t|X_t = \hat{X}_t^{(i)})p(X_t = \hat{X}_t^{(i)}|X_{t-1} = X_{t-1}^{(i)})}{p(X_t = \hat{X}_t^{(i)}|X_{t-1} = X_{t-1}^{(i)}, Y_t)} \\ &= \hat{w}_{t-1}^{\prime(i)} \times \\ &\quad \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})p(X_t = \hat{X}_t^{(i),(l)}|X_{t-1} = X_{t-1}^{(i)})}{\hat{w}_t^{\prime(i),(l)} \delta(0)} \end{aligned} \tag{11}$$

$$\hat{w}_t^{\prime(i)} = \frac{\hat{w}_t^{(i)}}{\sum_{k=1}^{N} \hat{w}_t^{(k)}} \tag{12}$$

Although $\delta(0)$ appears in eq. (11), it will be omitted when calculating the normalized weight $\hat{w}_t^{\prime(i)}$ in eq. (12). That is, the weight calculated without the $\delta(0)$ (right side of the following equation) yields the same normalized weight as follows:

$$\hat{w}_t^{(i)} \propto \hat{w}_{t-1}'^{(i)} \times \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})p(X_t = \hat{X}_t^{(i),(l)}|X_{t-1} = X_{t-1}^{(i)})}{\hat{w}_t'^{(i),(l)}} \quad (13)$$

Therefore, when executing the MCF-PF, eq. (13) can be used to obtain the one-step-ahead importance weight.

However, although an approximation error is included, we can also substitute eq. (7) into eq. (11) to cancel the $\delta(0)$ as follows:

$$\begin{aligned}\hat{w}_t^{(i)} &\approx \hat{w}_{t-1}'^{(i)} \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})\frac{1}{M}\delta(0)}{\hat{w}_t'^{(i),(l)}\delta(0)} \\ &\approx \hat{w}_{t-1}'^{(i)} \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})}{M\hat{w}_t'^{(i),(l)}} \end{aligned} \quad (14)$$

This formulation may be more appropriate that the eq. (13) since the eq (11) is equal to the division by infinity.

Since the importance weights all become $1/N$ after the particle resampling procedure and the sub-particle number M in eq. (14) do not change the resulting normalized weight, above sequential weight equations can be expressed as follows:

$$\hat{w}_t^{(i)} \propto \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})p(X_t = \hat{X}_t^{(i),(l)}|X_{t-1} = X_{t-1}^{(i)})}{\hat{w}_t'^{(i),(l)}} \quad (15)$$

$$\hat{w}_t^{(i)} \propto \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})}{\hat{w}_t'^{(i),(l)}} \quad (16)$$

In either case, the computational cost for performing the SIS algorithm is increased compared to the other SIS realization such as EKF-PF or UKF-PF due to the sub-particle calculation. However, since no assumptions are imposed when constructing the importance distribution such as the Gaussianity, the MCF-PF algorithm is expected to outperform these PFs. We summarize the MCF-PF algorithm for obtaining the predicted and the filtered state estimates in the next section.

C. Algorithm

The algorithm of MCF-PF is shown as follows:

$\mathbf{Monte\ Carlo\ Filter\ Particle\ Filter\ (MCF-PF)}$

$\hat{X}_0^{(i)} \sim p(X_0),\ \hat{w}_0'^{(i)} = 1/N,\ i = 1, 2, \cdots, N$

For each $t = 1, 2, \cdots$

Predicted state estimate

$$\Phi_t^{(i)} \sim p(\Phi_t),\ i = 1, 2, \cdots, N$$

$$\bar{X}_t^{(i)} = f(\hat{X}_{t-1}^{(i)}) + \Phi_t^{(i)}$$

$$\bar{w}_t'^{(i)} = \hat{w}_{t-1}'^{(i)}$$

$$\bar{X}_t = \sum_{k=1}^{N} \bar{w}_t'^{(j)} \bar{X}_t^{(j)}$$

Filtered state estimate

Importance distribution estimation

$$\bar{X}_t^{(i),(j)} \sim p(X_t|X_{t-1} = X_{t-1}^{(i)}), \quad (j = 1, 2, \cdots, M)$$

$$\hat{w}_t^{(i),(j)} = p(Y_t|X_t = \bar{X}_t^{(i),(j)})$$

$$\hat{w}_t'^{(i),(j)} = \frac{\hat{w}_t^{(i),(j)}}{\sum_{k=1}^{M} \hat{w}_t^{(i),(k)}}$$

