ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-6874-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSP49660.2025.10890621

Fusion of Information in Multiple Particle Filtering in the Presence of Unknown Static Parameters

Xiaokun Zhao, Marija Iloska, Yousef El-Laham, Mónica F. Bugallo
Department of Electrical and Computer Engineering
Stony Brook University
Stony Brook, NY, USA
{xiaokun.zhao, marija.iloska, yousef.ellaham, monica.bugallo}@stonybrook.edu

***Abstract*—An important and often overlooked aspect of particle filtering methods is the estimation of unknown static parameters. A simple approach for addressing this problem is to augment the unknown static parameters as auxiliary states that are jointly estimated with the time-varying parameters of interest. This can be impractical, especially when the system of interest is high-dimensional. Multiple particle filtering (MPF) methods were introduced to try to overcome the *curse of dimensionality* by using a "divide and conquer" approach, where the vector of unknowns is partitioned into a set of subvectors, each estimated by a separate particle filter. Each particle filter weighs its own particles by using predictions and estimates communicated from the other filters. Currently, there is no principled way to implement MPF methods where the particle filters share unknown parameters or states. In this work, we propose a fusion strategy to allow for the sharing of unknown static parameters in the MPF setting. Specifically, we study the systems which are separable in states and observations. It is proved that optimal Bayesian fusion can be obtained for state-space models with non-interacting states and observations. Simulations are performed to show that MPF with fusion strategy can provide more accurate estimates within fewer time steps comparing to existing algorithms.**

***Index Terms*—Information fusion, Monte Carlo methods, Bayesian inference, Particle filtering**

I. Introduction

Particle filtering (PF) is a Monte Carlo methodology that is used for dynamic state estimation in nonlinear and non-Gaussian probabilistic systems [1], [2]. The methodology is based on a classical statistical technique called importance sampling [3], whereby a set of samples (called particles) and weights are used to form a discrete random measure that approximates the posterior distribution of the unknown states given the observations. While other filtering methodologies that deal with nonlinear systems have been proposed (e.g., extended Kalman filtering (EKF) [4] or unscented Kalman filtering (UKF) [5]), PF methods can handle arbitrary distributional assumptions about the system of interest. This flexibility makes PF powerful for dealing with highly nonlinear and non-Gaussian systems.

Despite the success of PF in many practical applications [6]–[8], the performance of the method heavily depends on a number of factors. For instance, the presence of unknown static parameters in the probabilistic system of interest introduces some difficulties, since the framework for PF methods is traditionally setup for online state estimation [9]. One approach to dealing with this problem is to use Rao-Blackwellization to analytically integrate out the unknown static parameters [10]. Unfortunately, this can only be done for certain mathematical models. In the more general scenario, the typical approach is to augment the static parameter as an unknown state in the probabilistic system. One methodology, for instance, is to model the dynamics of the unknown static parameter as a Gaussian random walk with small noise variance [11]. Another approach uses a kernel-based approach [9], whereby the posterior distribution of the unknown parameters is approximated with a mixture of Gaussians, whose parameters are updated in each time instant. Another methodology, called *density-assisted particle filtering* (DAPF) [12], approximates the joint posterior of the unknown latent states and static parameters using a parameteric density. Special cases of this approach are Gaussian particle filtering (GPF) [13] and Gaussian sum particle filtering (GSPF) [14].

In addition to the challenge of dealing with static parameters, since PF is a Monte Carlo methodology, it also suffers from the *curse of dimensionality*, which causes the performance of the method to deteriorate as the dimension of the unknown state vector increases [15]. One way to alleviate the problems posed by the curse of dimensionality is to consider filtering methodologies which partition the state-space of the unknown states. Multiple particle filtering (MPF) [16], for instance, uses the so-called divide and conquer approach to address the curse of dimensionality. In MPF, the full state vector is partitioned into substates, each dealt with by a separate particle filter. The weighting of the particles is carried out by the exchange of information between filters, which can be done in a variety of ways [17], [18]. It is still unclear how one would also deal with unknown static parameters in the MPF framework, since information about the static parameters may be required across all filters.

