SEQUENTIAL MCMC WITH INVERTIBLE PARTICLE FLOW

Yunpeng Li and Mark Coates

Dept. of Electrical and Computer Engineering, McGill University, Montréal, Québec, Canada

ABSTRACT

Particle filters are among the most effective filtering algorithms for nonlinear and non-Gaussian models. When the state dimension is high, they are known to suffer from weight degeneracy. Sequential Markov chain Monte Carlo (SMCMC) methods have been proposed as an alternative sequential inference technique that can perform better in high dimensional state spaces. In this paper, we propose to construct a composite Metropolis-Hastings (MH) kernel within the SMCMC framework using invertible particle flow. Simulation results show that the proposed kernel significantly increases the acceptance rate and improves estimation accuracy compared with state-of-the-art filtering algorithms, in high dimensional simulation examples.

Index Terms— Markov chain Monte Carlo, particle flow, high dimensional filtering, Bayesian filtering

1. INTRODUCTION

Learning from sequential observations is an important task in various fields. In applications including multi-target tracking and weather forecasting, the underlying models usually have high dimensions. Particle filters [1, 2], which are the standard tools for sequential inference in nonlinear non-Gaussian models, often suffer from the so-called weight degeneracy issue in these high dimensional scenarios [3, 4].

Advanced particle filtering methods have been proposed to combat weight degeneracy, including the auxiliary particle filter [5] and Rao-Blackwellised particle filter [6]. Recently, particle flow methods have been proposed as a promising avenue for high dimensional filtering. These algorithms migrate particles continuously from the prior distribution to the posterior distribution [7–9]. By eliminating the importance sampling step, the weight degeneracy issue is avoided. The filters are not statistically consistent due to approximation errors introduced during implementation.

An alternative method is to use the particle flow to construct a proposal distribution that is close to the posterior distribution [10–13]. With this approach, the statistical consistency of particle filters is preserved. Most algorithms of this type are computationally expensive due to complex importance weight evaluation. One exception is the particle flow particle filter (PF-PF) [13], which constructs invertible particle flows to allow efficient importance weight calculations. It exhibits excellent performance when measurements are highly informative. However, when the state dimension is high, sampling errors from importance sampling may become more significant than approximation errors of particle flow.

Another direction in high dimensional filtering is to incorporate Markov chain Monte Carlo (MCMC) methods into particle filters. Gilks et al. proposed to use MCMC moves after the resampling step of the particle filter to diversify particles in a sequential inference setup [14]. However, in high dimensional filtering, it is often the case that very few duplicated particles remain after resampling. Thus, many MCMC moves may be needed for good mixing. A more general framework called sequential Markov chain Monte Carlo (SMCMC) [15–18] avoids resampling by sampling directly from the target distribution using rejection sampling. One popular class of SMCMC algorithms consists of three steps at time-step k. First, a joint draw step is used to sample from the joint posterior distribution and update all states $x_{1:k}$ up to the current time step. This is followed by two refinement steps. In the first of these the past states $x_{1:k-1}$ are updated; subsequently the current state x_k is updated.

Various MCMC kernels developed for sampling in high dimensional spaces can be used inside the SMCMC framework. Among them, the SmHMC algorithm [18], which uses the manifold Hamiltonian Monte Carlo (mHMC) kernel [19] to refine of the current state, has been shown to be one of the most effective algorithms in high dimensional spaces.

In this paper, we propose to incorporate the invertible particle flow proposed in [13] into the joint draw step of the SMCMC framework. The goal is to better approximate the optimal Metropolis-Hastings (MH) kernel in order to increase the acceptance rate of the joint draw stage. The proposed composite MH kernel incorporates the ability of the particle flow to migrate particles into high posterior density regions and exploits the power of mHMC methods to efficiently explore the state space. Numerical simulations show that the proposed composite MH kernel significantly increases the acceptance rate of the joint draw and reduces the estimation error.

The rest of the paper is organized as follows. Section 2 provides the problem statement and Section 3 discusses related work. We describe the proposed method in Section 4. The simulation results are presented in Section 5. Concluding remarks are provided in Section 6.

