Diffusion map-based algorithm for Gain function approximation in the Feedback Particle Filter*

Amirhossein Taghvaei†, Prashant G. Mehta†, and Sean P. Meyn‡

Abstract. Feedback particle filter (FPF) is a numerical algorithm to approximate the solution of the nonlinear filtering problem in continuous-time settings. In any numerical implementation of the FPF algorithm, the main challenge is to numerically approximate the so-called gain function. A numerical algorithm for gain function approximation is the subject of this paper. The exact gain function is the solution of a Poisson equation involving a probability-weighted Laplacian Δ_ρ. The numerical problem is to approximate this solution using *only* finitely many particles sampled from the probability distribution ρ. A diffusion map-based algorithm was proposed by the authors in a prior work [58, 60] to solve this problem. The algorithm is named as such because it involves, as an intermediate step, a diffusion map approximation of the exact semigroup e^{Δ_ρ}. The original contribution of this paper is to carry out a rigorous error analysis of the diffusion map-based algorithm. The error is shown to include two components: bias and variance. The bias results from the diffusion map approximation of the exact semigroup. The variance arises because of finite sample size. Scalings and upper bounds are derived for bias and variance. These bounds are then illustrated with numerical experiments that serve to emphasize the effects of problem dimension and sample size. The proposed algorithm is applied to two filtering examples and comparisons provided with the sequential importance resampling (SIR) particle filter.

Key words. Stochastic Processes, Nonlinear filtering, Poisson equation

AMS subject classifications. 93E11, 65N75, 65N15

1. Introduction.

This paper is concerned with a numerical solution of a certain linear partial differential equation (PDE) that arises in nonlinear filtering problem in continuous-time settings.

Nonlinear filtering problem: The standard model of the nonlinear filtering problem is given by the following stochastic differential equations (SDE) [65]:

$$\text{State process:} \quad \mathrm{d}X_t = a(X_t)\,\mathrm{d}t + \mathrm{d}B_t, \quad X_0 \sim p_0 \tag{1.1a}$$

$$\text{Observation process:} \quad \mathrm{d}Z_t = h(X_t)\,\mathrm{d}t + \mathrm{d}W_t, \tag{1.1b}$$

where $X_t \in \mathbb{R}^d$ is the (hidden) state at time t, $Z_t \in \mathbb{R}$ is the observation, and B_t, W_t are two mutually independent standard Wiener processes (w.p.) taking values in $\mathbb{R}^d$ and $\mathbb{R}$, respectively. The mappings $a(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ and $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$ are known C^1 functions, and p_0 is the density of the prior probability distribution. The scalar-valued observation is considered here for notational ease. The extension to the general vector-valued observation is straightforward and described in Remark 3.3.

*Submitted to the editors May 12, 2020.

Funding: Financial support from the NSF grant 1761622 and the ARO grant W911NF1810334 is gratefully acknowledged.

†Department of Mechanical Science and Engineering (MechSe), Coordinated Science Laboratory (CSL), University of Illinois at Urbana-Champaign, Urbana, IL (taghvae2@illinois.edu, mehtapg@illinois.edu).

‡Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL (spmeyn@gmail.com)

The objective of the filtering problem is to compute the posterior distribution of the state X_t given the time history of observations (filtration) $\mathcal{Z}_t := \sigma(Z_s : 0 \le s \le t)$.

The problem is *linear Gaussian* if $a(\cdot)$, and $h(\cdot)$ are linear functions and p_0 is a Gaussian density. We use A and H to denote the matrices that define these linear functions, i.e, $a(x) = Ax$ and $h(x) = Hx$. The background on the linear Gaussian problem, along with its solution given by the Kalman-Bucy filter [35], appears in [40].

Feedback particle filter (FPF) is a numerical algorithm to approximate the posterior distribution in nonlinear non-Gaussian settings [67, 66]. The FPF algorithm is an alternative to the sequential importance resampling (SIR) particle filters [30, 25, 3, 22]. The distinguishing feature of the FPF is that the importance sampling step is replaced with feedback control. Steps such as resampling, reproduction, death or birth of particles are altogether avoided. The particles in FPF have uniform importance weights by construction. Therefore, the FPF does not suffer from the particle degeneracy issue that is commonly observed in implementations of the SIR particle filters [25]. In independent numerical evaluations and comparisons, it has been observed that FPF exhibits smaller simulation variance and better scaling properties with the problem dimension [9, 54, 55]. However, as is the focus of the remainder of this paper, implementing the FPF algorithm is computationally challenging because of the gain function approximation problem (1.5).