In this paper, we propose a novel approach based on Bayesian fusion to tackle the problem of shared unknown static parameters in MPF when the states and observations in the state-space model are non-interacting. We provide numerical simulation results to show that the MPF framework with fusion outperforms the standard particle filtering (SPF) and DAPF especially in high dimensional space. The estimation also converges faster compared to MPF without information fusion.

Authorized licensed use limited to: SUNY AT STONY BROOK. Downloaded on July 05,2025 at 18:36:49 UTC from IEEE Xplore. Restrictions apply.

II. Problem Formulation

Let $\mathbf{x}_t \in \mathbb{R}^{d_x}$ denote a d_x-dimensional latent state vector, $\mathbf{y}_t \in \mathbb{R}^{d_y}$ a d_y-dimensional measurement vector, and t a time-index. We consider a state-space model, where the system evolves according to the following set of equations

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t, \boldsymbol{\theta}), \tag{1}$$

$$\mathbf{y}_t = g(\mathbf{x}_t, \mathbf{v}_t, \boldsymbol{\theta}), \tag{2}$$

where $f(\cdot)$ is the system transition function, $\mathbf{u}_t$ is the state noise that has a distribution $p(\mathbf{u}_t)$, $g(\cdot)$ is the measurement function, $\mathbf{v}_t$ is the measurement noise that has a distribution $p(\mathbf{v}_t)$, and $\boldsymbol{\theta} \in \mathbb{R}^{d_\theta}$ is a d_θ-dimensional unknown static parameter vector that parameterizes both $f(\cdot)$ and $g(\cdot)$.

For a given time t, our goal is to jointly estimate the latent state $\mathbf{x}_t$ and the static parameter vector $\boldsymbol{\theta}$ given the set of measurements $\mathbf{y}_{1:t} \triangleq \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t\}$ under the Bayesian paradigm. This amounts to determining the joint posterior distribution of $\mathbf{x}_t$ and $\boldsymbol{\theta}$

$$p(\mathbf{x}_t, \boldsymbol{\theta}|\mathbf{y}_{1:t}) = p(\mathbf{x}_t|\boldsymbol{\theta}, \mathbf{y}_{1:t})p(\boldsymbol{\theta}|\mathbf{y}_{1:t}). \tag{3}$$

The distribution $p(\mathbf{x}_t|\boldsymbol{\theta}, \mathbf{y}_{1:t})$ represents the filtering distribution under the assumption that the static parameter vector is known, while $p(\boldsymbol{\theta}|\mathbf{y}_{1:t})$ is the posterior distribution of $\boldsymbol{\theta}$.

III. Multiple Particle Filter with Shared Unknown Parameters

In this section, we propose a novel MPF algorithm that fuses information across filters in cases that filters share parameters for systems where the states and observations are non-interacting.

Let the state vector $\mathbf{x}_t$ be partitioned as $\mathbf{x}_t = [\mathbf{x}_{1,t}^\intercal, \mathbf{x}_{2,t}^\intercal, \ldots, \mathbf{x}_{K,t}^\intercal]^\intercal$ and let the unknown static parameter vector $\boldsymbol{\theta}$ be partitioned as $\boldsymbol{\theta} = [\boldsymbol{\theta}_{1,\ell}^\intercal, \boldsymbol{\theta}_{2,\ell}^\intercal, \ldots, \boldsymbol{\theta}_{K,\ell}^\intercal, \boldsymbol{\theta}_g^\intercal]^\intercal$. Since the states and observations are non-interacting, after partitioning, the kth subsystem of the state-space model can be represented via the following set of equations:

$$\mathbf{x}_{k,t} = f_k(\mathbf{x}_{k,t-1}, \mathbf{u}_{k,t}, \boldsymbol{\theta}_{k,\ell}, \boldsymbol{\theta}_g), \quad k = 1, \ldots, K \tag{4}$$

