Gaussian Particle Flow Implementation of PHD Filter

Lingling Zhao[a], Junjie Wang[a], Yunpeng Li[b], and Mark J. Coates[b]

[a]Harbin Institute of Technology, Harbin, China
[b]McGill University, Montreal, Canada

ABSTRACT

Particle filter and Gaussian mixture implementations of random finite set filters have been proposed to tackle the issue of jointly estimating the number of targets and their states. The Gaussian mixture PHD (GM-PHD) filter has a closed-form expression for the PHD for linear and Gaussian target models, and extensions using the extended Kalman filter or unscented Kalman Filter have been developed to allow the GM-PHD to accommodate mildly nonlinear dynamics. Errors resulting from linearization are unavoidable. A particle filter implementation of the PHD filter (P-PHD) is more suitable for nonlinear and non-Gaussian target models. The particle filter implementations are much more computationally expensive and performance can suffer when the proposal distribution is not a good match to the posterior. In this paper, we propose a novel implementation of the PHD filter named the Gaussian particle flow PHD filter (GPF-PHD). It employs a bank of particle flow filters to approximate the PHD; these play the same role as the Gaussian components in the GM-PHD filter but they are better suited to non-linear dynamics and measurement equations. Using the particle flow filter allows the GPF-PHD filter to migrate particles to the dense regions of the posterior, which leads to higher efficiency than the PF-PHD. We explore the performance of the new algorithm through numerical simulations.

Keywords: random set filter, probability hypothesis density filter, particle flow, Gaussian mixture

1. INTRODUCTION

The mathematical foundation for multi-target tracking was developed by Mahler as a systematic means of combining evidence in the presence of uncertainty in a unified way using random finite sets (RFS).[1] The probability hypothesis density (PHD) filter,[1] using random finite sets to model the collections of targets and measurements, can jointly estimate the number of targets and their states from a sequence of observations. It propagates the first-order statistical moment of the joint distribution and is a tractable alternative to the optimal multi-target Bayesian filter. However, in general, it is difficult to obtain closed-form expressions for the PHD filter due to the presence of multi-dimensional integrals. Therefore, sequential Monte Carlo (SMC),[23] and Gaussian mixture implementations of the PHD filter,[4],[56] have been proposed, leading to a number of further developments and applications,[7].[8] The Gaussian mixture implementation of the PHD filter (GM-PHD) is a closed-form solution to the PHD recursion for linear Gaussian multi-target models. Although the GM-PHD filter has been extended to accommodate mildly nonlinear target dynamical models by replacing the Kalman filter with the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF),[5] the sequential Monte Carlo implementation of the PHD filter, also called the particle PHD filter (PF-PHD), is still a better-performing solution for nonlinear and/or non-Gaussian scenarios.

In its basic form, the prior is used as the proposal distribution in the PF-PHD filter. In many cases, the prior distribution has a much larger variance than the posterior, because the most recent measurements are highly informative about the state. Particles proposed using the prior frequently have very low likelihoods and their contributions to the posterior estimation become negligible, which leads to a large Monte Carlo variance. The inefficient proposal distribution in the basic particle-PHD filter leads to a need for a large number of particles. The UKF-PHD filter[9] and auxiliary particle PHD filter,[1011] have been proposed to address these issues. They both take the most recent observations into consideration in constructing a proposal distribution.

Further author information: (Send correspondence to Mark Coates)
Lingling Zhao: E-mail: zhaoll@hit.edu.cn, Telephone: 86 0451 86412824
Mark Coates: E-mail: Mark.Coates@mcgill.ca, Telephone:1 514 398 7137

Particle filters struggle to perform well in relatively high dimensional state spaces unless a very large number of particles is employed. To avoid the severe degeneracy induced by sampling in a high dimensional space, particle flow algorithms were recently proposed. These are based on the idea of constructing bridging distributions and involve identifying a particle flow via a partial differential equation. A series of papers have been proposed to provide different solutions to the problem based on different assumptions,[12],[13].[14] The particle flow approach can lead to improved filter performance especially for high-dimensional state spaces or highly informative measurements.

Inspired by these advantages, we propose a novel PHD filter implementation based on the particle flow filter. We refer to the filter as the Gaussian particle flow PHD filter (GPF-PHD). The algorithm employs a bank of particle flow filters to update the Gaussian components in a GM-PHD filter. Each particle flow filter independently migrates its particles according to the latest observations. Then the PHD is approximated by the weighted particle groups after the flow. The use of the particle flow filter allows the GPF-PHD filter to achieve a better match to the true posterior by migrating particles continuously from the prior distribution to the posterior distribution. The evolution of each particle flow filter replaces the evolution of the Gaussian components in the GM-PHD. The resulting algorithm is much less computationally demanding the PF-PHD but can adapt equally well to non-linearities.

The rest of paper is organized as follows. Section 2 reviews the particle flow filter formulation and the PHD filter. Section 3 gives a detailed description on the proposed GPF-PHD, presenting both linear and nonlinear versions. Simulation results are provided in Section 4. Section 5 concludes the paper.