The construction of FPF is based on the following two steps:

Step 1: Construct a stochastic process, denoted by $\bar{X}_t \in \mathbb{R}^d$, whose conditional distribution (given $\mathcal{Z}_t$) is equal to the conditional distribution of X_t;

Step 2: Simulate N stochastic processes, denoted by $\{X_t^i\}_{i=1}^N$, to empirically approximate the distribution of $\bar{X}_t$.

$$\underbrace{\mathsf{E}[f(X_t)|\mathcal{Z}_t] \overset{\text{Step 1}}{=} \mathsf{E}[f(\bar{X}_t)|\mathcal{Z}_t]}_{\text{exactness condition}} \overset{\text{Step 2}}{\approx} \frac{1}{N}\sum_{i=1}^{N} f(X_t^i).$$

The process $\bar{X}_t$ is referred to as mean-field process and the N processes $\{X_t^i\}_{i=1}^N$ are referred to as particles. The construction ensures that the filter is *exact* in the mean-field ($N = \infty$) limit.

The details of the two steps are as follows:

Mean-field process: In the FPF, the mean-field process $\bar{X}_t$ evolves according to the SDE given by

$$\mathrm{d}\bar{X}_t = \underbrace{a(\bar{X}_t)\,\mathrm{d}t + \mathrm{d}\bar{B}_t}_{\text{propagation}} + \underbrace{\mathsf{K}_t(\bar{X}_t) \circ (\,\mathrm{d}Z_t - \frac{h(\bar{X}_t) + \hat{h}_t}{2}\,\mathrm{d}t)}_{\text{feedback control law}}, \quad \bar{X}_0 \sim p_0, \tag{1.2}$$

where $\bar{B}_t$ is a standard Wiener processes independent of $\bar{X}_0$ and $\hat{h}_t := \mathsf{E}[h(\bar{X}_t)|\mathcal{Z}_t]$. The $\circ$ indicates that the SDE is expressed in its Stratonovich form. The gain function is $\mathsf{K}_t(x) :=$

$\nabla\phi_t(x)$ where ϕ_t is the solution of the Poisson equation:

$$\begin{aligned} &\text{Poisson equation:} \quad \frac{1}{p_t(x)}\nabla\cdot(p_t(x)\nabla\phi_t(x)) = -(h(x)-\hat{h}_t), \quad \forall\, x\in\mathbb{R}^d, \\ &\text{s.t.} \quad \int \phi_t(x)p_t(x)\,\mathrm{d}x = 0, \end{aligned} \tag{1.3}$$

where ∇ and $\nabla\cdot$ denote the gradient and the divergence operators, respectively, and p_t denotes the conditional density of $\bar{X}_t$ given $\mathcal{Z}_t$. The operator on the left-hand side of the Poisson equation (1.3) is referred to as the *probability-weighted Laplacian*. It is denoted as Δ_ρ where the probability density ρ is the conditional density p_t.

Particles: The particles $\{X_t^i\}_{i=1}^N$ evolve according to:

$$\mathrm{d}X_t^i = a(X_t^i)\,\mathrm{d}t + \mathrm{d}B_t^i + \mathsf{K}_t^{(N)}(X_t^i)\circ(\mathrm{d}Z_t - \frac{h(X_t^i)+\hat{h}_t^{(N)}}{2}\,\mathrm{d}t), \quad X_0^i \overset{\text{i.i.d.}}{\sim} p_0, \tag{1.4}$$

for $i=1,\dots N$, where $\{B_t^i\}_{i=1}^N$ are mutually independent standard Wiener processes, $\hat{h}_t^{(N)} := \frac{1}{N}\sum_{i=1}^N h(X_t^i)$, and $\mathsf{K}_t^{(N)}$ is the output of an algorithm that approximates the solution to the Poisson equation (1.3)

$$\text{Gain function approximation:} \quad \mathsf{K}_t^{(N)} := \text{Algorithm}(\{X_t^i\}_{i=1}^N; h). \tag{1.5}$$

The notation is suggestive of the fact that algorithm is adapted to the ensemble $\{X_t^i\}_{i=1}^N$ and the function h; the density $p_t(x)$ is not known in an explicit manner.