$$\mathbf{y}_{k,t} = g_k(\mathbf{x}_{k,t}, \mathbf{v}_{k,t}, \boldsymbol{\theta}_{k,\ell}, \boldsymbol{\theta}_g), \quad k = 1, \ldots, K, \tag{5}$$

where $f_k(\cdot)$ and $g_k(\cdot)$ are the system transition function and measurement function local to the kth subsystem, $\boldsymbol{\theta}_{k,\ell}$ stands for the local parameter vector in the kth subsystem, and $\boldsymbol{\theta}_g$ is the global parameter vector. In this type of system, the states and observations contained in each subsystem are independent of the states and observations from all other subsystems when the global parameter $\boldsymbol{\theta}_g$ is known.

In MPF, K particle filters are used to jointly estimate the unknowns $\mathbf{x}_t$ and $\boldsymbol{\theta}$, where each filter tracks one of the subsystems above. In this setting, we consider that the kth filter jointly tracks $\mathbf{z}_{k,t} \triangleq \{\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}, \boldsymbol{\theta}_g\}$. At each time instant t, the kth particle filter represents an approximation of the joint posterior distribution $p(\mathbf{z}_{k,t}|\mathbf{y}_{1:t})$ using a set of particles and weights

$$\hat{p}(\mathbf{z}_{k,t}|\mathbf{y}_{1:t}) = \sum_{n=1}^{N_k} w_{k,t}^{(n)} \delta(\mathbf{z}_{k,t}^{(n)} - \mathbf{z}_{k,t}), \tag{6}$$

where $\mathbf{z}_{k,t}^{(n)} \triangleq \{\mathbf{x}_{k,t}^{(n)}, \boldsymbol{\theta}_{k,\ell}^{(n)}, \boldsymbol{\theta}_g^{(n,k)}\}$ is the nth particle of the kth filter. Using the chain rule of probability theory, we can write the posterior distribution $p(\mathbf{z}_{k,t}|\mathbf{y}_{1:t})$ as

$$p(\mathbf{z}_{k,t}|\mathbf{y}_{1:t}) = p(\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}|\boldsymbol{\theta}_g, \mathbf{y}_{1:t})p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}). \tag{7}$$

This essentially means that each filter has an approximation of the marginal posterior of the global parameters $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$. We denote the approximation of $p(\boldsymbol{\theta}|\mathbf{y}_{1:t})$ according to the kth filter as $\hat{p}_k(\boldsymbol{\theta}|\mathbf{y}_{1:t})$ and it is given by

$$\hat{p}_k(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) = \sum_{n=1}^{N_k} w_{k,t}^{(n)} \delta(\boldsymbol{\theta}_g - \boldsymbol{\theta}_g^{(n,k)}). \tag{8}$$

Since each filter has its own approximation to $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$, we would like to fuse these approximations to obtain a better/more accurate distribution $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$. Given the fused distribution, we can then update the kth filter's full posterior accordingly by replacing $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$ in (7) with $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$:

$$q(\mathbf{z}_{k,t}|\mathbf{y}_{1:t}) = p(\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}|\boldsymbol{\theta}_g, \mathbf{y}_{1:t})q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}). \tag{9}$$

The update rule in (9) can be implemented as a resampling step in the MPF framework. This is shown in Algorithm 1.