2. PROBLEM STATEMENT AND BACKGROUND

We address a multi-target PHD filter implementation task with the following models and assumptions. In the multi-target tracking problem, a time-varying and unknown number of targets evolve in the state space. Since their number and states vary in time, the states of T_k tracked targets at time k can be naturally represented as a random finite set $\Gamma_k = \{x_{k,1}, ..., x_{k,T_k}\}$, where $x_{k,i}$ is the state of the ith individual target, assumed to follow a Markov process on the state space with transition density $f_{k|k-1}(x|x_{k-1})$. Each target $x_{k,i}$ can either be detected with detection probability $p_{D,k}(x_{k,i})$ or goes undetected with probability $1 - p_{D,k}(x_{k,i})$. For the detected targets, each is observed by a measurement $z_{k,i}$ in the observation space governed by likelihood density $g_k(z_k|x_k)$. A total of M_k measurements are observed at time k and these can similarly be represented by a random finite set $Z_k = \{z_{k,1}, ..., z_{k,M_k}\}$, noting that $z_{k,i}$ can be an observation from a target or due to clutter. The goal of multi-target Bayes filtering is to estimate the multi-target marginal posterior density $p_k(\cdot|Z_{1:k})$.

Since the computation of the multi-target posterior density $p_k(\cdot|Z_{1:k})$ is usually intractable, Mahler developed the PHD filter[1] to propagate the first-order statistical moment of the multi-target posterior density $p_k(X_k|Z_{1:k})$. This is called the probability hypothesis density (PHD), and is denoted by $D_{k|k}$. The PHD filter evolves over time by a prediction step and an update step, and the prediction operator is defined by

$$D_{k|k-1}(x|Z_{1:k}) = \gamma_k(x) + \int \varphi_{k|k-1}(x, x_{k-1}) D_{k-1|k-1} dx_{k-1}\,. \tag{1}$$

Here γ_k is the intensity function of a newborn targets and $\varphi_{k|k-1}(x, x_(k-1)) = P_{S,k}(x_{k-1}) f_{k|k-1}(x|x_{k-1}) + b_{k|k-1}(x|x_{k-1})$, with $b_{k|k-1}$ denoting the PHD of a spawned target, probability of target survival $P_{S,k}$ is state independent. $D_{k|k}(x|Z_{1:k})$ is abbreviated in the remainder of the paper by $D_{k|k}(x) = D_{k|k}(x|Z_{1:k})$ and is updated by the following equation:

$$D_{k|k}(x) = \nu(x) + \left[\sum_{z \in Z_k} \frac{\psi_{k,z}(x)}{\kappa_k(z) + \langle D_{k|k-1}, \psi_{k,z} \rangle} \right] D_{k|k-1}(x)\,. \tag{2}$$

Here $\nu(x) = 1 - P_{D,k}(x)$ is the probability of a missed detection, and $\psi_{k,z} = P_{D,k}(x) h_k(z|x)$. The PHD of the clutter RFS is denoted by $\kappa_k(z)$ and in this model we have $\kappa_k(z) = \lambda_k C_k(z)$, which means that the clutter points in the surveillance region follow a probability distribution $C_k(z)$ and the average number of clutter per scan is λ_k. Finally, $\langle D_{k|k-1}, \psi_{k,z} \rangle$ is used to denote $\int D_{k|k}(x_k|Z_(1:k))\phi(x_k)d(x_k)$ For the PHD filter, we impose the three assumptions A.1-A.3, as proposed in the Gaussian-mixture PHD filter paper by Vo et al.[5]

Our task is to design a new implementation of the PHD filter using the particle flow technique to improve its filtering performance in some challenging situations such as a high state dimension, highly informative measurements or strongly nonlinear dynamical models.

3. THE GAUSSIAN PATICLE FLOW IMPLEMENTATION

This section describes a particle flow filter implementation of the PHD filter under Gaussian assumption. The idea is to propagate the intensity function represented by a group of particle sets, where each particle set is controlled by an individual particle flow filter approximating a Gaussian component in the GM-PHD framework.

In formulating the Gaussian particle flow PHD filter, we impose the additional assumptions: A.4: Each target follows a Gaussian dynamical model and the measurement model is also Gaussian, i.e.

$$f_{k|k-1}(x|x(k-1)) = p(x; f_k(x_{k-1}, 0), Q_{k-1}), \tag{3}$$
$$g_k(z|x) = p(z; h_k(x, 0), R_k). \tag{4}$$

Here $p(.; m, P)$ denotes a Gaussian density with mean m and covariance P. f_k and h_k can be linear or nonlinear functions. To accommodate nonlinear models, we adapt the proposed particle flow PHD filter by local linearization of the mapping h_k in the particle flow motion. This idea is also inspired by the extension of Gaussian mixture PHD filter to nonlinear Gaussian models. However, in the particle flow PHD filter, the covariance of the state vector is approximated by the sample covariance of the particles belonging to each Gaussian component, so there is no need for linearization of the dynamics.

A.5: The birth and spawn RFSes are formed as Gaussian mixtures:[5]

$$\gamma_k(x) = \sum_{i=1}^{J_{\gamma,k}} w^i_{\gamma,k} p(x; m^i_{\gamma,k}, P^i_{\gamma,k}); \qquad \beta_k(x|x_{k-1}) = \sum_{i=1}^{J_{\beta,k}} w^i_{\beta,k} p(x; F^i_{\beta,k-1} x_{k-1} + d^i_{\beta,k-1}, Q^i_{\beta,k}) \tag{5}$$

The parameters $J_{\gamma,k}, J_{\beta,k}$ specify the number of components in each mixture; the Gaussian mixture parameters (weights, means, covariances) are $w^i_{\gamma,k}, m^i_{\gamma,k}, P^i_{\gamma,k}), w^i_{\beta,k}, Q^i_{\beta,k}$, and $F^i_{\beta,k-1}$ and $d^i_{\beta,k-1}$ specify the linear spawning dynamics. Further detail on these parameters is provided by Vo et al.[5] For the Gaussian multiple-target model, the posterior intensity at time $k-1$ is represented by a Gaussian mixture of the form

$$D_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w^i_{k-1} N(x; m^i_{k-1}, P^i_{k-1}) \tag{6}$$