Development and error analysis of one such gain function approximation algorithm is the subject of the present paper. Before describing the general case, it is useful to review the filter for the linear Gaussian case where the solution of the Poisson equation is explicitly known.

FPF for Linear Gaussian setting: Suppose $a(x)=Ax$, $h(x)=Hx$, and p_t is a Gaussian density with mean $\bar{m}_t$ and variance $\bar{\Sigma}_t$. Then the solution of the Poisson equation is known in an explicit form [66, Sec. D]. The resulting gain function is constant and equal to the Kalman gain:

$$\mathsf{K}_t(x) \equiv \bar{\Sigma}_t H^\top, \quad \forall\, x\in\mathbb{R}^d. \tag{1.6}$$

Therefore, the mean-field process (1.2) for the linear Gaussian problem is given by:

$$\mathrm{d}\bar{X}_t = A\bar{X}_t\,\mathrm{d}t + \mathrm{d}\bar{B}_t + \bar{\Sigma}_t H^\top(\mathrm{d}Z_t - \frac{H\bar{X}_t + H\bar{m}_t}{2}\,\mathrm{d}t), \quad \bar{X}_0 \sim p_0.$$

Given the explicit form of the gain function (1.6), the empirical approximation of the gain is simply $\mathsf{K}_t^{(N)} = \Sigma_t^{(N)} H^\top$ where $\Sigma_t^{(N)}$ is the empirical covariance of the particles. Therefore, the evolution of the particles is:

$$\mathrm{d}X_t^i = AX_t^i\,\mathrm{d}t + \mathrm{d}B_t^i + \mathsf{K}_t^{(N)}(\mathrm{d}Z_t - \frac{HX_t^i + Hm_t^{(N)}}{2}\,\mathrm{d}t), \quad X_0^i \overset{\text{i.i.d.}}{\sim} p_0, \tag{1.7}$$

for $i = 1\ldots, N$, where $m_t^{(N)}$ is the empirical mean of the particles. The empirical quantities are computed as:

$$m_t^{(N)} := \frac{1}{N}\sum_{i=1}^{N} X_t^i, \quad \Sigma_t^{(N)} := \frac{1}{N-1}\sum_{i=1}^{N}(X_t^i - m_t^{(N)})(X_t^i - m_t^{(N)})^\top.$$

The linear Gaussian FPF (1.7) is identical to the square-root form of the ensemble Kalman filter (EnKF) [8, Eq. 3.3].

One extension of the Kalman gain is the so called *constant gain approximation* formula whereby the gain K_t is approximated by its expected value (which represents the best least-squared approximation of the gain by a constant). Remarkably, the expected value admits a closed-form expression which is then readily approximated empirically using the particles (see Remark 2.3 for derivation):

$$\begin{aligned}\text{Const. gain approx:} \quad \mathsf{E}[\mathsf{K}_t(X_t)|\mathcal{Z}_t] &= \int_{\mathbb{R}^d} (h(x) - \hat{h}_t)\, x\, p_t(x)\, \mathrm{d}x \\ &\approx \frac{1}{N}\sum_{i=1}^{N} (h(X_t^i) - \hat{h}_t^{(N)})\, X_t^i.\end{aligned} \tag{1.8}$$

The constant gain approximation formula has been used in nonlinear extensions of the EnKF algorithm [21]. The connection to the Poisson equation provides a justification for this formula. The formula is attractive because it provides a consistent (as the number of particles $N \to \infty$) approximation of the Kalman gain in the linear Gaussian setting.

Design and analysis of the gain function approximation algorithm (1.5) in the general case is a challenging problem because of two reasons: (i) Apart from the Gaussian case, there are no known closed-form solutions of (2.1); (ii) The density $p_t(x)$ is not explicitly known. At each time-step, one only has samples $\{X_t^i\}_{i=1}^N$. For the purpose of this paper, these samples are assumed to be i.i.d. drawn from p_t. The assumption is justified because in the limit of large N, the particles are approximately i.i.d. (by the propagation of chaos); cf., [56].