Algorithm 1 Fusion Strategy for Multiple Particle Filtering

Input: $\{\mathbf{z}_{1,t}^{(n)}, w_{1,t}^{(n)}\}_{n=1}^{N_1}, \ldots, \{\mathbf{z}_{K,t}^{(n)}, w_{K,t}^{(n)}\}_{n=1}^{N_K}$
Output: $\{\tilde{\mathbf{z}}_{1,t}^{(n)}\}_{n=1}^{N_1}, \ldots, \{\tilde{\mathbf{z}}_{K,t}^{(n)}\}_{n=1}^{N_K}$

```
for k = 1, ..., K do
    Approximate p(z_{k,t}|y_{1:t}) with a parametric density p̃(z_{k,t}; β_{k,t}) by estimating the parameters β_{k,t} using the set of samples and weights {z_{k,t}^{(n)}, w_{k,t}^{(n)}}_{n=1}^{N_k}.
end for
Obtain q(θ_g|y_{1:t}) by fusing the marginal distributions by (16)
for k = 1, ..., K do
    Sample N_k particles from the fused marginal distribution by (19)
    for n = 1, ..., N_k do
        Sample a particle from the conditional distribution by (20)
        Set z̃_{k,t}^{(n)} = {x̃_{k,t}^{(n)}, θ̃_{k,ℓ}^{(n)}, θ̃_g^{(n,k)}}.
    end for
end for
```

The idea of Algorithm 1 is to use the set of samples and weights from each of the particle filters and to return to each of those particle filters a set of resampled particles. The resampling of the particles is based on fusing the information from the filters. In order to accomplish this fusion in a principled way, we approximate the current filtering density

Authorized licensed use limited to: SUNY AT STONY BROOK. Downloaded on July 05,2025 at 18:36:49 UTC from IEEE Xplore. Restrictions apply.

of each particle filter using a parametric density

$$\tilde{p}(\mathbf{z}_{k,t};\boldsymbol{\beta}_{k,t}) \approx p(\mathbf{z}_{k,t}|\mathbf{y}_{1:t}), \tag{10}$$

where the parameters $\boldsymbol{\beta}_{k,t}$ are estimated using set of samples and weights from the kth particle filter, $\{\mathbf{z}_{k,t}^{(n)}, w_{k,t}^{(n)}\}_{n=1}^{N_k}$. The idea is to choose the parametric density $\tilde{p}(\mathbf{z}_{k,t};\boldsymbol{\beta}_{k,t})$ from a family of probability distributions such that:

1) The marginal density $\tilde{p}(\boldsymbol{\theta}_g|\boldsymbol{\beta}_{k,t})$ can be obtained analytically.
2) The conditional density $\tilde{p}(\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}|\tilde{\boldsymbol{\theta}}_g;\boldsymbol{\beta}_{k,t})$ can be obtained analytically.

One parametric family of distributions that is known to satisfy the above requirements is the Gaussian distribution. In this case $\boldsymbol{\beta}_{k,t} \triangleq \{\boldsymbol{\mu}_{k,t}, \boldsymbol{\Sigma}_{k,t}\}$, where $\boldsymbol{\mu}_{k,t}$ and $\boldsymbol{\Sigma}_{k,t}$ correspond to the mean and covariance matrix of that Gaussian.

After, estimating the parameters $\boldsymbol{\beta}_{k,t}$, we marginalize the approximate filtering densities in order to obtain K different approximations to the posterior distribution of the global parameters $\boldsymbol{\theta}_g$. The goal is to fuse these K approximate posteriors into a single probability distribution. This fused distribution $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$ is obtained by using a fusion function $h : \mathcal{P}^K \to \mathcal{P}$ [19]:

$$q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) = h\left(\tilde{p}(\boldsymbol{\theta}_g;\boldsymbol{\beta}_{1,t}), \ldots, \tilde{p}(\boldsymbol{\theta}_g;\boldsymbol{\beta}_{K,t})\right), \tag{11}$$

where $\mathcal{P}$ denotes the set of all valid probability density functions for the random vector $\boldsymbol{\theta}_g$.

In terms of choosing the fusion function for the system represented by (4)-(5), the fusion function which allows for optimal Bayesian fusion can be obtained. We refer to optimal Bayesian fusion as a fusion rule that can fuse the local posterior distributions of the global parameter vectors (given their latest local measurement) into the posterior distribution of the global parameter vector given *all measurements* up to current time instant. To get the approximation of global posterior distribution $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$, we combine all the approximations of the local posterior distributions $p(\boldsymbol{\theta}_g|\mathbf{y}_{k,t}, \mathbf{y}_{1:t-1})$ learned from each filter and the global posterior distribution $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})$ learned from last time step.

