SEQUENTIAL MCMC WITH THE DISCRETE BOUNCY PARTICLE SAMPLER

Soumyasundar Pal and Mark Coates

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

ABSTRACT

Sequential MCMC (SMCMC) methods are a useful alternative to particle filters for performing sequential inference in a Bayesian framework in nonlinear and non-Gaussian state-space models. The weight degeneracy phenomenon which impacts the performance of even the most advanced particle filters in higher dimensions is avoided. In this paper, we explore the applicability of the discrete bouncy particle sampler, which is based on constructing a guided random walk and performing delayed rejection, to perform more effective sampling within SMCMC. We perform numerical simulations to examine when the proposed method offers advantages compared to state-of-the-art SMCMC techniques.

Index Terms— Markov chain Monte Carlo, Discrete Bouncy Particle Sampler, Bayesian inference, high dimensional filtering

1. INTRODUCTION

Particle filters [1] are one of the most widely used tools for performing Bayesian inference in a sequential setting, a crucial ingredient in many tracking and localization tasks. They rely on sequential importance sampling, which can be very inefficient in high dimensions as the weights of most of the particles become negligible, because they fall into the regions of state-space where the posterior is very low. This leads to a poor representation and high variance [2]. Although strategies have been explored to improve performance [3–5], particle filtering in very higher dimensions remains challenging.

Another direction in solving the filtering problem in higher dimensions is to use Markov chain Monte Carlo (MCMC) techniques in a sequential setting. Different variants of Sequential Markov chain Monte Carlo were proposed in [6–8]. In [9], Septier and Peters provide a unifying framework for SMCMC methods that employ composite Metropolis-Hastings (MH) kernels. At time-step k, the j'th iteration of these SMCMC algorithms involves a joint draw step to sample a trajectory $x^j_{k,0:k}$ from the approximate joint posterior and subsequently two rounds of refinements (partial updates to the samples) via *Metropolis within Gibbs* steps. First $x^j_{k,0:k-1}$ is refined conditioned on $x^j_{k,k}$; this is followed by refinement of $x^j_{k,k}$ conditioned on $x^j_{k,0:k-1}$. Septier and Peters [9] compared different MCMC techniques, developed to efficiently explore high dimensional spaces (e.g. MALA [10] and mHMC [11]), as refinement strategies for $x^j_{k,k}$ in the SMCMC framework.

Most MCMC algorithms are based on discrete time reversible Markov processes, where the transition kernels are variations of Metropolis-Hastings algorithm, but recently a novel class of non reversible piecewise deterministic MCMC (PD-MCMC) methods have been developed where the target distribution is explored using continuous time non-reversible piecewise deterministic Markov processes (PD-MP) [12, 13]. Theoretical and empirical results suggest that non-reversible Markov processes have better mixing properties and thus provide estimates of ergodic averages with lower variance [14]. Although the continuous-time methods are appealing, they have several limitations. The algorithms require simulation of event times (times when the continuous-time process is updated). The simulation is challenging and problem specific; often more information about the target distribution is required than being able to evaluate it and its gradient pointwise. For example, in order to implement the bouncy particle sampler (BPS) [15], which is a non-reversible rejection free MCMC method, one needs to simulate first the arrival times of a non-homogeneous Poisson process. This requires local upper bounds on the derivative of the log of the target distribution. In many problem settings it is difficult to derive such bounds and evaluating them is computationally expensive. Aside from these limitations, it is not straightforward to use such continuous time processes directly in the SMCMC framework. The refinement procedures in SMCMC require the use of a discrete MCMC scheme.

In recent work, [16, 17] propose several different discrete MCMC algorithms, which can be viewed as approximations to the continuous time PD-MCMC algorithms. These algorithms preserve the correct invariant distribution in spite of the discretizations. Moreover these techniques do not require event time simulation, which ensures that they have wider applicability than the continuous-time techniques. In fact, these methods can be implemented if we can evaluate the target and its derivative pointwise. In this paper, we propose the use of the discrete bouncy particle sampler (DBPS) [17] as the refinement technique within SMCMC. It was shown in [17] that the BPS of [15] can be understood as a scaling limit of a special case of the DBPS.