1.1. Contributions of this paper. The paper presents a diffusion map-based algorithm for the gain function approximation problem. The algorithm is named as such because it involves, as an intermediate step, a diffusion map approximation of the exact semigroup e^{Δ_ρ}. The following is a summary of specific original contributions made in this paper:

(i) Error estimates that relate the exact semigroup to its diffusion map approximation. The error estimates are derived by employing a Feynman-Kac representation of the semigroup (Proposition 3.4);

(ii) A uniform spectral gap for the diffusion map based on the use of the Foster-Lyapunov function method from the theory of stochastic stability of Markov processes (Proposition 4.2); and

(iii) Error estimates for the empirical approximation of the diffusion map (Proposition 3.5).

The results from (i) and (ii) are used to derive estimates for the bias and to show that the bias converges to zero in a certain limit (Theorem 4.3). Results from (iii) are used to prove

the convergence of the variance error term to zero in the infinite-N limit (Theorem 4.4). The paper contains numerical experiments that serve to illustrate the effects of problem dimension and sample size. The algorithm is applied to two filtering examples and comparisons provided with the sequential importance resampling (SIR) particle filter.

1.2. Relationship to prior work. The gain function algorithm first appeared in the conference version of this paper [58]. Its preliminary error analysis was reported in the conference paper [60]. The important distinction is that the results in these conference papers were preliminary in nature. The proofs were either altogether omitted or based on formal arguments. The main techniques employed in this paper, namely, (i) the use of Feyman-Kac representation to quantify the error due to the diffusion map approximation of the exact semigroup, and (ii) the use of stochastic stability theory to derive uniform spectral gap for the diffusion map, are original and do not appear in the conference papers. These techniques are important to be able to obtain precise estimates as enumerated above in the list of contributions. Since the main technical tools are new, *all* the proofs, based on these techniques, are new and original contributions of this paper. The diffusion map was introduced in [15], in the context of spectral clustering [6, 63]. Results on its convergence analysis appears in [32, 52, 15, 28, 31, 64, 7]. The use of diffusion map approximations for filtering problems is originally due to the authors.

1.3. Literature survey. Apart from its direct relevance to numerical approximation of the FPF, there are three topics of current research interest that are relevant to the subject of this paper: (i) ensemble Kalman filter; (ii) particle flow algorithms for nonlinear filtering; and (iii) optimal transport. Specifically, the algorithms for gain function approximation described in this paper are also directly applicable to these other topics. These relationships are briefly discussed next:

Ensemble Kalman filter: The EnKF algorithm was first developed in the discrete-time setting [27]. In the continuous-time setting, two formulations of the EnKF have been developed: stochastic EnKF, and the more recent deterministic EnKF [8, 51]. As has already been noted, the deterministic EnKF is in fact identical to the FPF algorithm (1.7) in the linear Gaussian setting [8, 57].

The EnKF algorithm provides a consistent approximation in the linear Gaussian setting. Compared to the Kalman filter, the main utility of EnKF is that it does not require propagation of the covariance matrix. This reduces the computational complexity from $O(d^2)$ for the Kalman filter to $O(Nd)$. This is clearly advantageous in high dimensional problems when $N \ll d$. This property has made EnKF popular in applications such as weather prediction in high dimensional settings [36, 47]. The disadvantage of the EnKF algorithm, of course, is that it does not provide a consistent approximation for nonlinear problems.

FPF represents a the generalization of the EnKF to the nonlinear non-Gaussian setting [57]: With the constant gain approximation, the algorithms are identical. Given this parallel, the problem of improving the EnKF algorithm in more general nonlinear non-Gaussian settings is directly related to the problem of better approximating the gain function in the FPF. In an application software based on EnKF, it is a relatively simple matter to replace the constant gain formula for the gain by more sophisticated approximations described in this paper. Certain empirical evaluations on the performance of FPF in high-dimensional settings

are reported in [55, 54, 53, 9].

Error analysis and stability of EnKF is an active area of research; see [43, 41, 24] for linear models and [21, 23, 37] for nonlinear models. The error analysis for the gain function approximation reported in this paper is a step towards error analysis of the FPF along these lines.