Sequential importance sampling

$$\hat{X}_t^{(i)} \sim \sum_{j=1}^{M} \hat{w}_t'^{(i),(j)} \delta(X_t - \hat{X}_t^{(i),(j)}), \quad (i = 1, 2, \cdots, N)$$

$$\hat{X}_t^{(i),(l)} \equiv \hat{X}_t^{(i)}$$

$$\hat{w}_t^{(i)} \propto \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})p(X_t = \hat{X}_t^{(i),(l)}|X_{t-1} = X_{t-1}^{(i)})}{\hat{w}_t'^{(i),(l)}}$$

or

$$\hat{w}_t^{(i)} \propto \frac{p(Y_t|X_t = \hat{X}_t^{(i),(l)})}{\hat{w}_t'^{(i),(l)}}$$

$$\hat{w}_t'^{(i)} = \frac{\hat{w}_t^{(i)}}{\sum_{k=1}^{N} \hat{w}_t^{(k)}}$$

$$\hat{X}_t = \sum_{k=1}^{N} \hat{w}_t'^{(k)} \hat{X}_t^{(k)}$$

The algorithm for obtaining the predicted state estimate is the same as that for the MCF. We call the MCF-PF using the weight equation in eq. (15) and that using the weight equation in eq. (16) as MCF-PF-1 and MCF-PF-2, respectively. Their performance difference is numerically investigated in the next section in which we compare the performance of the MCF-PF with those of the other state-of-the-art filters.

IV. NUMERICAL SIMULATIONS

We first investigate the effectiveness of the MCF-PF on the scalar nonlinear state-space models[10] corrupted by Gaussian system and observation noises [Prob.1]. We then examine the performance of the MCF-PF on non-Gaussian system and observation noises such as the Laplace system noise and Cauchy observation noise [Prob.2].

The comparison filters are UKF (unscented Kalman filter), MCF (Monte Carlo filter), GPF (Gaussian particle filter[11]), and UKF-PF (unscented Kalman filter particle filter). Here, the GPF assumes the Gaussian-distributed posterior PDF of state in the MCF algorithm and at every time step, new particles are randomly drawn from that PDF. Therefore the particle resampling procedure required in the MCF algorithm is not necessary for the GPF and the so-called particle impoverishment problem that the almost all of the particles have the same values can be circumvented. The GPF is expected to work with relatively small number of particles, leading to the less computational burden compared to the

MCF. Therefore, we also compare the performance of the MCF-PF with that of the GPF in this section.

The [Prob.1] is on the sequential state estimation problem for the following state-space model.

$$
\begin{aligned}
&[\text{Prob.1}]\\
&x_t = \frac{1}{2}x_{t-1} + \frac{25x_{t-1}}{1+(x_{t-1})^2} + 8\cos(1.2t) + \phi_t,\\
&\text{where } \phi_t \sim \mathrm{N}(\phi_t; 0, 1) = \frac{1}{\sqrt{(2\pi)}}\exp\left(-\frac{\phi_t^2}{2}\right)\\
&y_t = \frac{(x_t)^2}{20} + \omega_t,\\
&\text{where } \omega_t \sim \mathrm{N}(\omega_t; 0, 1) = \frac{1}{\sqrt{(2\pi)}}\exp\left(-\frac{\omega_t^2}{2}\right)
\end{aligned}
$$

We generated the observation data from $t = 1$ to $t = 1000$ based on the initial state $x_0 = 0$. The initial conditions for the filters were all set to $\mathbf{N}(0, 10^2)$. Table I shows average absolute state estimation errors ($\frac{1}{1000}\sum_{t=1}^{1000}|x_t - \hat{x}_t|$) of UKF, MCF, GPF, UKF-PF, MCF-PF-1 and MCF-PF-2 over 100 Monte Calro simulations. Here, the x_t is the true state at time t. Particle numbers for the MCF, GPF, UKF-PF and MCF-PF were all $N = 100$ and the number of sub-particles used in the MCF-PF algorithms was $M = 100$. In Table I, the bold number indicates the smallest state estimation error.

TABLE I

AVERAGE ABSOLUTE STATE ESTIMATION ERRORS OVER 100 MONTE CARLO SIMULATIONS WITH $N = 100$ AND $M = 100$ SETTING.

UKF	MCF	GPF
4.27	1.74	1.77
UKF-PF	MCF-PF-1	MCF-PF-2
1.71	1.81	**1.70**

From the result in Table I, we first confirmed that the PFs are much better than the Gaussian filter represented by the UKF in terms of state estimation accuracy. However, since the UKF was much faster than the other PFs, whether using the UKF or PFs depends on the problem we solve. For example, when the case that the filtering (processing) time is rather important, the UKF still remains as one of the candidate filtering algorithms. Although the GPF was also superior in terms of calculation cost over the other PFs, its state estimation accuracy was very slightly worse than that for the MCF. This was due to the Gaussian assumption on the posterior PDF of state.