978-1-5090-4117-6/17/$31.00 ©2017 IEEE

2. SIMULATION MODEL

We consider the nonlinear filtering task with the following models:

$$x_k = g_k(x_{k-1}, v_k) \tag{1}$$

$$z_k = h_k(x_k, w_k)\,. \tag{2}$$

$g_k : \mathbb{R}^d \times \mathbb{R}^{d'} \to \mathbb{R}^d$ specifies the dynamic model of the state $x_k \in \mathbb{R}^d$ at time step k and $v_k \in \mathbb{R}^{d'}$ is the dynamic noise. The measurement $z_k \in \mathbb{R}^S$ is described by the measurement model $h_k : \mathbb{R}^d \times \mathbb{R}^{S'} \to \mathbb{R}^S$ and $w_k \in \mathbb{R}^{S'}$ is the measurement noise. $h_k(x_k, 0)$ is a C^1 function, i.e. $h_k(x_k, 0)$ is a differentiable function whose derivatives are smooth. We use $x_{a:b}$ to denote the set $\{x_a, x_{a+1}, \ldots, x_b\}$ and $z_{a:b}$ to denote the set $\{z_a, z_{a+1}, \ldots, z_b\}$, where a and b are integers and $a < b$.

3. RELATED WORK

3.1. Sequential Markov chain Monte Carlo methods

A unifying framework of the Sequential Markov chain Monte Carlo (SMCMC) methods has been provided in [18]. Instead of performing the joint draw of $p(x_{1:k}|z_{1:k})$ whose dimension is increasing over time, we adopt the strategy in [16] in which the target joint distribution is $p(x_k, x_{k-1}|z_{1:k})$. The purpose of the joint draw is to avoid numerical integration of the predictive density when the target distribution is $p(x_k|z_{1:k})$ (see [16] for more details). Individual refinement steps that update x_{k-1} and x_k individually follow the joint draw step. The general form of the composite MH kernel in the SMCMC framework is summarized in Algorithm 1. Different choices of the MCMC kernel for high dimensional SMCMC are discussed in [18].

Algorithm 1: Composite MH Kernels in a unifying framework of SMCMC [16, 18].
Input: $x_k^{j-1}, \{x_{k-1}^s\}_{s=N_b+1}^{N_b+N_p}$.
Output: x_k^j.

```
Joint draw:
Propose $\{x_k^*, x_{k-1}^*\} \sim q_1(x_k, x_{k-1}|x_k^{j-1}, x_{k-1}^{j-1})$;
Compute the MH acceptance probability $\rho_1 =$
    $\min\big(1, \frac{p(x_k^*, x_{k-1}^*|z_{1:k})}{q_1(x_k^*, x_{k-1}^*|x_k^{j-1}, x_{k-1}^{j-1})} \frac{q_1(x_k^{j-1}, x_{k-1}^{j-1}|x_k^*, x_{k-1}^*)}{p(x_k^{j-1}, x_{k-1}^{j-1}|z_{1:k})}\big)$;
Accept $x_k^j = x_k^*$, and $x_{k-1}^j = x_{k-1}^*$ with probability $\rho_1$
    otherwise set $x_k^j = x_k^{j-1}$ and $x_{k-1}^j = x_{k-1}^{j-1}$;
Individual refinement of $x_{k-1}^j$:
Propose $x_{k-1}^* \sim q_2(x_{k-1}|x_k^j, x_{k-1}^j)$;
Compute the MH acceptance probability
    $\rho_2 = \min\big(1, \frac{p(x_{k-1}^*|x_k^j, z_{1:k})}{q_2(x_{k-1}^*|x_k^j, x_{k-1}^j)} \frac{q_2(x_{k-1}^j|x_k^j, x_{k-1}^*)}{p(x_{k-1}^j|x_k^j, z_{1:k})}\big)$;
Accept $x_{k-1}^j = x_{k-1}^*$ with probability $\rho_2$;
Individual refinement of $x_k^j$:
Propose $x_k^* \sim q_3(x_k|x_k^j, x_{k-1}^j)$;
Compute the MH acceptance probability
    $\rho_3 = \min\big(1, \frac{p(x_k^*|x_{k-1}^j, z_{1:k})}{q_3(x_k^*|x_k^j, x_{k-1}^j)} \frac{q_3(x_k^j|x_k^*, x_{k-1}^j)}{p(x_k^j|x_{k-1}^j, z_{1:k})}\big)$;
Accept $x_k^j = x_k^*$ with probability $\rho_3$;
```

3.2. Particle flow particle filters

The particle flow particle filters (PF-PF) proposed in [13] construct invertible particle flows in a pseudo-time interval $\lambda \in [0, 1]$, in order to move particles drawn from the prior distribution into regions where the posterior density is high.