Particle flow algorithms: The following first-order (and hence an under determined) form of the Poisson equation appears in most types of particle flow algorithms:

$$\nabla \cdot (p_t(x)\mathsf{K}(x)) = (\text{rhs}),$$

where the righthand-side (rhs) is given and $\mathsf{K}(x)$ defines a vector field that must be obtained to implement the particle flow. The PDE appears in the first interacting particle representation of the continuous-time filtering in [17, 18] and the discrete-time filtering in [19]. Stochastic extensions of these have also recently appeared in [20] where approximate solutions are also described based on Gaussian assumption on the density. The algorithm described here represent an approximation of a particular gradient form solution of the first-order PDE.

Optimal transport: The mean-field SDE (1.2) represents a transport that maps the prior distribution at time 0 to the posterior distribution at an (arbitrary) future time $t > 0$. Synthesis of optimal transport maps for implementing the Bayes formula appears in [50, 14, 26, 59, 33, 13]. The relationship with the Poisson equation is through the ensemble transform filter which relies on a linear programming construction to approximate the optimal transport map [14]. As discussed in [57, Sec. 5.5], the solution of the Poisson equation yields an infinitesimal optimal transport map from the "prior" $p_t(x)$ to "posterior" $\frac{1}{\gamma}p_t(x)e^{-th(x)}$, where γ is the normalization constant. Another closely related approach is transportation through Gibbs flow [33].

Directly related to the FPF, the Galerkin method for the numerical solution of the Poisson equation appeared in original papers [66, 67]. The Galerkin algorithm represents the 'direct" PDE approach to construct a numerical approximation. The constant gain approximation is a particular example of a Galerkin solution. In general, the main problem with the Galerkin approximation is that it requires a selection of basis functions. This becomes intractable in high dimensions. To mitigate this issue, a proper orthogonal decomposition (POD)-based procedure to select basis functions is introduced in [11]. Other existing approaches are a continuation scheme for approximation [44], a probabilistic approach based on dynamic programming [48], and a procedure based on expressing the gain function in a reproducible Hilbert kernel space [49]. A comparison of different gain function approximation methods appears in [10].

1.4. Paper outline. The outline of the remainder of this paper is as follows: The mathematical problem of the gain function approximation together with a summary of known results on this topic appears in section 2. The diffusion-map based algorithm is described in a self-contained fashion in section 3. The main theoretical results of this paper including the bias and variance estimates appear in section 4. Some numerical experiments for the same appear in section 5. All the proofs appear as part of the supplementary material.

1.5. Notation. For vectors $x, y \in \mathbb{R}^d$, the dot product is denoted as $x \cdot y$ and $|x| := \sqrt{x \cdot x}$. The space of positive definite $d \times d$ matrices is denoted as S^d_{++}. The Borel σ-algebra on $\mathbb{R}^d$ is denoted by $\mathcal{B}(\mathbb{R}^d)$. The indicator function, for a measurable set $A \in \mathbb{B}(\mathbb{R}^d)$, is denoted as $\mathbb{1}_A(\cdot)$. The space of measurable functions $f : \mathbb{R}^d \to \mathbb{R}$ such that $\|f\|_{L^p(\rho)} := \left(\int |f(x)|^p \rho(x)\,\mathrm{d}x\right)^{1/p} < \infty$ is denoted as $L^p(\rho)$. The inner product on $L^2(\rho)$ is defined by $\langle f, g \rangle := \int f(x)g(x)\rho(x)\,\mathrm{d}x$. The space $H^1(\rho)$ is the space functions $f \in L^2(\rho)$ whose derivative (defined in the weak sense) is in $L^2(\rho)$. For a (weakly) differentiable function f, $\|\nabla f\|_{L^p(\rho)} := \left(\int |\nabla f(x)|^p \rho(x)\,\mathrm{d}x\right)^{1/p}$. For an integrable function f, $\hat{f}_\rho := \int f(x)\rho(x)\,\mathrm{d}x$ denotes the mean. $L^2_0(\rho) := \{f \in L^2(\rho) \mid \hat{f}_\rho = 0\}$ and $H^1_0(\rho) := \{f \in H^1(\rho) \mid \hat{f}_\rho = 0\}$ denote the co-dimension 1 subspace of functions whose mean is zero. $L^\infty(\Omega)$ denotes the space of bounded functions on $\Omega \subset \mathbb{R}^d$ with the sup-norm denoted as $\|\cdot\|_{L^\infty(\Omega)}$. The space of bounded and continuous functions on $\Omega \subset \mathbb{R}^d$ and the space of bounded and smooth functions on Ω is denoted as $C_b(\Omega)$ and $C^\infty_b(\Omega)$ respectively. For a linear operator T, on a Banach space $\mathcal{X}$ with norm $\|\cdot\|_{\mathcal{X}}$, the operator norm is denoted as $\|T\|_{\mathcal{X}}$. The Gaussian distribution with mean m and covariance Σ is denoted as $\mathcal{N}(m, \Sigma)$. The variance of the random variable X is denoted as $\mathrm{Var}(X)$.