At a particular time instant t, we can write the marginal posterior distribution of $\boldsymbol{\theta}_g$ given $\mathbf{y}_{1:t}$ as

$$p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\boldsymbol{\theta}_g, \mathbf{y}_{1:t-1})p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1}). \tag{12}$$

In a state-space model described by (4)-(5), the likelihood $p(\mathbf{y}_t|\boldsymbol{\theta}_g, \mathbf{y}_{1:t-1})$ can be proven to be the product of likelihoods of all subsystems

$$p(\mathbf{y}_t|\boldsymbol{\theta}_g, \mathbf{y}_{1:t-1}) = \prod_{k=1}^{K} p(\mathbf{y}_{k,t}|\boldsymbol{\theta}_g, \mathbf{y}_{1:t-1}). \tag{13}$$

Applying Bayes' theorem, we can rewrite the likelihood of kth filter as

$$p(\mathbf{y}_{k,t}|\boldsymbol{\theta}_g, \mathbf{y}_{1:t-1}) \propto \frac{p(\boldsymbol{\theta}_g|\mathbf{y}_{k,t}, \mathbf{y}_{1:t-1})}{p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})}. \tag{14}$$

Thus, we can get the optimal Bayesian fusion

$$p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) \propto \frac{\prod_{k=1}^{K} p(\boldsymbol{\theta}_g|\mathbf{y}_{k,t}, \mathbf{y}_{1:t-1})}{p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})^{K-1}}. \tag{15}$$

The result in (15) is rather intuitive and has a counterpart for static (time-invariant) systems. Here, the posterior distribution $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})$ from last time instant serves as a prior distribution for $\boldsymbol{\theta}_g$ at time instant t. After each filter processes its local measurements to obtain an updated posterior $p(\boldsymbol{\theta}_g|\mathbf{y}_{k,t}, \mathbf{y}_{1:t-1})$ based on the prior $p(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})$. A correction needs to be made since each filter utilized the prior and a simple multiplication of the K posteriors would count the prior multiple times. This correction is made by multiplying the K posteriors and dividing out the prior $K-1$ times. Then the marginal posterior $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$ in (11) can be obtained:

$$q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) = \frac{\prod_{k=1}^{K} \tilde{p}(\boldsymbol{\theta}_g;\boldsymbol{\beta}_{k,t})}{q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t-1})^{K-1}}. \tag{16}$$

When we assume the distributions are all Gaussian distributions, we can get the mean and covariance matrix of the fused distribution $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}) = \mathcal{N}(\boldsymbol{\mu}_{q,t}, \boldsymbol{\Sigma}_{q,t})$:

$$\boldsymbol{\mu}_{q,t} = \boldsymbol{\Sigma}_{q,t}\Big(\sum_{k=1}^{K} \boldsymbol{\Sigma}_{k,t}^{-1}\boldsymbol{\mu}_{k,t} - (K-1)\boldsymbol{\Sigma}_{q,t-1}^{-1}\boldsymbol{\mu}_{q,t-1}\Big), \tag{17}$$

$$\boldsymbol{\Sigma}_{q,t} = \Big(\sum_{k=1}^{K} \boldsymbol{\Sigma}_{k,t}^{-1} - (K-1)\boldsymbol{\Sigma}_{q,t-1}^{-1}\Big)^{-1}. \tag{18}$$

In obtaining the fused marginal posterior $q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t})$, we can now communicate this density to each of the particle filters so that they may use it in their resampling step. For each filter, a set of particles is drawn from the fused marginal posterior

$$\tilde{\boldsymbol{\theta}}_g^{(n,k)} \sim q(\boldsymbol{\theta}_g|\mathbf{y}_{1:t}), \quad n = 1, \ldots, N_k. \tag{19}$$