The UKF-PF scored the second best estimation accuracy. The MCF-PF-2 was better than the MCF-PF-1. This result indicated that the delta function in the weight equation (11) was better to be eliminated. Although the statistical significance difference was not observed between the results of UKF-PF and MCF-PF-2, the MCF-PF-2 scored the best state estimation accuracy among the comparison filters.

We also executed the same 100 Monte Carlo simulations with less number of particles for all of the PFs to investigate the performance dependency on the particle number. We set $N = 10$ for all of the PFs and $M = 10$ in the MCF-PF algorithm. The results are shown in Table II below and again, the bold number indicates the smallest state estimation error. From Table II, we can see that when the number of particles became small, the GPF became superior over the MCF. This was due to the random sampling effect from the Gaussian-assumed posterior state PDF and the GPF seemed to work for the particle impoverishment problem that was inherent in particle resampling procedure for the MCF algorithm. We also found that the UKF-PF scored the lowest estimation error and this observation leads to the conclusion that when the system and observation noises follow Gaussian distributions, the UKF-PF is the best choice for addressing the filtering problem.

TABLE II

AVERAGE ABSOLUTE STATE ESTIMATION ERRORS OVER 100 MONTE CARLO SIMULATIONS WITH $N = 10$ AND $M = 10$ SETTING.

MCF	GPF	UKF-PF	MCF-PF-1	MCF-PF-2
3.41	3.24	**3.04**	3.43	3.12

The state-space model for the [Prob.2] is described as follows. At this time, the system noise follows a Laplace distribution and the observation noise follows a Cauchy distribution.

$$
\begin{aligned}
&[\text{Prob.2}]\\
&x_t = \frac{1}{2}x_{t-1} + \frac{25x_{t-1}}{1+(x_{t-1})^2} + 8\cos(1.2t) + \phi_t,\\
&\text{where } \phi_t \sim \text{Laplace}(\phi_t; 0, 1) = \frac{1}{2}\exp(-|\phi_t|)\\
&y_t = \frac{(x_t)^2}{20} + \omega_t,\\
&\text{where } \omega_t \sim \text{Cauchy}(\omega_t; 0, 1) = \frac{1}{\pi\left(1+(\omega_t)^2\right)}
\end{aligned}
$$

As well as the previous problem, we again generated the observation data from $t = 1$ to $t = 1000$ based on $x_0 = 0$. The initial conditions for the filters were $\mathbf{N}(0, 10^2)$. Table III shows the average absolute state estimation errors ($\frac{1}{N-10}\sum_{t=11}^{N(=1000)}|x_t - \hat{x}_t|$) of MCF, GPF, UKF-PF, MCF-PF-1 and MCF-PF-2 over 100 Monte Calro simulations. Here, when employing UKF and UKF-PF, we assumed that the Laplace system noise was approximated by $\mathrm{N}(0, 2)$ and also assumed that the Cauchy observation noise was approximated by $\mathrm{N}(0, 10)$, respectively. These Gaussian-assumed PDF settings were required for the UKF and the UKF-PF algorithms. As well as in the previous simulations, we set $N = 100$ for all of the PFs and $M = 100$ for the two MCF-PF algorithms.

From Table III, we can see that the UKF was not capable of filtering observations corrupted by the non-Gaussian noises. MCF was better than the GPF while sacrificing the filtering speed. The UKF-PF sometimes suffered from the problem that all of the importance weights became zero, which implied that sampled particles from the importance distributions were not located within the support domain of the likelihood PDF or the state transition PDF. Because of the large noise realization of the Cauchy observation noise, the observation update for each particle state sometimes produced the deviated importance distributions for each particle, and that made the sampled particles being located out of the aforementioned support domains. This result indicated that for filtering problems of noisy observations, the UKF-PF might be not suitable. The setting of large observation noise variance in the UKF calculation is one remedy for this problem, however, it leads to degrade the entire state estimation accuracy.

TABLE III

AVERAGE ABSOLUTE STATE ESTIMATION ERRORS OVER 100 MONTE CARLO SIMULATIONS WITH $N = 100$ AND $M = 100$ SETTING.

UKF	MCF	GPF
9.34	2.69	2.86
UKF-PF	MCF-PF-1	MCF-PF-2
4.21	2.85	**2.66**

Figure 1 shows the state filtering results for the MCF-PF-2. We can observe two large estimation errors at the noisy

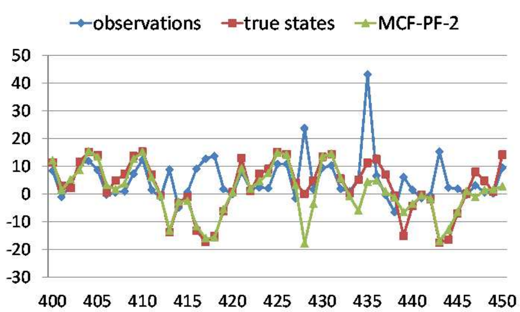

Fig. 1. Plots of observation examples, true states and state estimates of MCF-PF-2. The horizontal axis is the time step from $t = 400$ and $t = 450$.

observations corrupted by the realizations of the additive Cauchy noise. The MCF-PF-2 estimated the other true states well from the nonlinear, bimodal observations.