Our motivation in exploring the use of the discrete bouncy

particle sampler is to examine whether we can achieve improved performance compared to existing SMCMC algorithms that rely on Langevin or Hamiltonian-based refinement [9]. The discrete bouncy particle sampler has the potential to provide faster mixing with reduced computational cost, meaning that in a sequential setting such as tracking, where computation time is limited, we can generate more relevant samples from the posterior and achieve more accurate estimates of the location of a target and improved assessments of the uncertainty of these estimates.

The rest of the paper is organized as follows. Section 2 specifies the filtering problem we address. Section 3 reviews the unifying framework of the SMCMC methods. The proposed refinement scheme, based on DBPS is described in Section 4. Section 5 describes the numerical simulations and reports the results. In Section 6 we conclude the paper and summarize our observations.

2. PROBLEM STATEMENT

We consider the sequential inference task with the following state-space model:

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

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

Here $g_k : \mathbb{R}^d \times \mathbb{R}^{d'} \to \mathbb{R}^d$ denotes the transition function of the state $x_k \in \mathbb{R}^d$ at time step k and $\zeta_k \in \mathbb{R}^{d'}$ is the process noise. The observation $z_k \in \mathbb{R}^S$ is generated from 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. We also assume that $h_k(x_k, 0)$ is a C^1 function, i.e., $h_k(x_k, 0)$ is a differentiable function whose first derivative is continuous. Given a set of observations $z_{1:k} = \{z_i, i = 1, \ldots, k\}$ and an initial distribution $p(x_0)$, our goal is to generate samples from the posterior $p(x_{0:k}|z_{1:k})$ and the marginal posterior $p(x_k|z_{1:k})$. These samples can be used to form estimates of x_k and to assess uncertainty.

3. SMCMC

At time step k, the target distribution $p(x_{0:k}|z_{1:k})$ is denoted by $\pi_k(x_{0:k})$ and can be computed pointwise up to a constant in a recursive manner as follows :

$$\pi_k(x_{0:k}) \propto p(x_k|x_{k-1})p(z_k|x_k)\pi_{k-1}(x_{0:k-1})\,. \tag{3}$$

As $\pi_{k-1}(x_{0:k-1})$ is not analytically tractable, it is impossible to sample from it in a general nonlinear state-space model. In all SMCMC methods, that distribution is replaced by its empirical approximation in (3), which leads to an approximation of the target distribution as follows:

$$\breve{\pi}(x_{0:k}) \propto p(x_k|x_{k-1})p(z_k|x_k)\hat{\pi}_{k-1}(x_{0:k-1})\,; \tag{4}$$

where,

$$\hat{\pi}_{k-1}(x_{0:k-1}) = \frac{1}{N_p}\sum_{s=N_b+1}^{N_b+N_p}\delta(x_{0:k-1} - x^s_{k-1,0:k-1})\,. \tag{5}$$

Here, at time $k-1$, the N_p samples obtained from the Markov chain are denoted by $\{x^s_{k-1,0:k-1}\}_{s=N_b+1}^{N_b+N_p}$, whose stationary distribution is $\breve{\pi}_{k-1}(x_{0:k-1})$. It is noted in [9] that $\breve{\pi}_k \to \pi_k$, because $\hat{\pi}_{k-1} \to \pi_{k-1}$ as $N_p \to \infty$. At time step k, $N_b + N_p$ iterations of SMCMC are run to generate samples $\{x^s_{k,0:k}\}_{s=N_b+1}^{N_b+N_p}$ from the invariant distribution $\breve{\pi}_k(x_{0:k})$, and then the target is approximated as:

$$\hat{\pi}_k(x_{0:k}) = \frac{1}{N_p}\sum_{s=N_b+1}^{N_b+N_p}\delta(x_{0:k} - x^s_{k,0:k})\,. \tag{6}$$