We then use the conditional distribution $\tilde{p}(\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}|\tilde{\boldsymbol{\theta}}_g;\boldsymbol{\beta}_{k,t})$ to sample corresponding substates and local parameters:

$$\left\{\tilde{\mathbf{x}}_{k,t}^{(n)}, \tilde{\boldsymbol{\theta}}_{k,\ell}^{(n)}\right\} \sim \tilde{p}(\mathbf{x}_{k,t}, \boldsymbol{\theta}_{k,\ell}|\tilde{\boldsymbol{\theta}}_g^{(n,k)};\boldsymbol{\beta}_{k,t}), \quad n = 1, \ldots, N_k. \tag{20}$$

Using this strategy, each particle filter obtains a set of resampled particles $\{\tilde{\mathbf{z}}_{k,t}^{(n)}\}_{n=1}^{N_k}$, where $\tilde{\mathbf{z}}_{k,t}^{(n)} \triangleq \{\tilde{\mathbf{x}}_{k,t}^{(n)}, \tilde{\boldsymbol{\theta}}_{k,\ell}^{(n)}, \tilde{\boldsymbol{\theta}}_g^{(n,k)}\}$.

IV. Simulation Results

In this section, we present numerical results with different scaled systems. Four different algorithms are tested: SPF with a Gaussian random walk model [11]; DAPF [12]; MPF-1, where each partition performs as a SPF and the global estimate is given by the mean of estimates from each partition; MPF-2, which is the proposed method in this work.

For simplicity, we assume that there are no local unknown parameters. Specifically, we consider a system which can be described by the following transition function and measure-

Authorized licensed use limited to: SUNY AT STONY BROOK. Downloaded on July 05,2025 at 18:36:49 UTC from IEEE Xplore. Restrictions apply.

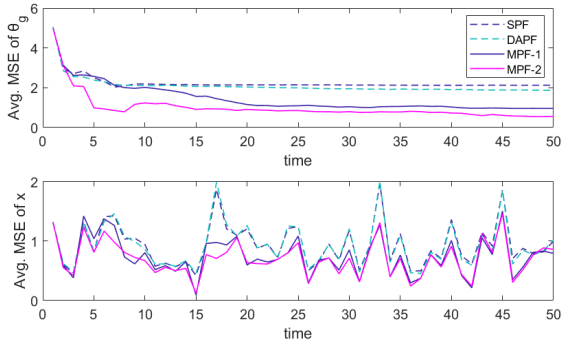


Fig. 1: Averaged MSE of parameter and state estimates with $d_{\mathbf{x}} = 10$ and $d_{\boldsymbol{\theta}_g} = 2$.

ment function:

$$x_{i,t} = \frac{\theta_1}{1 + e^{-x_{i,t-1}+\theta_5}} + \theta_2 + u_{i,t}, \qquad i = 1, \ldots, d_{\mathbf{x}} \quad (21)$$

$$y_{i,t} = \theta_3 x_{i,t} + \theta_4 + v_{i,t}, \qquad i = 1, \ldots, d_{\mathbf{x}} \quad (22)$$

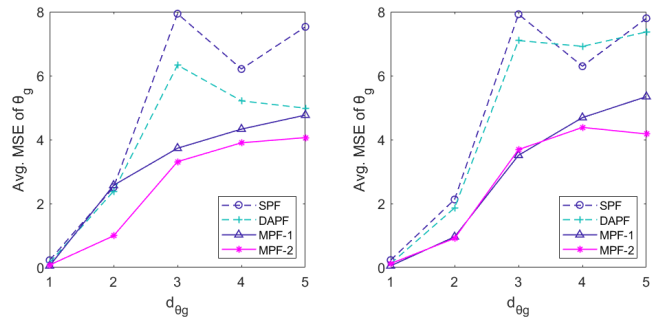
where $d_{\mathbf{x}} = 10$, $u_{i,t} \sim \mathcal{N}(0, \sigma^2_{u_i})$ and $v_{i,t} \sim \mathcal{N}(0, \sigma^2_{v_i})$ are independent and identically distributed zero-mean Gaussian noise with variance $\sigma^2_{u_i} = 2$ and $\sigma^2_{v_i} = 1$, respectively. $\boldsymbol{\theta}_g = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^\intercal = [2, -2, 2, -2, 3]^\intercal$ is a global parameter vector. We test all algorithms with different $d_{\boldsymbol{\theta}_g}$, where the first $d_{\boldsymbol{\theta}_g}$ parameters are assumed to be unknown parameters to estimate, while the rest are known. For further clarification, when $d_{\boldsymbol{\theta}_g} = j$, we assume that only $\theta_1, \ldots, \theta_j$ is an unknown parameter that will be estimated and $\theta_{j+1}, \ldots, \theta_5$ are known.