We denote each Gaussian component $N(x; m^i_{k-1}, P^i_{k-1})$ as $D^i_{k-1}(x)$, which allows us to rewrite $D_{k-1}(x)$ as

$$D_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w^i_{k-1} D^i_{k-1}(x) \tag{7}$$

3.1 The prediction operator

We draw L_{k-1} samples $\{x\}^{i,j}_{k-1}$ from $N(x; m^i_{k-1}, P^i_{k-1})$, that is, the Gaussian mixture component $D^i_{k-1}(x)$ of $D_{k-1}(x)$, in order to form a particle approximation:

$$D^i_{k-1}(x) \leftarrow \frac{1}{L_{k-1}} \sum_{j=1}^{L_{k-1}} \delta_{x^{i,j}_{k-1}(x_{k-1})}. \tag{8}$$

The predicted intensity $D_{k|k-1}(x)$ is also a Gaussian mixture and is given by:

$$D_{k|k-1}(x) = D_{S,k|k-1}(x) + \gamma_k(x) + D_{\beta,k|k-1}(x) \tag{9}$$

Its particle approximation can be then derived by handling each of its terms separately. For the predicted survived PHD term $D_{S,k|k-1}(x)$, we have

$$D_{S,k|k-1}(x) = P_{S,k} \int p(x; f_{k|k-1}(\xi,0), Q_{k-1}) \sum_{i=1}^{J_{k-1}} w_{k-1}^i p(\xi; m_{k-1}^i, P_{k-1}^i) d\xi$$
$$= P_{S,k} \sum_{i=1}^{J_{k-1}} w_{k-1}^i \int p(x; f_{k|k-1}(\xi,0), Q_{k-1}) p(\xi; m_{k-1}^i, P_{k-1}^i) d\xi \quad (10)$$

3.2 The particle flow operator for each intensity component

To allow for a cleaner description of the filter, we use the common terminology $D_{k|k-1}^i(x)$ for any mixture component of the predicted intensity, whether it correspond to a surviving component, $D_{S,k|k-1}^i(x)$, a newborn component, $D_{\gamma,k}^i(x)$, or a spawned component, $D_{\beta,k|k-1}^{i,j}(x)$. We also denote the predicted weight by $w_{k|k-1}^i$, which can correspond to survival, $w_{k|k-1}^i = P_{S,k} w_{k-1}^i$, birth, $w_{k|k-1}^i = w_{\gamma,k}^i$ or spawning, $w_{k|k-1}^i = w_(\beta,k)^i w_{k-1}^i$. We can then express the predictive posterior hypothesis density as:

$$D_{k|k-1}(x) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^i D_{k|k-1}^i(x) \quad (11)$$

Each intensity component $D_{k|k-1}^i(x)$ is predicted from $D_{k-1|k-1}^i(x)$ and approximated by $\frac{1}{L_{k-1}} \sum_{j=1}^{L_{k-1}} \delta_{x_{k|k-1}^{i,j}}(x_{k|k-1})$. Since it is predicted independently from other particle sets and is updated by each observation separately, its predict and update process can be considered as a single-target single-observation tracking problem. For intensity components representing the survived target cases, the transition density follows $x_k = f_k(x_{k-1}, v_{S,k})$, for the spawn target cases $x_k = b_k(x_{k-1}, v_{\beta,k})$ and for the new-born target cases $x_k = \gamma_k(m_{\gamma,k}^i, v_{\gamma,k})$. Irrespective of the origin of a predicted intensity component, it should be updated by each observation based on the same observation model $z_k = h_k(x_k) + w_k$, according to the PHD filter update equations, thus forming M_k updated components.

After the samples $\left\{x_{k|k-1}^{i,j}\right\}_{j=1}^{L_{k-1}}$ approximating the intensity component $D_{k|k-1}^i(x)$ have been propagated by a transition density, they approximate its corresponding prior distribution at time k (we denote these as $\{\tilde{u}_j\}_{j=1}^{L_{k-1}}$). To migrate particles from the prior to the posterior related to each observation $z_{k,i}, i = 1, \ldots, M_k$ effectively and efficiently, we apply particle flow equations to obtain a particle set $\{\tilde{u}_j\}_{j=1}^{L_{k-1}}$ distributed approximately according to the posterior distribution.

According to the Bayes' rule, the posterior distribution $p(x) = \frac{f(x)l(x)}{K}$, where $f(x)$ is the prior distribution for the M-dimensional state x, $l(x)$ is the likelihood function, and K is a normalizing constant. A homotopy function $\varphi(x,\lambda)$ can be defined as

$$\varphi(x,\lambda) = log(\tilde{p}(x)) = \log(f(x)) + \lambda log(l(x)) \quad (12)$$

Here the normalizing constant K is ignored. Then a map $\oslash : X \times [0,1] \to X'$ is built, such that $\varphi(x,0) = log(f(x))$ and $\varphi(x,1) = log(p(x,\lambda))$. Then through this continuous deformation, the log prior distribution and unnormalized log posterior distribution can be linked now.