In filtering problems, we are only interested in approximating the marginal posterior distribution of x_k, which is obtained from Eq. (6). In that case, we only need to store $\{x^s_{k-1,k-1}\}_{s=N_b+1}^{N_b+N_p}$ from the previous time step $k-1$, instead of the full trajectories of the samples. Different choices of the MCMC kernels are discussed in [9]. If the target distribution has high dimensionality or complex structure, it is very difficult to construct a Metropolis-Hastings (MH) kernel that is well-matched to the target. This causes poor mixing and the overall performance deteriorates. Instead [9, 18] propose the use of composite MH kernels that combine a joint proposal (where both $x^j_{k,0:k-1}$ and $x^j_{k,k}$ are updated) with subsequent individual state variable refinements using *Metropolis within Gibbs* steps, within a single MCMC iteration. The algorithm is summarized in Algorithm 1.

Algorithm 1: Composite MH Kernels in a unifying framework of SMCMC [9, 18].
Input: $x^{j-1}_{k,0:k}$.
Output: $x^j_{k,0:k}$.

Joint draw:
1: Propose $x^*_{k,0:k} \sim q_{k,1}(x_{0:k}|x^{j-1}_{k,0:k})$;
2: Compute the MH acceptance probability $\rho_1 = \min\big(1, \frac{\breve{\pi}_k(x^*_{k,0:k})}{q_{k,1}(x^*_{k,0:k}|x^{j-1}_{k,0:k})}\frac{q_{k,1}(x^{j-1}_{k,0:k}|x^*_{k,0:k})}{\breve{\pi}_k(x^{j-1}_{k,0:k})}\big)$;
3: Accept $x^j_{k,0:k} = x^*_{k,0:k}$ with probability ρ_1, otherwise set $x^j_{k,0:k} = x^{j-1}_{k,0:k}$;
Individual refinement of $x^j_{k,0:k-1}$:
4: Propose $x^*_{k,0:k-1} \sim q_{k,2}(x_{0:k-1}|x^j_{k,0:k})$;
5: Compute the MH acceptance probability $\rho_2 = \min\big(1, \frac{\breve{\pi}_k(x^*_{k,0:k-1}, x^j_{k,k})}{q_{k,2}(x^*_{k,0:k-1}|x^j_{k,0:k})}\frac{q_{k,2}(x^j_{k,0:k-1}|x^*_{k,0:k-1}, x^j_{k,k})}{\breve{\pi}_k(x^j_{k,0:k})}\big)$;
6: Accept $x^j_{k,0:k-1} = x^*_{k,0:k-1}$ with probability ρ_2;
Individual refinement of $x^j_{k,k}$:
7: Propose $x^*_{k,k} \sim q_{k,3}(x_k|x^j_{k,0:k})$;
8: Compute the MH acceptance probability $\rho_3 = \min\big(1, \frac{\breve{\pi}_k(x^j_{k,0:k-1}, x^*_{k,k})}{q_{k,3}(x^*_{k,k}|x^j_{k,0:k})}\frac{q_{k,3}(x^j_{k,k}|x^j_{k,0:k-1}, x^*_{k,k})}{\breve{\pi}_k(x^j_{k,0:k})}\big)$;
9: Accept $x^j_{k,k} = x^*_{k,k}$ with probability ρ_3;

In this paper, we use invertible particle flow [19] to con-

struct a composite MH kernel for the joint draw in SMCMC, as proposed in [20]. This was shown to lead to a higher acceptance rate in the joint draw step compared to employing an independent MH kernel based on the prior as the proposal, as used in the numerical experiments with the SmHMC algorithm in [9]. Both the localized exact Daum and Huang (LEDH) [21] flow and the exact Daum and Huang (EDH) [22] flow can be used to construct the kernel. We use EDH because the computational requirements are much less. For individual refinement of $x^j_{k,0:k-1}$, we use $q_{k,2} = \hat{\pi}_{k-1}$.