Suppose that N_p particles $\{x_{k-1}^j\}_{j=1}^{N_p}$ approximate the posterior distribution at time step $k-1$. At time step k, the dynamic model is first used to generate two sets of particles: $\eta_0^j = g_k(x_{k-1}^j, v_k)$ and $\bar{\eta}_0^j = g_k(x_{k-1}^j, 0)$ for $j = 1, \ldots, N_p$. $\{\eta_0^j\}_{j=1}^{N_p}$ are distributed according to the prior distribution. The trajectory of the j-th particle η_λ^j in $\lambda \in [0, 1]$ is defined by an ordinary differential equation (ODE):

$$\frac{d\eta_\lambda^j}{d\lambda} = \zeta(\eta_\lambda^j, \lambda)\,, \tag{3}$$

where $\zeta(\eta_\lambda^j, \lambda)$ is a deterministic drift term. In the localized exact Daum Huang (LEDH) version of the PF-PF,

$$\zeta(\eta_\lambda^j, \lambda) = A^j(\lambda)\eta_\lambda^j + b^j(\lambda)\,, \tag{4}$$

where both $A^j(\lambda)$ and $b^j(\lambda)$ admit analytic expressions and are derived using $\bar{\eta}_\lambda^j$, which is generated and propagated independent of the sampling noise in the prior propagation step [13]. Introducing $\bar{\eta}_\lambda^j$ allows the flow of η_λ^j to constitute an invertible mapping between η_0^j and η_1^j, and this in turn enables efficient calculations of the proposal densities.

The solutions to the ODEs are approximated using discretized pseudo-time integration at N_λ positions $[\lambda_1, \lambda_2, \ldots, \lambda_{N_\lambda}]$, with a sequence of discrete steps $\{\epsilon_m\}_{m=1}^{N_\lambda}$ where $\lambda_m = \sum_{s=1}^{m} \epsilon_s$, $\epsilon_m > 0$ for $m \in \{1, \ldots, N_\lambda\}$, and $\sum_{m=1}^{N_\lambda} \epsilon_m = 1$. Applying Euler's method at the pseudo-time λ_{m-1}, we get

$$\begin{aligned} \eta_{\lambda_m}^j &= \eta_{\lambda_{m-1}}^j + \epsilon_m (A^j(\lambda_m)\eta_{\lambda_{m-1}}^j + b^j(\lambda_m)) \\ &= (I + \epsilon_m A^j(\lambda_m))\eta_{\lambda_{m-1}}^j + \epsilon_m b^j(\lambda_m) \end{aligned} \tag{5}$$

From Equation (5), we can derive that

$$\eta_1^j = C^j \eta_0^j + D^j\,, \tag{6}$$

where

$$C^j = \prod_{m=1}^{N_\lambda} (I + \epsilon_{N_\lambda+1-m} A^j(\lambda_{N_\lambda+1-m})) \tag{7}$$

$$\begin{aligned} D^j = &\sum_{m=1}^{N_\lambda-1} ([\prod_{m=1}^{N_\lambda-m} (I + \epsilon_{N_\lambda+1-m} A^j(\lambda_{N_\lambda+1-m}))]\epsilon_m b^j(\lambda_m)) \\ &+ \epsilon_{N_\lambda} b^j(\lambda_{N_\lambda})\,. \end{aligned} \tag{8}$$

In [13], we prove that the mapping between η_0^j and η_1^j is invertible with sufficiently small ϵ_m. Thus, C^j is invertible and $\det(C^j) \neq 0$. The invertible particle flow mapping is summarized in Algorithm 2 and the proposal density is

$$q(\eta_1^j|x_{k-1}^j, z_k) = \frac{p(\eta_0^j|x_{k-1}^j, z_k)}{|\det(C^j)|} = \frac{p(\eta_0^j|x_{k-1}^j)}{|\det(C^j)|}\,. \tag{9}$$