Under different assumptions a series of variants of Daum-Huang filters[14] are developed, for example, incompressible flow Daum and Huang (DH) filter assumes that particles move with the condition that $\varphi(x,\lambda)$ stays constant with respect to λ. With the chain rule, the incompressible flow of particles can be defined by equation 13.

$$\frac{d\varphi(x,\lambda)}{d\lambda} = \frac{\partial\varphi(x,\lambda)}{\partial x}\frac{dx}{d\lambda} + \frac{\partial\varphi(x,\lambda)}{\partial\lambda} = \frac{\partial\varphi(x,\lambda)}{\partial x}\frac{dx}{d\lambda} + log(l(x)) = 0 \quad (13)$$

There are many methods to solve this partial differential equation (PDE)[14] proposes using the generalized inverse to obtain the unique minimum solution of the induced flow, which shows that the flow is in the direction of the gradient of the log homotopy, with speed proportional to $log(l(x))$.

A generalized version of DH filters has been derived by Daum and Huang for the linear Gaussian scenario, that is so-called exact particle flow DH filter. Since functions $f(x,\lambda) = dx/d\lambda$ and $\Psi(x,\lambda) = f(x)l(x)^{\lambda}$, according to the Fokker-Planck equation and the chain rule, we can get

$$-Tr(\frac{\partial f}{\partial x}) = \frac{\partial \varphi(x,\lambda)}{\partial x} f + log(l(x)) \tag{14}$$

Its closed-form solution of Gaussian case is

$$\frac{dx}{d\lambda} = A(\lambda)x + b(\lambda) \tag{15}$$

where $A(\lambda) = -\frac{1}{2}PH^T(\lambda HPH^T + R)^{-1}H$, $b(\lambda) = (I + 2\lambda A)PH^T R^{-1} z_k + A\bar{x}$

Here $\bar{x}$ is the mean of the prior distribution. P is the covariance matrix of the prediction error for the prior distribution, which can be estimated by the sample covariance matrix, or through an extended or unscented Kalman filter (EKF/UKF). Particularly, for nonlinear models, H can be approximated by the linearization of the measurement model, i.e. $H = \frac{\partial h_k(\mu,0)}{\partial \mu}$. R is the covariance matrix of the measurement error. Pseudo-codes of two typical exact flow algorithms under the Gaussian assumption, the EDH[15] and the LEDH, are both presented in.[16] Consequently, to obtain the particle approximation for each intensity component $\Delta D^i_{k|k}(x)$, we can directly migrate particles $\tilde{u_j}_{j=1}^{L_{k-1}}$ to ${u_j}_{j=1}^{L_{k-1}}$ in small steps using Euler's method:

$$\mu_j(\lambda_l) = \mu_j(\lambda_{l-1}) + \Delta_l(A(\lambda_{l-1})\mu_j(\lambda_{l-1}) + b(\lambda_{l-1})) \tag{16}$$

Where the step size at the l-th iteration $\Delta_l = \lambda_l - \lambda_{l-1}$, the iteration starts with$\lambda_0 = 0$(corresponding to particle set $\left\{x^j_{k|k-1}\right\}_{j=1}^{L_{k-1}}$) and ends when $\lambda_N = 1$, then the particles after migration are just $\left\{x^{i,j}_{k|k}\right\}_{j=1}^{L_{k-1}}$. Then,

$$m^i_{k|k} = \frac{\sum_{j=1}^{L_{k-1}} x^{i,j}_{k|k}}{L_{k-1}}, P^i_{k|k} = \sum_{j=1}^{L_{k-1}} (x^{i,j}_{k|k} - m^i_{k|k})(x^{i,j}_{k|k} - m^i_{k|k})^T \tag{17}$$

It has been proved that $m^i_{k|k}$ and $P^i_{k|k}$ converge to the MMSE of the mean and covariance of the Gaussian posterior. Then a set of particles${x^{i,j}_{k|k}}_{j=1}^{L_{k-1}}$(rewritten as $\{\mu_j\}_{j=1}^{L_{k-1}}$will be generated.

3.3 The update operator

The posterior intensity at time k can be given by

$$D_k(x) = (1 - P_{D,k})D_{k|k-1}(x) + \sum_{z \in Z_k} D_(D,k)(x|z) \tag{18}$$

Based on all the resultant updated intensity components $\Delta D^i_{k|k}(x) = \frac{1}{L_{k-1}} \sum_{j=1}^{L_{k-1}} \delta_{x^{i,j}_{k|k}}(x_{k|k})$, the posterior intensity can be rewritten as

$$D_{D,k}(x|z) = \sum_{i=1}^{J_{k|k-1}} w^i_{k|k}(z) \Delta D^i_{k|k}(x) \tag{19}$$

where

$$w^i_{k|k}(z) = \frac{(P_{D,k} w^i_{k|k-1} g_k(z m^i_{k|k-1}))}{(\kappa_k(z) + P_{D,k} \sum_{l=1}^{J_{k|k-1}} w^l_{k|k-1} g_k(z m^l_{k|k-1}))} \tag{20}$$

Daniel Clark and Ba-Ngu Vo showed each step in time of the PHD filter will maintain a suitable approximation error that converges to zero as the number of Gaussians in the mixture tends to infinity[6] , while each approximated intensity component $\Delta D^{i}_{k|k}(x)$ by particles via flow has been proved to converge to the corresponding Gaussian distribution, therefore, it can be known the convergence of the proposed Gaussian particle flow PHD filter.