4. INDIVIDUAL REFINEMENT BASED ON DISCRETE BOUNCY PARTICLE SAMPLER

In this section, we introduce a novel individual refinement scheme for x_k (lines 7 and 8 of Algorithm 1) based on the discrete bouncy particle sampler (DBPS) [17]. The refinement of x_k consists of a *Metropolis within Gibbs* step to sample a new value of $x^j_{k,k} \sim \breve{\pi}(x^j_{k,0:k-1}, x_k)$, based on a local proposal. We propose to use the DBPS algorithm to accomplish this. The DBPS scheme is run for N_{thinning} iterations, initialized at $x^{(0)}_k = x^j_{k,k}$, and the output of the Markov chain $x_k^{(N_{\text{thinning}})}$ is taken to be the refined $x^j_{k,k}$. Pseudocode is provided in Algorithm 2.

The algorithm is based upon a guided random walk and a delayed rejection procedure [23]. For simplicity, let us assume that the preconditioning matrix, M is the identity matrix for now. We will explain its role below.

Once converged, each iteration of DBPS samples form a extended joint distribution of (x_k, v) in a $2d$ dimensional space, which admits $\phi(x_k)$ as its x_k marginal. The orientation v is an auxiliary variable used to explore the space by local moves. Its marginal is spherically symmetric.

At the beginning of i'th iteration of the DBPS, a move from the current state $(x_k^{(i-1)}, v^{(i-1)})$ to $(x_k^{(i-1)} + v^{(i-1)}, -v^{(i-1)})$ is proposed and accepted with the MH acceptance probability. This move is thus reversible with respect to the extended target. Irrespective of the acceptance or rejection of the move, a flip move in orientation v always follows. The net result of two reversible moves is a non-reversible Markov chain which keeps moving in the same direction, until a rejection happens at line 7 in algorithm 2. If $(x_k^{(i-1)} + v^{(i-1)}, -v^{(i-1)})$ is rejected, the gradient at the rejected point is calculated and a reflection of $(x_k^{(i-1)}, v^{(i-1)})$ in the tangent hyperplane perpendicular to the gradient is proposed. The acceptance probability of this proposal is set in such a way that the detailed balance is preserved with respect to the extended target, following the delayed rejection approach.

The preconditioning helps the procedure to explore the space in a more efficient way if the target has significantly different scales across its various dimensions. When we multiply by M^{-1}, we adjust for these differences. This eliminates the need to find a suitable step-size to propose the local move, without affecting the desirable theoretical properties of the DBPS.

Algorithm 2: Individual refinement according to DBPS [17].
Input: $x_k^{(0)}, N_{\text{thinning}}$.
Output: $x_k^{(N_{\text{thinning}})}$.

1: Define the target distribution : $\phi(x_k) = \breve{\pi}(x^j_{k,0:k-1}, x_k) \propto p(x_k|x^j_{k,k-1})p(z_k|x_k)$;
2: Compute $\Gamma \approx -\mathbb{E}_{z_k|x_k}[\nabla^2 \ln \phi(x_k)]|_{x_k = x_k^{(0)}}$;
3: Compute the preconditioning matrix, M, from the Cholesky decomposition of Γ, i.e., $\Gamma = M^T M$;
4: Sample initial orientation $v_*^{(0)}$ from the uniform distribution on the unit d dimensional hypersphere;
5: Apply preconditioning : $v^{(0)} = M^{-1} v_*^{(0)}$;
6: **for** $i = 1$ to N_{thinning} **do**
7: Propose, $(x_k'^{(i)}, v'^{(i)}) = (x_k^{(i-1)} + v^{(i-1)}, -v^{(i-1)})$;
8: Accept the proposal $(x_k^{(i)}, v^{(i)}) \leftarrow (x_k'^{(i)}, v'^{(i)})$ with probability, $\alpha(x_k^{(i-1)}, x_k'^{(i)}) = \min\big(1, \frac{\phi(x_k'^{(i)})}{\phi(x_k^{(i-1)})}\big)$, and go to step 11;
9: Propose, $(x_k''^{(i)}, v''^{(i)})$ where

$$v''^{(i)} = -v^{(i-1)} + 2\frac{\langle v^{(i-1)}, \mathrm{w}'(i)\rangle}{\mathrm{w}'(i)^T \Gamma \mathrm{w}'(i)} \Gamma \mathrm{w}'(i)\,,$$