Algorithm 2: Function$(\eta_1^j, C^j, D^j) = T(\eta_0^j, \bar{\eta}_0^j)$

```
Set $\lambda = 0, \bar{\eta} = \bar{\eta}_0^j$;
for $m = 1, \dots, N_\lambda$ do
    Set $\lambda = \lambda + \epsilon_j$;
    Calculate $A_m^j(\lambda)$ and $b_m^j(\lambda)$ with the linearization being performed at $\bar{\eta}$;
    Migrate $\bar{\eta}$: $\bar{\eta} = \bar{\eta} + \epsilon_m(A_m^j(\lambda)\bar{\eta} + b_m^j(\lambda))$;
end for
Calculate $C^j = \prod_{m=1}^{N_\lambda}(I + \epsilon_{N_\lambda+1-m}A_{N_\lambda+1-m}^j(\lambda))$;
Calculate $D^j = \sum_{m=1}^{N_\lambda-1}([\prod_{m=1}^{N_\lambda-m}(I + \epsilon_{N_\lambda+1-m}A_{N_\lambda+1-m}^j(\lambda))]\epsilon_m b_m^j(\lambda)) + \epsilon_{N_\lambda}b_{N_\lambda}^j(\lambda)$;
Migrate particles: $\eta_1^j = C^j\eta_0^j + D^j$;
```

4. SMHMC WITH INVERTIBLE PARTICLE FLOW

The SMCMC framework allows for various choices of MCMC kernels for the joint draw, as discussed in [18]. But many choices are difficult to construct, especially in complex and high dimensional problems. The simplest choice is the *independent MH kernel based on prior as proposal* summarized in [18], which constructs $q_1()$ based on the empirical marginal posterior distribution at the previous time step and the dynamic model. This strategy is adopted in the implementation of SmHMC in [18]. However, the acceptance rate ρ_1 with this kernel can be very low in high dimensional systems.

Another choice of $q_1()$ is the *approximation of the optimal independent MH kernel* discussed in [18]. It is noted in [18] that this kernel is difficult to construct in complex and high dimensional problems. Here, we propose to use invertible particle flow to approximate the optimal independent MH kernel. The proposed composite MH kernel which combines SmHMC and invertible particle flow is presented in Algorithm 3. N_b is the burn-in period and N_p is the number of retained MCMC samples.

x_{k-1}^* is drawn uniformly at random from a set of equally weighted particles at time $k-1$ denoted by $\sum_{s=N_b+1}^{N_p+N_b}\delta_{x_{k-1}^s}(dx_{k-1})$. Thus,

$$p(x_{k-1}^*|x_k^{j-1}, x_{k-1}^{j-1}, z_k) = \frac{1}{N_p}\,. \tag{10}$$

The acceptance rate of the joint draw can then be derived as follows:

$$\begin{aligned}\rho_1 &= \min\big(1, \frac{p(x_k^*, x_{k-1}^*|z_{1:k})p(x_k^{j-1}, x_{k-1}^{j-1}|x_k^*, x_{k-1}^*, z_k)}{p(x_k^*, x_{k-1}^*|x_k^{j-1}, x_{k-1}^{j-1}, z_k)p(x_k^{j-1}, x_{k-1}^{j-1}|z_{1:k})}\\ &= \min\big(1, \frac{p(x_k^*|x_{k-1}^*)p(z_k|x_k^*)|\det(C^*)|p(\eta_0^{j-1}|x_{k-1}^{j-1})}{p(\eta_0^*|x_{k-1}^*)p(x_k^{j-1}|x_{k-1}^{j-1})p(z_k|x_k^{j-1})|\det(C^{j-1})|}\big)\,.\end{aligned} \tag{11}$$

When evaluating Equation (11) in Line 4 of Algorithm 3, the value of x_k^{j-1} is needed. Since x_k^{j-1} may be generated by the manifold Hamiltonian Monte Carlo kernel $q_3()$, the corresponding η_0^j is not available through Line 2 and 5 of Algorithm 3. This can be resolved using the invertible mapping property of the invertible particle flow. As C^j is invertible, we can calculate η_0^j given x_k^j by solving Equation (6):