3.4 Implementation issues

Similar to the Gaussian mixture PHD filter, the particle flow PHD filter also suffers from the computational consumption problems resulted from increasing J_k as time evolves and particle flow computational cost. For the increasing number of approximated ΔD_k, a similar pruning procedure can be exploited by discarding the $\Delta D_{k|k}$ with light associated probabilities or merging those close enough to each other into one $\Delta D_{k|k}$. For the particle flow computations, we can save some computations by only performing particle flow migrating on those $\Delta D_{k|k}$ with higher associated weights. Since the influence of a $\Delta D_{k|k}$ with low associated weight on the whole PHD approximation is negligible, its flow computation is unnecessary as well. The framework of Gaussian particle flow PHD filter is presented in algorithm 1.

For the particle flow motion step of algorithm 2, localized exact Daum-Huang filter can be also applied to get more reasonable linearized Hessen matrix for each particle instead of the unique Hessen matrix at the mean.

4. SIMULATION AND RESULTS

4.1 Simulation setup

In order to further verify the performance of the proposed GPF-PHD filter in the nonlinear problems, we exploit the MTT simulations using the bearing and range tracking model. Consider over the $[-1000, 1000] \times [-1000, 1000]$region, targets move according to the linear Gaussian dynamics as equation(14). Where the target state $x_k = [x_{1,k}, x_{2,k}, x_{3,k}, x_{4,k}]^T$consists of the position $[x_{1,k}, x_{3,k}]^T$ and the velocity $[x_{2,k}, x_{4,k}]^T$ at time step k, and the sampling period $T = 1$s.The $v_{1,k}$ and $v_{2,k}$ denote the process noise which are mutually independently zero-mean Gaussian white noise with the standard deviation $\sigma_{v_1} = 1$ and $\sigma_{v_2} = 0.1$ respectively. No spawning is considered in our simulations. The process of the new targets birth are Poisson point process with intensity function $\gamma_k = 0.2p(\cdot; \bar{x}, Q)$, where $\bar{x} = [0, 3, 0, -3]^T$, and $Q = diag([10, 1, 10, 1]^T)$. The sensor, located at $[-100, -100]^T$, measures the targets according to the following measurement equations and

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} T^2/2 & 0 \\ 1 & 0 \\ 0 & T^2/2 \\ 0 & 1 \end{bmatrix} \omega_k \tag{21}$$

$$\theta_k = \arctan\left(\frac{x_{3,k} + 100}{x_{1,k} + 100}\right) + w_{1,k}, r_k = \left\| \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x_k - \begin{bmatrix} -100 \\ -100 \end{bmatrix} \right\| + w_{2,k} \tag{22}$$

Where θ_k denotes the bearing measurement and r_k denotes the range measurement. The $w_{1,k}$ and $w_{2,k}$ denote the measurement noise which are mutually independently zero-mean Gaussian white noise with the standard deviation $\sigma_{w_1} = 0.0005$ and $\sigma_{w_2} = 0.1$ respectively. Under the assumption that there is no loss of detection, we set the probability of detection $P_D = 1$. The sensor also gives some false measurements called clutters, which are uniformly distributed over the observation space $[-\pi, \pi] \times [-1000, 1000]$ with average rate of r points per scan. The number of the particles, sampled from the intensity function $\gamma_k = 0.2(\cdot; \bar{x}, Q)$, $L_0 = 1000$ in the initialization. In our simulations, the number of targets is 6.

4.2 Experimental results

To verify the clutter influence on these PHD filters performance, tracking scenarios with average clutter of rate are tested separately with the GM-PHD, the PF-PHD filter, the auxiliary PF-PHD filter(APF-PHD) and the GPF-PHD filters. For the GM-PHD and GPF-PHD, pruning and merge algorithm is performed with a same weight threshold of $W_th = 0.0001$ and distance threshold $D_th = 4m$.400 particles are assigned to approximate a Gaussian term in the GPF-PHD and different number of particles for each target in the PF-PHD. 50 independent

Algorithm 1 Pseudo-code for Gaussian particle flow PHD filter

Given $\left\{x_{k-1}^{i,j}\right\}_{i=1,\ldots,J_{k-1},j=1,\ldots,L_{k-1}}, \left\{w_{k-1}^{i}\right\}_{i=1,\ldots,L_{k-1}}$

Step1: prediction for survival targets

$i = 0$

for $l = 1$ to L_{k-1} **do**

  $i = i + 1$

  for $j = 1$ to J_{k-1} **do**

    $x_{k|k-1}^{i,j} = f(x_{k-1}^{l,j}, 0)$

  end for

  $w_{k|k-1}^{i} = w_{S,k}^{i} w_{k-1}^{i}, m_{k|k}^{i} = \frac{\sum_{j=1}^{L_{k-1}} x_{k|k}^{i,j}}{L_{k-1}}, P_{k|k}^{i} = \frac{\sum_{j=1}^{L_{k-1}} (x_{k|k}^{i,j} - m_{k|k}^{i})(x_{k|k}^{i,j} - m_{k|k}^{i})^T}{L_{k-1}}$

end for

Step2: prediction for birth targets

for $l = 1$ to $J_{\gamma,k}$ **do**

  $i = i + 1$

  Randomly draw L_{k-1}particles $x_{k|k-1}^{i,j}$from $p(x; m_{\gamma,k}^{l}, P_{\gamma,k}^{l})$