$\mathrm{w}'(i) = \nabla \ln \phi(x_k)|_{x_k = x_k'^{(i)}}$ and
$x_k''^{(i)} = x_k^{(i-1)} + v^{(i-1)} - v''^{(i)}$;
10: Accept the proposal $(x_k^{(i)}, v^{(i)}) \leftarrow (x_k''^{(i)}, v''^{(i)})$ with probability, $\alpha(x_k^{(i-1)}, x_k'^{(i)}, x_k''^{(i)}) = \min\big(1, \frac{(1-\alpha(x_k''^{(i)}, x_k'^{(i)}))}{(1-\alpha(x_k^{(i-1)}, x_k'^{(i)}))} \frac{\phi(x_k''^{(i)})}{\phi(x_k^{(i-1)})}\big)$, otherwise set $(x_k^{(i)}, v^{(i)}) = (x_k^{(i-1)}, v^{(i-1)})$;
11: Update $(x_k^{(i)}, v^{(i)}) \leftarrow (x_k^{(i)}, -v^{(i)})$;
12: **end for**

5. NUMERICAL EXPERIMENTS AND RESULTS

The SmHMC (SMCMC with manifold Hamiltonian Monte Carlo kernel) algorithm achieves the lowest average MSE among a variety of SMCMC algorithms in the large sensor field simulation setup used in [9]. Thus, we would like to compare the proposed SDBPS+EDH with SmHMC and SmHMC+EDH in the same simulation example. The setup has d sensors uniformly deployed on a two-dimensional grid $\{1, 2, \ldots, \sqrt{d}\} \times \{1, 2, \ldots, \sqrt{d}\}$. The observation $z_k = [z_k^1, z_k^2, \ldots, z_k^d]$ contains noisy measurements conditioned on the state vector $x_k = [x_k^1, x_k^2, \ldots, x_k^d]$. The state transition density follows a multivariate generalized hyperbolic skewed-t distribution:

$$p(x_k|x_{k-1}) = \frac{e^{(x_k-\alpha x_{k-1})^T\Sigma^{-1}\gamma}}{\sqrt{(\nu+Q(x_k))(\gamma^T\Sigma^{-1}\gamma)}^{-\frac{\nu+d}{2}}(1+\frac{Q(x_k)}{\nu})^{\frac{\nu+d}{2}}} \times K_{\frac{\nu+d}{2}}(\sqrt{(\nu+Q(x_k))(\gamma^T\Sigma^{-1}\gamma)}) \quad (7)$$

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:

$$\Sigma_{i,j} = \alpha_0 e^{-\frac{||R^i - R^j||_2^2}{\beta}} + \alpha_1\delta_{i,j} \quad (8)$$

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 covariance is given as :

$$\tilde{\Sigma} = \frac{\nu}{\nu-2}\Sigma + \frac{\nu^2}{(2\nu-8)(\frac{\nu}{2}-1)^2}\gamma\gamma^T . \quad (9)$$

The measurements are count data which are distributed according to:

$$p(z_k|x_k) = \prod_{c=1}^{d}\mathcal{P}(z_k^c; m_1 e^{m_2 x_k^c}), \quad (10)$$

where $\mathcal{P}(\cdot;\Lambda)$ is the Poisson distribution with mean Λ. Parameter values are set according to [9]: $\alpha = 0.9, \alpha_0 = 3, \alpha_1 = 0.01, \beta = 20, \nu = 7$. All elements of the skewness parameter vector γ are set to 0.3. True states start with $x_0^c = 0$, for $c = 1,\ldots,d$. For the measurement model, $m_1 = 1$ and $m_2 = \frac{1}{3}$. We set $d = 144$. The experiment is executed 100 times for 10 time steps. We approximate the Γ matrix required in line 2 in algorithm (2) as follows :

$$\Gamma = \tilde{\Sigma}^{-1} + \Lambda(x_k^{(0)}), \quad (11)$$

where $\Lambda(x_k)$ is a $d \times d$ diagonal matrix with $\Lambda(x_k)_{(c,c)} = m_1 m_2^2 e^{m_2 x_k^c}$.