$$\eta_0^j = (C^j)^{-1}(x_k^j - D^j)\,. \tag{12}$$

Algorithm 3: Composite MH Kernels constructed with the manifold Hamiltonian Monte Carlo kernel and invertible particle flow.
Input: $x_k^{j-1}, \eta_0^{j-1}, C^{j-1}, \{x_{k-1}^s\}_{s=N_b+1}^{N_b+N_p}$.
Output: x_k^j, η_0^j, C^j.

```
    Joint draw:
 1: Draw $x_{k-1}^*$ uniformly from $\sum_{s=N_b+1}^{N_p+N_b}\delta_{x_{k-1}^s}(dx_{k-1})$;
 2: Sample $\eta_0^* = g_k(x_{k-1}^*, v_k)$,
    calculate $\bar{\eta}_0^* = g_k(x_{k-1}^*, 0)$;
 3: Perform invertible particle flow (Algorithm 2)
    $(x_k^*, C^*, D^*) = T(\eta_0^*, \bar{\eta}_0^*)$;
 4: Compute the MH acceptance probability $\rho_1 =$
    $\min\big(1, \frac{p(x_k^*|x_{k-1}^*)p(z_k|x_k^*)|\det(C^*)|p(\eta_0^{j-1}|x_{k-1}^{j-1})}{p(\eta_0^*|x_{k-1}^*)p(x_k^{j-1}|x_{k-1}^{j-1})p(z_k|x_k^{j-1})|\det(C^{j-1})|}\big)$;
 5: Accept $x_k^j = x_k^*$, $x_{k-1}^j = x_{k-1}^*$, $\eta_0^j = \eta_0^*$, $C^j = C^*$
    and $D^j = D^*$ with probability $\rho_1$.
    Otherwise set $x_k^j = x_k^{j-1}$, $x_{k-1}^j = x_{k-1}^{j-1}$, $\eta_0^j = \eta_0^{j-1}$,
    $C^j = C^{j-1}$ and $D^j = D^{j-1}$;
    Individual refinement of $x_{k-1}^j$:
 6: Draw $x_{k-1}^*$ uniformly from $\sum_{s=N_b+1}^{N_p+N_b}\delta_{x_{k-1}^s}(dx_{k-1})$;
 7: Compute the MH acceptance probability
    $\rho_2 = \min\big(1, \frac{p(x_k^j|x_{k-1}^*)}{p(x_k^j|x_{k-1}^j)}\big)$;
 8: Accept $x_{k-1}^j = x_{k-1}^*$ with probability $\rho_2$;
    Individual refinement of $x_k^j$:
 9: Propose $x_k^* \sim q_3(x_k|x_k^j, x_{k-1}^j)$ using the manifold
    Hamiltonian MCMC kernel;
10: Compute the MH acceptance probability
    $\rho_3 = \min\big(1, \frac{p(x_k^*|x_{k-1}^j, z_{1:k})}{q_3(x_k^*|x_k^j, x_{k-1}^j)}\frac{q_3(x_k^j|x_k^*, x_{k-1}^j)}{p(x_k^j|x_{k-1}^j, z_{1:k})}\big)$;
11: Accept $x_k^j = x_k^*$ with probability $\rho_3$;
12: Calculate $\eta_0^j = (C^j)^{-1}(x_k^j - D^j)$;
```

5. SIMULATION AND RESULTS

5.1. Simulation setup

The SmHMC algorithm exhibits the smallest average mean squared error among a variety of SMCMC algorithms in the large sensor field simulation setup used in [18]. Thus, we would like to compare the proposed composite MH kernel with SmHMC in the same simulation examples.

The setup includes d sensors evenly deployed on a two-dimensional grid $\{1, 2, \dots, \sqrt{d}\} \times \{1, 2, \dots, \sqrt{d}\}$. The measurement vector $z_k = [z_k^1, z_k^2, \dots, z_k^d]$ contains measurements of the state at each sensor's location. The state vector $x_k = [x_k^1, x_k^2, \dots, x_k^d]$ evolves according to the multivariate generalized hyperbolic skewed-t distribution:

$$p(x_k|x_{k-1}) = \frac{e^{(x_k - \alpha x_{k-1})^T \Sigma^{-1} \gamma}}{\sqrt{(\nu + Q(x_k))(\gamma^T \Sigma^{-1} \gamma)}^{-\frac{\nu+d}{2}} (1 + \frac{Q(x_k)}{\nu})^{\frac{\nu+d}{2}}} \times K_{\frac{\nu+d}{2}}(\sqrt{(\nu + Q(x_k))(\gamma^T \Sigma^{-1} \gamma)}) \qquad (13)$$

where γ and ν determine the shape of the distribution, $K_{\frac{\nu+d}{2}}$ is the modified Bessel function of the second kind of order $\frac{\nu+d}{2}$, and $Q(x_k) = (x_k - \alpha x_{k-1})^T \Sigma^{-1} (x_k - \alpha x_{k-1})$. The (i, j)-th entry of the dispersion matrix Σ is:

$$\Sigma_{i,j} = \alpha_0 e^{-\frac{||R^i - R^j||_2^2}{\beta}} + \alpha_1 \delta_{i,j} \qquad (14)$$

where $|| \cdot ||_2$ is the L2-norm, R^i is the physical location of the i-th sensor, and $\delta_{i,j}$ is the Kronecker symbol. The measurements are count data which are distributed according to:

$$p(z_k|x_k) = \prod_{c=1}^{d} \mathcal{P}(z_k^c; m_1 e^{m_2 x_k^c}), \qquad (15)$$

where $\mathcal{P}(\cdot; \Lambda)$ is the Poisson(Λ) distribution. m_1 and m_2 are scalars which control the mean of the Poisson distribution.

5.2. Parameter values

Parameter values are set according to [18]: $\alpha = 0.9, \alpha_0 = 3, \alpha_1 = 0.01, \beta = 20, \nu = 7$. All elements of the vector γ are set to 0.3. True states start with $x_0^c = 0$, for $c = 1, \dots, d$. For the measurement model, $m_1 = 1$ and $m_2 = \frac{1}{3}$. We evaluate two scenarios with $d = 144$ or 400. Each scenario is repeated 100 times while each simulation lasts for 10 time steps.

We compare the proposed SmHMC with particle flow (SmHMC+flow) algorithm with SmHMC [18], the exact Daum and Huang (EDH) filter [7], the PF-PF based on EDH [13], and the bootstrap particle filter (BPF) [1]. All filters are initialized with the same true state. Following the practice in [13, 18], the burn-in period N_b is set to 20 for SmHMC and SmHMC+flow. The number of retained particles N_p is set to be 200 for SmHMC and SmHMC+flow. EDH also uses 200 particles. The number of particles for BPF and PF-PF (EDH) is 10000, as particle filters suffer from severe weight degeneracy in such high dimensional scenarios.

5.3. Results

We report average estimation errors, acceptance rates (if applicable) and execution time per step in Table 1.

Table 1. Average performance of different filters based on 100 simulation trials. Results are produced with an Intel i7-4770K 3.50GHz CPU and 32GB RAM.

d	Algorithm	# particle	MSE	Acceptance rate			Time (s)
				ρ_1	ρ_2	ρ_3	
144	SmHMC + flow	200	0.68	0.55	0.01	0.68	41
	SmHMC	200	0.82	0.003	0.01	0.73	16
	EDH	200	0.69	-	-	-	0.050
	PF-PF (EDH)	10000	0.82	-	-	-	1.6
	BPF	10000	2.28	-	-	-	0.073
400	SmHMC + flow	200	0.59	0.48	0.02	0.52	430
	SmHMC	200	0.73	0.002	0.02	0.63	90
	EDH	200	0.60	-	-	-	0.55
	PF-PF (EDH)	10000	0.89	-	-	-	4.6
	BPF	10000	4.98	-	-	-	0.24

The proposed SmHMC + flow method leads to the smallest average MSE in both simulation examples ($d = 144$ or $d = 400$). Compared with SmHMC, we notice a significant increase of ρ_1, which is the joint draw acceptance rate. The acceptance rate of SmHMC is so low that the refinement step is responsible for generating almost all of the particles. By increasing the joint draw acceptance, SmHMC+flow applies the mHMC sampling procedure from many more initializations, leading to a better exploration of the state space and a reduction in estimation error. This comes with a higher computational cost due to the relatively expensive flow operation.

The SmHMC+flow method achieves estimation accuracy very similar to that of the EDH filter. Although the EDH filter is computationally much more efficient, it is not statistically consistent. The PF-PF (EDH) suffers from weight degeneracy because it performs importance sampling (and resampling) in such high dimensional spaces. The BPF suffers from severe weight degeneracy and has the highest estimation error.

6. CONCLUSION

In this paper, we propose the use of invertible particle flow to construct proposal distributions in the joint draw step of the SMCMC algorithm. The proposed composite MH kernel combines the desirable attributes of both of its constituents. By incorporating particle flow, it can create initial particles in regions of higher posterior density. The Hamiltonian Monte Carlo process then allows it to efficiently explore the state space in the neighbourhood of these initial particles. In the examined simulation setups with 144 or 400 states, where the SmHMC algorithm exhibited the smallest average MSE in [18], the proposed SmHMC+invertible particle flow algorithm leads to significantly smaller MSEs and much higher acceptance rates in the joint draw stage.