  $w_{k|k-1}^{i} = w_{\gamma,k}^{l}, m_{k|k-1}^{i} = m_{\gamma,k}^{l}, P_{k|k-1}^{i} = P_{\gamma,k}^{l}$

end for

$J_{k|k-1} = i$

Step 3: Update for undetected targets

for $l = 1$ to $J_{k|k-1}$ **do**

  $w_{k}^{l} = (1 - P_{D,k}) w_{k|k-1}^{l}$ $,P_{k|k}^{l} = P_{k|k-1}^{l}, m_{k|k}^{l} = m_{k|k-1}^{l}$

  for $j = 1$ to L_{k-1} **do**

    $x_{k|k}^{l,j} = x_{k|k-1}^{l,j}$

  end for

end for

Step 4:Update for detected targets

$i = 0$

for each observation $z \in Z_k$ **do**

  $i = i + 1$

  for $l = 1$ to $J_{k|k-1}$ **do**

    $H_{k}^{l} = \frac{\partial h_k(x_k, 0)}{\partial x_k}|_{x_k = m_{k|k-1}^{l}}$

    $\eta_{k|k-1}^{l} = h_k(m_{k|k-1}^{l}, 0)$

    $S_{k}^{l} = R_k + H_{k}^{l} P_{k|k-1}^{l} H_{k}^{l\,T}$

    $w_{k}^{iJ_{k|k-1}+l} = P_{D,k} w_{k|k-1}^{l} q(z; \eta_{k|k-1}^{l}, S_{k}^{l})$

  end for

  for $l = 1$ to $J_{k|k-1}$ **do**

    $w_{k}^{iJ_{k|k-1}+l} = \frac{w_{k}^{iJ_{k|k-1}+l}}{\kappa_k(z) + \sum_{j=1}^{J_{k|k-1}} w_{k}^{iJ_{k|k-1}+l}}$

  end for

  for $l = 1$ to $J_{k|k-1}$ **do**

    $x_{k}^{iJ_{k|k-1}+l} = \text{particleFlow}(\left\{x_{k|k-1}^{l,j}\right\}, P_{k|k-1}^{l}, H_{k}^{l}, m_{k|k-1}^{l}, R_k, z)$

    $m_{k|k}^{iJ_{k|k-1}+l} = \frac{\sum_{j=1}^{L_{k-1}} x_{k|k}^{iJ_{k|k-1}+l}}{L_{k-1}}$

    $P_{k|k}^{iJ_{k|k-1}+l} = \frac{1}{L_{k-1}} (x_{k|k}^{iJ_{k|k-1}+l} - m_{k|k}^{iJ_{k|k-1}+l})(x_{k|k}^{iJ_{k|k-1}+l} - m_{k|k}^{iJ_{k|k-1}+l})^T$

  end for

end for

$J_k = iJ_{k|k-1} + J_{k|k-1}$

$L_k = L_{k-1}$

output: $\left\{x_{k}^{i,j}\right\}_{i=1,\ldots,J_k,j=1,\ldots,L_k}, \left\{w_{k}^{i}, m_{k}^{i} P_{k}^{i}\right\}_{i=1,\ldots,L_k}$

Algorithm 2 Pseudo-code for particle flow motion algorithm

Given $\{x_j\}_{j=1,\ldots,L_{k-1}}, P, m, H_k, R_k, z$
Step1: particle motion
for all particles **do**
  $\mu_j^i = x_j, \bar{\mu}_j = m$
end for
for $i = 1$ to N_λ **do**
  $\lambda = n \cdot \Delta\lambda$
  $H_\mu = H_k m$
  for $j = 1$ to L_{k-1} **do**
    $A = -\frac{1}{2} P H_k^T (\lambda H_k P H_k^T + R_k)^{-1} H_k$
    $b = (I + 2\lambda A)[(I + \lambda A) P H_k^T R_k^{-1} z + Am]$
    Migrate particles:$\mu_j^i = \mu_j^i + \frac{\Delta \mu_j^i}{d\lambda}$
  end for
  Re-evaluate $\bar{\mu}_j = \frac{\sum_{j=1}^{L_{k-1}} \mu_j^i}{L_{k-1}}$
end for
Update P using the sample covariance matrix of sample set μ_j^i, EKF, or UKF;
Step2:optional redraw operation
Redraw particles $x_j \sim N(\mu_j^i, P), i^j = (j-1)\rho + 1, .., j\rho$ for each estimated target state $\hat{x}_{k,j}$.
output: $\{x_j\}_{j=1,\ldots,L_{k-1}}, P, m$