We report mean square error obtained at each sensor location, acceptance rates (if applicable) and execution time per step in Table 1. We observe that the proposed SDBPS+EDH algorithm performs slightly better than SmHMC+EDH. Both SmHMC and SDBPS algorithms exhibit considerably lower MSE if particle flow is incorporated in the joint draw. The EDH filter achieves the smallest MSE and is computationally very fast, but it is not a statistically consistent algorithm, because of the approximations in the flow and modelling mismatch. The PF-PF (EDH) [19] algorithm performs reasonably whereas the BPF has the highest MSE because of weight degeneracy. As the computational burden for various SMCMC methods are quite different form one another, we evaluate the SMCMC methods while varying the number of particles to understand the tradeoff between accuracy and execution time. Figure 1 shows the variation of MSE with execution time for the SMCMC methods considered here. For comparable computational cost, the SMCMC algorithms employing the DBPS can generate many more samples than those employing HMC.

Table 1. Average MSE, acceptance rates (if applicable) and execution time per step based on 100 simulation trials. Results are produced with an Intel i7-4770K 3.50GHz CPU and 32GB RAM.

Algorithm	No. of Particles	Avg. MSE	Acceptance rate			Exec. time (s)
			ρ_1	ρ_2	ρ_3	
SmHMC	1000	0.81	0.0007	0.003	0.72	78.93
SDBPS	35000	0.99	0.02	0.0001	0.91	57.82
SmHMC + EDH	1000	0.72	0.0357	0.002	0.69	81.72
SDBPS + EDH	35000	0.71	0.02	0.0001	0.91	70.58
EDH	200	0.69	-	-	-	0.040
PF-PF (EDH)	10^5	0.80	-	-	-	6.07
BPF	10^6	1.39	-	-	-	5.32

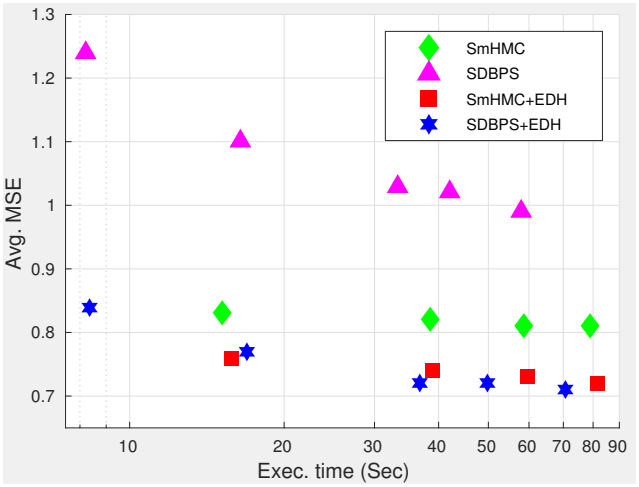

Fig. 1. MSE (averaged over 100 trials) vs Exec. time of various SMCMC algorithms, Exec. time is shown in log scale.

6. CONCLUSION

In this paper, we explore the use of the discrete bouncy particle sampler as a refinement strategy within the SMCMC framework. We use an independent Metropolis-Hastings kernel based on invertible particle flow in the joint draw step. We observe that for a similar computation time the proposed algorithm can generate many more samples than SMCMC algorithms employing (manifold) Hamiltonian Monte Carlo for refinement. Although many more samples are provided, this translates to only a small improvement in the accuracy of the state estimates.

Future research will involve more thorough experimentation with other state-space models and exploration of other PD-MCMC schemes in SMCMC. The promising performance of the discrete bouncy particle sampler suggests that there is value in exploring other discrete sampling algorithms that can better exploit the structure of the target distribution.