The MCF-PF-2 was also better than the MCF-PF-1 for this problem. The MCF-PF-2 scored the best state estimation accuracy among the comparison filters. As well as in the previous problem, we also executed the same 100 Monte Carlo simulations with $N = 10$ setting for all of the PFs and $M = 10$ setting in the MCF-PF algorithm. The results are shown in Table IV. From Table IV, when the limited number of particles was allowed, we found that the GPF became superior over the MCF. The UKF-PF again suffered from the problem of importance weights becoming zero after the time proceeded. For this problem, the GPF also outperformed the MCF-PF-2 and it was the fastest filtering algorithm among the five comparison filters. The MCF-PF-2 scored the second best estimation accuracy, showing the effectiveness of the proposed filter.

TABLE IV

AVERAGE ABSOLUTE STATE ESTIMATION ERRORS OVER 100 MONTE CARLO SIMULATIONS WITH $N = 10$ AND $M = 10$ SETTING.

MCF	GPF	UKF-PF	MCF-PF-1	MCF-PF-2
4.06	**3.70**	5.08	4.45	3.80

We found that the low state estimation accuracy of the UKF were implying that the prior state PDF was not better to be assumed as Gaussian. On the other hand, since the GPF was successful, the posterior state PDF could be assumed as Gaussian for resampling new particles.

To summarize the overall results, when the relatively large number of particles was allowed in the filter execution, we confirmed that the MCF-PF-2 was the best among the other PFs. However, for a situation that the small number of particles was desired due to the computational concern, the GPF provided good state estimation results. To the contrary, when the sufficiently large number of particles is allowed, the accuracy difference between MCF and MCF-PF becomes negligible. In that case, since the MCF is much faster than the MCF-PF, the choice of the MCF will be preferred.

V. CONCLUSION

We propose the Monte Carlo filter particle filter (MCF-PF) in which the importance distribution in the SIS algorithm is approximated by the MCF using sub-particles. Therefore, the non-Gaussianity nature of the importance distribution can be incorporated and it leads to the better state estimation accuracy than the other PFs such as GPF and UKF-PF. The numerical simulations show the effectiveness of the MCF-PF and also reveal that when the limited number of particles is allowed, the GPF outputs good state estimation results.

REFERENCES

[1] K. Ito and K. Xiong. “Gaussian Filters for Nonlinear Filtering Problems”, IEEE Trans. on Automatic Control, Vol. 45, No. 5, pp. 910-927, 2000.

[2] A. Jazwinski. “Adaptive Filtering”, Automatica, 5(4), pp. 475-485, 1969.

[3] S. J. Julier, J. K. Uhlmann and H. F. Durrant-Whyte. “A New Approach for Filtering Nonlinear Systems”, In Proc. American Control Conference, pp. 1628-1632, 1995.

[4] A. Doucet. “On Sequential Simulation-based Methods for Bayesian Filtering”, Tech. Rep. CUED/FINFENG/TR 310, Department of Engineering, Cambridge University, 1998.

[5] S. Sarkka. “Bayesian Filtering and Smoothing”, Cambridge University Press, Cambridge CB2 8BS, 2013.

[6] N. Gordon, D. J. Salmond and A. F. M. Smith. “Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation”, IEEE Proceedings-F 140(2); pp. 107-113, 1993.

[7] R. Van der Merwe and N. De Freitas, A. Doucet and E. Wan. “The Unscented Particle Filter”, Advances in Neural Information Processing Systems 13, pp. 584-590, 2001.

[8] A. Doucet, S. J. Godsill and C. Andrieu. “On Sequential Monte Carlo Sampling Methods for Bayesian Filtering”, Statistics and Computing, 10(3), pp. 197-208, 2000.

[9] G. Kitagawa. “Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models”, Journal of Computational and Graphical Statistics, 5(1), pp. 1-25, 1996.

[10] T. Katayama. “Nonlinear Kalman Filter”, Asakura-Shoten, 2011 (in Japanese).

[11] J. H. Kotecha and P. M. Djuric. “Gaussian Particle Filtering”, IEEE Trans. Signal Processing, Vol.51, No.10, pp.2592-2601, 2003.