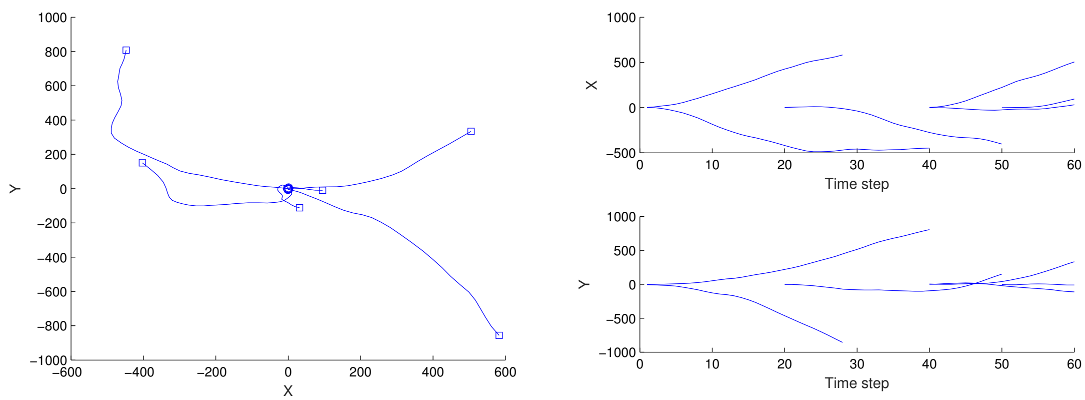

Figure 1. The true trajectory with $r = 10$

Monte Carlo runs are executed and the average results are shown in Fig. 2 and listed in Table 1. When there is no clutter in the surveillance region, these kinds of filters show similar estimation performance due to the accurate target number estimation.As the clutter rate increases, more particles are required for the PF-PHD filter and the APF-PHD to track all targets while the GPF-PHD filter still maintains steady tracking performance and exhibits the smaller average tracking error than GM-PHD with different clutter rates as shown in Fig. 2 and table 1. Compared to the general PF-PHD filter and APF-PHD filter, the GPF-PHD filter provides comparative or better tracking performance in term of tracking error (OSPA), target number estimation error as well as significantly less computational cost. In addition, as the clutter rate increases, the degradation in performance of three filters happen unavoidably in table 1. It should be noted that underestimation happens in the PF-PHD and APF-PHD filter due to clutter influence if the particles for each target are not enough.

To further confirm the reason why particle flow motion helps PHD filter achieves better tracking performance, we observe the particle distributions before and after particle flow motion as shown in Fig. 3, where the gray dots are original particles after prediction, the black dots are the particles after particle flow motion, and the red dots denote the real targets. It's observed that through particle flow migration, particles spread more concentrated

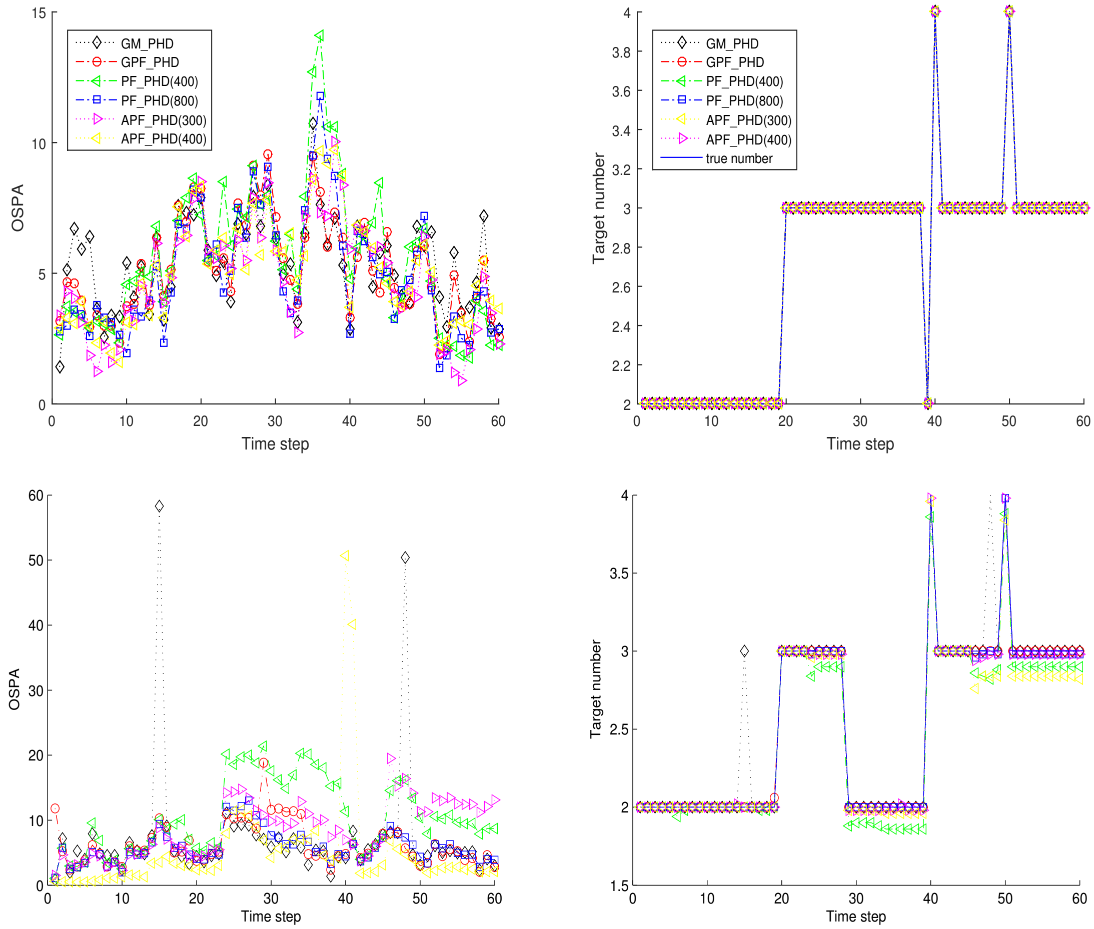

Figure 2. The average results on OSPA and estimated target number with $r = 0$ and 10

on the high probability region of the posterior. This improvement leads to better approximation of the posterior.

5. CONCLUSTIONS

This paper presents a particle flow filter implementation of the PHD filter, which propagates a bank of particle flow filters in the Gaussian mixture PHD filter framework. Particle flow motion drives particles moving close to the real posterior, leading to better tracking performance compared to the general PF-PHD filter as well as better adaption for nonlinear problems than the GM-PHD. This work will also motivate the further parallelization of PHD filter, since resampling operation is not necessary in GPF-PHD filter.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (NSFC,Grant no. 61305013).