7. REFERENCES

[1] N.J. Gordon, D.J. Salmond, and A.F.M. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” in *IEE Proc. F Radar and Signal Processing*, Apr. 1993, vol. 140, pp. 107–113.

[2] A. Doucet and A. M Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” in *The Oxford Handbook of Nonlinear Filtering*, chapter 24, pp. 656–704. Oxford University Press, Oxford, UK, 2009.

[3] T. Bengtsson, P. Bickel, and B. Li, “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*, D. Nolan and T. Speed, Eds., vol. 2, pp. 316–334. Institute of Mathematical Statistics, Beachwood, OH, 2008.

[4] A. Beskos, D. Crisan, and A. Jasra, “On the stability of sequential Monte Carlo methods in high dimensions,” *Ann. Appl. Prob.*, vol. 24, no. 4, pp. 1396–1445, 2014.

[5] M.K. Pitt and N. Shephard, “Filtering via simulation: Auxiliary particle filters,” *J. Am. Statist. Assoc.*, vol. 94, no. 446, pp. 590–599, Jun. 1999.

[6] A. Doucet, N. d. Freitas, K. P. Murphy, and S. J. Russell, “Rao-Blackwellised particle filtering for dynamic Bayesian networks,” in *Proc. Conf. Uncertainty in Artificial Intelligence (UAI)*, San Francisco, CA, 2000, pp. 176–183.

[7] F. Daum, J. Huang, and A. Noushin, “Exact particle flow for nonlinear filters,” in *Proc. SPIE Conf. Signal Proc., Sensor Fusion, Target Recog.*, Orlando, FL, Apr. 2010, p. 769704.

[8] T. Ding and M. J. Coates, “Implementation of the Daum-Huang exact-flow particle filter,” in *Proc. IEEE Statistical Signal Processing Workshop (SSP)*, Ann Arbor, MI, Aug. 2012, pp. 257–260.

[9] S. Mori, F. Daum, and J. Douglas, “Adaptive step size approach to homotopy-based particle filtering Bayesian update,” in *Proc. Intl. Conf. Info. Fusion*, Heidelbreg, Germany, July 2016, pp. 2035–2042.

[10] S. Reich, “A guided sequential Monte Carlo method for the assimilation of data into stochastic dynamical systems,” in *Recent Trends in Dynamical Systems*, vol. 35, pp. 205–220. Springer Basel, 2013.

[11] P. Bunch and S. Godsill, “Approximations of the optimal importance density using Gaussian particle flow importance sampling,” *J. Amer. Statist. Assoc.*, vol. 111, no. 514, pp. 748–762, 2016.

[12] J. Heng, A. Doucet, and Y. Pokern, “Gibbs flow for approximate transport with applications to Bayesian computation,” *arXiv:1509.08787*, 2015.

[13] Y. Li and M. Coates, “Particle filtering with invertible particle flow,” *arXiv:1607.08799*, 2016.

[14] W. R. Gilks and C. Berzuini, “Following a moving target–Monte Carlo inference for dynamic Bayesian models,” *J. R. Stat. Soc. B*, vol. 63, no. 1, pp. 127–146, 2001.

[15] Z. Khan, T. Balch, and F. Dellaert, “MCMC-based particle filtering for tracking a variable number of interacting targets,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1805–1819, Nov 2005.

[16] F. Septier, S. K. Pang, A. Carmi, and S. Godsill, “On MCMC-based particle methods for Bayesian filtering: Application to multitarget tracking,” in *Proc. Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Aruba, Dutch Antilles, The Netherlands, Dec. 2009, pp. 360–363.

[17] A. Brockwell, P. Del Moral, and A. Doucet, “Sequentially interacting Markov chain Monte Carlo methods,” *Ann. Stat.*, vol. 38, no. 6, pp. 3387–3411, 2010.

[18] F. Septier and G. W. Peters, “Langevin and Hamiltonian based sequential MCMC for efficient Bayesian filtering in high-dimensional spaces,” *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 2, pp. 312–327, Mar. 2016.

[19] M. Girolami and B. Calderhead, “Riemann manifold Langevin and Hamiltonian Monte Carlo methods,” *J. R. Stat. Soc. B*, vol. 73, no. 2, pp. 123–214, 2011.