The number of particles used in the simulation is $100 d_{\boldsymbol{\theta}_g}$ per state dimension. In other words, when $d_{\boldsymbol{\theta}_g} = 2$, we estimate 2 unknown parameters. The number of particles in SPF and DAPF are both $N = 2000$. In MPF, with the number of partitions being $K = 5$, each particle filter utilizes $N_k = 400$ particles. The initial particles are drawn as follows: $\mathbf{x}_0^{(n)} \sim \mathcal{N}(\mathbf{0}, \sigma^2_{u_i} \boldsymbol{I})$, $\boldsymbol{\theta}_g \sim \mathcal{N}(\mathbf{1}, 2\boldsymbol{I})$.

Fig. 1 shows the average mean square errors (MSEs) of the unknown static parameters and the latent states calculated from 100 realizations of each algorithm with $d_{\boldsymbol{\theta}_g} = 2$. In each realization, the system iterates $T = 50$ times. It is obvious that MPF methods outperform the single particle filter methods (SPF and DAPF) and give lower MSEs on both state and parameter estimation. Though the two MPF methods seem to land to a comparable result after 50 iterations, the MPF-2 in this work coverges faster than the MPF-1 which fuses only the final estimates by taking the mean.

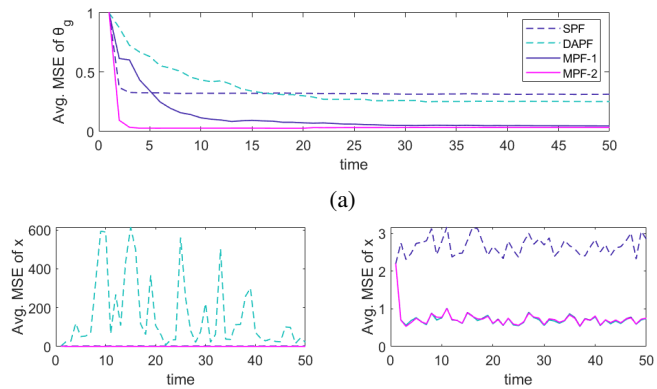
The effect of $d_{\boldsymbol{\theta}_g}$ can be found in Fig. 2. Since $\boldsymbol{\theta}_g$ cannot be partitioned in any of these methods, when the dimensions of unknown static parameters grow larger, all methods will perform worse and provide higher MSEs. However, with the fusion strategy, the performance of our work degrades less than other methods. Specifically, our work can provide a satisfying estimate within only a few time steps as shown in Figure 2a.

The system tested above is a relatively smaller system with $d_{\mathbf{x}} = 10$, which PFs can still handle. However, due to the *curse of dimensionality*, PFs cannot even provide reasonable



(a) $t = 5$ (b) $t = 50$

Fig. 2: Averaged MSE of the static parameter estimates under different $d_{\boldsymbol{\theta}_g}$ and t with $d_{\mathbf{x}} = 10$.