REFERENCES

[1] Mahler, R. P., "Multitarget bayes filtering via first-order multitarget moments," *Aerospace and Electronic Systems, IEEE Transactions on* **39**(4), 1152–1178 (2003).

[2] Zajic, T. and Mahler, R. P., "Particle-systems implementation of the phd multitarget-tracking filter," in [*AeroSense 2003*], 291–299, International Society for Optics and Photonics (2003).

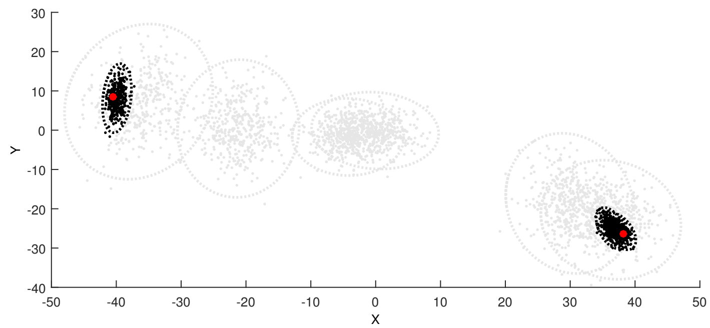

Figure 3. The particle distribution before and after flow

Table 1. Average results over 50 runs with different clutter of rate

Clutter rate	Filter	OSPA mean	OSPA std	Execution time
r=0	PF-PHD(400)	5.6081	2.6873	0.3089
	PF-PHD(800)	5.0066	2.2963	0.6473
	APF-PHD(400)	5.1104	2.0372	0.6586
	GM-PHD	5.3869	1.7768	0.0020
	GPF-PHD	5.1875	1.9338	0.1559
r=10	PF-PHD(400)	17.6414	12.9911	1.1608
	PF-PHD(800)	5.4803	2.5118	2.4510
	APF-PHD(400)	5.5444	2.7381	2.6863
	GM-PHD	7.0912	9.1201	0.0045
	GPF-PHD	5.7021	2.4740	0.2071
r=50	PF-PHD(400)	18.3675	19.2078	5.6176
	PF-PHD(800)	14.0495	19.1775	11.5254
	APF-PHD(400)	15.2785	18.7407	11.6077
	GM-PHD	21.5488	24.6043	0.0214
	GPF-PHD	15.9692	20.1502	0.5072

[3] Vo, B.-N., Singh, S., and Doucet, A., "Sequential monte carlo methods for multitarget filtering with random finite sets," *Aerospace and Electronic Systems, IEEE Transactions on* **41**(4), 1224–1245 (2005).

[4] Vo, B.-N. and Ma, W.-K., "A closed-form solution for the probability hypothesis density filter," in [*Information Fusion, 2005 8th International Conference on*], **2**, 8–pp, IEEE (2005).

[5] Vo, B.-N. and Ma, W.-K., "The gaussian mixture probability hypothesis density filter," *Signal Processing, IEEE Transactions on* **54**(11), 4091–4104 (2006).

[6] Clark, D. and Vo, B.-N., "Convergence analysis of the gaussian mixture phd filter," *Signal Processing, IEEE Transactions on* **55**(4), 1204–1212 (2007).

[7] Vo, B.-N., Pasha, A., and Tuan, H. D., "A gaussian mixture phd filter for nonlinear jump markov models," in [*Decision and Control, 2006 45th IEEE Conference on*], 3162–3167, IEEE (2006).

[8] Granström, K. and Orguner, U., "A phd filter for tracking multiple extended targets using random matrices," *Signal Processing, IEEE Transactions on* **60**(11), 5657–5671 (2012).

[9] Melzi, M., Ouldali, A., and Messaoudi, Z., "The unscented kalman particle phd filter for joint multiple target tracking and classification," in [*Signal Processing Conference, 2011 19th European*], 1415–1419, IEEE (2011).

[10] Whiteley, N., Singh, S., and Godsill, S., "Auxiliary particle implementation of probability hypothesis density filter," *Aerospace and Electronic Systems, IEEE Transactions on* **46**(3), 1437–1454 (2010).

[11] Baser, E. and Efe, M., "A novel auxiliary particle phd filter," in [*Information Fusion (FUSION), 2012 15th International Conference on*], 165–172, IEEE (2012).

[12] Daum, F. and Huang, J., "Nonlinear filters with log-homotopy," in [*Optical Engineering+ Applications*], 669918–669918, International Society for Optics and Photonics (2007).

[13] Daum, F., Huang, J., and Noushin, A., “Exact particle flow for nonlinear filters,” in [*SPIE Defense, Security, and Sensing*], 769704–769704, International Society for Optics and Photonics (2010).

[14] Daum, F. and Huang, J., “Particle flow with non-zero diffusion for nonlinear filters,” in [*SPIE Defense, Security, and Sensing*], 87450P–87450P, International Society for Optics and Photonics (2013).

[15] Choi, S., Willett, P., Daum, F., and Huang, J., “Discussion and application of the homotopy filter,” in [*SPIE Defense, Security, and Sensing*], 805021–805021, International Society for Optics and Photonics (2011).

[16] Ding, T. and Coates, M. J., “Implementation of the daum-huang exact-flow particle filter,” in [*Statistical Signal Processing Workshop (SSP), 2012 IEEE*], 257–260, IEEE (2012).