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] P. Bickel, B. Li, and T. Bengtsson, "Sharp failure rates for the bootstrap particle filter in high dimensions," in *Pushing the limits of contemporary statistics: Contributions in honor of Jayanta K. Ghosh*, pp. 318–329. Institute of Mathematical Statistics, May 2008.

[3] P. M. Djurić and M. F. Bugallo, "Particle filtering for high-dimensional systems," in *Proc. IEEE Intl. Workshop Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, St. Martin, France, Dec. 2013, pp. 352–355.

[4] A. Beskos, D. Crisan, A. Jasra, K. Kamatani, and Y. Zhou, "A stable particle filter for a class of high-dimensional state-space models," *Advances in Applied Probability*, vol. 49, no. 1, pp. 24–48, Mar. 2017.

[5] P. Rebeschini and R. van Handel, "Can local particle filters beat the curse of dimensionality?," *Ann. Appl. Probab.*, vol. 25, no. 5, pp. 2809–2866, Oct. 2015.

[6] A. Golightly and D. J. Wilkinson, "Bayesian sequential inference for nonlinear multivariate diffusions," *Stat. Comput.*, vol. 16, no. 4, pp. 323–338, May 2006.

[7] C. Berzuini, N. G. Best, W. R. Gilks, and C. Larizza, "Dynamic conditional independence models and Markov chain Monte Carlo methods," *J. Amer. Statist. Assoc.*, vol. 92, no. 440, pp. 1403–1412, Dec. 1997.

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

[9] 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.

[10] T. Xifara, C. Sherlock, S. Livingstone, S. Byrne, and M. Girolami, "Langevin diffusions and the Metropolis-adjusted Langevin algorithm," *Stat. Comput.*, vol. 91, pp. 14–19, Aug. 2014.

[11] 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, Mar. 2011.

[12] J. Bierkens, P. Fearnhead, and G. Roberts, "The zig-zag process and super-efficient sampling for Bayesian analysis of big data," *ArXiv e-prints arxiv:1607.03188*, July 2016.

[13] P. Fearnhead, J. Bierkens, M. Pollock, and G. O Roberts, "Piecewise deterministic Markov processes for continuous-time Monte Carlo," *ArXiv e-prints arxiv:1611.07873*, Nov. 2016.

[14] M. Ottobre, "Markov chain Monte Carlo and irreversibility," *Reports on Math. Physics*, vol. 77, no. 3, pp. 267 – 292, June 2016.

[15] A. Bouchard-Côté, S. J. Vollmer, and A. Doucet, "The bouncy particle sampler: A non-reversible rejection-free Markov chain Monte Carlo method," *ArXiv e-prints arXiv:1510.02451v6*, Oct. 2015.

[16] P. Vanetti, A. Bouchard-Côté, G. Deligiannidis, and A. Doucet, "Piecewise deterministic Markov chain Monte Carlo," *ArXiv e-prints arXiv:1707.05296*, July 2017.

[17] C. Sherlock and A. H. Thiery, "A discrete bouncy particle sampler," *ArXiv e-prints arXiv:1707.05200*, July 2017.

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

[19] Y. Li and M. Coates, "Particle filtering with invertible particle flow," *IEEE Trans. Signal Processing*, vol. 65, no. 15, pp. 4102–4116, Aug. 2017.

[20] Y. Li and M. Coates, "Sequential MCMC with invertible particle flow," in *Proc. Intl. Conf. Acoustics, Speech and Signal Proc. (ICASSP)*, New Orleans, USA, Mar. 2017, pp. 3844–3848.

[21] 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, USA, Aug. 2012, pp. 257–260.

[22] F. Daum and J. Huang, "Particle flow for nonlinear filters with log-homotopy," in *Proc. SPIE Conf. Signal and Data Processing of Small Targets*, Orlando, FL, USA, Apr. 2008, p. 696918.

[23] L. Tierney and A. Mira, "Some adaptive Monte Carlo methods for Bayesian inference," *Stat. Med.*, vol. 18, pp. 2507 – 2515, Aug. 1999.