(a) (b) (c)

Fig. 3: Averaged MSE of parameter and state estimates with $d_{\mathbf{x}} = 100$ and $d_{\boldsymbol{\theta}_g} = 1$. Figure (c) shows the invisible MSEs in Figure (b) due to the scale.

estimates when the scale of the system becomes large. We provide the following results for the same system with $d_{\mathbf{x}} = 100$ and $d_{\boldsymbol{\theta}_g} = 1$. The number of particles per dimension remains the same. With more state dimensions, we choose $K = 50$. Still, 100 realizations are run and the MSEs are recorded in Fig. 3. In such large system, the SPF and DAPF are not able to estimate the latent states properly.

V. Conclusion

In this paper we propose a novel fusion strategy used in MPF framework. It allows for the sharing of unknown static parameters for state-space models that can be partitioned into independent subsystems. We prove that optimal Bayesian fusion can be obtained in this case and the information fusion can be implemented as a resampling step in MPF. Simulations are run to show that the algorithm give better estimation for both latent states and unknown parameters comparing to the SPF, DAPF, and MPF without fusion strategy. Since the fusion function is only given for independent subsystems, it remains as future work to derive the approximation of best fusion results in models where the states and observations are interacting.

Authorized licensed use limited to: SUNY AT STONY BROOK. Downloaded on July 05,2025 at 18:36:49 UTC from IEEE Xplore. Restrictions apply.

REFERENCES

[1] A. Doucet, N. De Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*. Springer, 2001, pp. 3–14.

[2] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[3] C. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer Science & Business Media, 2013.

[4] G. Welch, "An introduction to the Kalman filter," 1995.

[5] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. Ieee, 2000, pp. 153–158.

[6] P. M. Djuric, M. Vemula, and M. F. Bugallo, "Target tracking by particle filtering in binary sensor networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2229–2238, 2008.

[7] K. Nakamura, R. Yoshida, M. Nagasaki, S. Miyano, and T. Higuchi, "Parameter estimation of in silico biological pathways with particle filtering towards a petascale computing," in *Biocomputing 2009*. World Scientific, 2009, pp. 227–238.

[8] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[9] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," in *Sequential Monte Carlo Methods in Practice*. Springer, 2001, pp. 197–223.

[10] P. Li, R. Goodall, and V. Kadirkamanathan, "Estimation of parameters in a linear state space model using a Rao-Blackwellised particle filter," *IEE Proceedings-Control Theory and Applications*, vol. 151, no. 6, pp. 727–738, 2004.

[11] G. Kitagawa, "A self-organizing state-space model," *Journal of the American Statistical Association*, pp. 1203–1215, 1998.

[12] P. M. Djuric, M. F. Bugallo, and J. Míguez, "Density assisted particle filters for state and parameter estimation," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2. IEEE, 2004, pp. ii–701.

[13] J. Kotecha and P. Djuric, "Gaussian particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2592–2601, 2003.

[14] ——, "Gaussian sum particle filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2602–2612, 2003.

[15] T. Bengtsson, P. Bickel, B. Li *et al.*, "Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems," in *Probability and Statistics: Essays in Honor of David A. Freedman*. Institute of Mathematical Statistics, 2008, pp. 316–334.

[16] P. M. Djuric, T. Lu, and M. F. Bugallo, "Multiple particle filtering," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 3. IEEE, 2007, pp. III–1181.

[17] P. M. Djurić and M. F. Bugallo, "Particle filtering for high-dimensional systems," in *2013 5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE, 2013, pp. 352–355.

[18] B. Ait-El-Fquih and I. Hoteit, "A variational Bayesian multiple particle filtering scheme for large-dimensional systems," *IEEE Transactions on Signal Processing*, vol. 64, no. 20, pp. 5409–5422, 2016.

[19] G. Koliander, Y. El-Laham, P. M. Djurić, and F. Hlawatsch, "Fusion of probability density functions," *Proceedings of the IEEE*, vol. 110, no. 4, pp. 404–453, 2022.

Authorized licensed use limited to: SUNY AT STONY BROOK. Downloaded on July 05,2025 at 18:36:49 UTC from IEEE Xplore. Restrictions apply.