2. Gain function approximation.

2.1. Problem formulation. The mathematical problem is to numerically approximate the solution of the Poisson's equation (1.3) introduced in section 1 and also repeated below:

$$\begin{aligned} -\Delta_\rho \phi &= h - \hat{h}_\rho, \\ \text{s.t.} \quad \int \phi(x)\rho(x)\,\mathrm{d}x &= 0, \end{aligned} \tag{2.1}$$

where the weighted Laplacian $\Delta_\rho \phi(x) := \frac{1}{\rho(x)} \nabla \cdot (\rho(x) \nabla \phi(x))$; $\rho(x)$ is an everywhere positive probability density on $\mathbb{R}^d$; $h(x)$ is a real-valued function defined on $\mathbb{R}^d$ and $\hat{h}_\rho := \int h(x)\rho(x)\,\mathrm{d}x$. The function ϕ is referred to as the solution. Its gradient is referred to as the gain function and denoted as $\mathsf{K}(x) := \nabla \phi(x)$. The PDE (2.1) is referred to as the Poisson's equation.

The numerical approximation problem is as follows:

Problem statement: Given N samples $\{X^1, \ldots, X^i, \ldots, X^N\}$, drawn i.i.d.. from ρ, approximate the gains $\{\mathsf{K}^1, \ldots, \mathsf{K}^i, \ldots, \mathsf{K}^N\}$, where $\mathsf{K}^i := \mathsf{K}(X^i) = \nabla \phi(X^i)$. The density ρ is not known in an explicit form.

2.2. Mathematical preliminaries. Assumptions: The following assumptions are made throughout the paper:

(i) **Assumption A1:** The probability density ρ is of the form $\rho(x) = e^{-V(x)}$ where the function $V(x) = \frac{1}{2}(x-m)^\top \Sigma^{-1}(x-m) + w(x)$ for some $m \in \mathbb{R}^d$, $\Sigma \in S^d_{++}$, and $w \in C^\infty_b(\mathbb{R}^d)$;

(ii) **Assumption A2:** The function $h : \mathbb{R}^d \to \mathbb{R}$ is (weakly) differentiable with $\|h\|_{L^4(\rho)} < \infty$, $\|\nabla h\|_{L^4(\rho)} < \infty$.

Remark 2.1. Assumption A1 is used to prove the approximation result (Proposition 3.4) and to derive the spectral gap (Proposition 4.2) for the diffusion map approximation first introduced in section 3. In prior literature, a similar assumption has been previously used for studying functional inequalities to obtain Poincaré inequality with a constant that does

not depend on the dimension [62, Ch. 8]. Assumption A1 is restrictive, e.g., a mixture of Gaussians does not satisfy the assumption. Based on numerical experiments, it is conjectured that Assumption A1 can be relaxed. A weaker assumption would be to assume $\rho = \rho_g * w$, the convolution of a Gaussian density ρ_g with a density w that has a compact support. Proving the theoretical results under this weaker assumption is the subject of future work.

2.2.1. Spectral representation. Under Assumption (A1), the weighted Laplacian Δ_ρ has a discrete spectrum with an ordered sequence of eigenvalues $0 = \lambda_0 < \lambda_1 \le \lambda_2 \le \dots$ and associated eigenfunctions $\{e_n\}$ that form a complete orthonormal basis of $L^2(\rho)$ [5, Cor. 4.10.9]. The trivial eigenfunction $e_0(x) = 1$, and for $f \in L^2_0(\rho)$, the spectral representation yields:

$$-\Delta_\rho f = \sum_{m=1}^{\infty} \lambda_m \langle e_m, f\rangle e_m. \tag{2.2}$$

The positivity of the smallest non-trivial eigenvalue ($\lambda_1 > 0$) is referred to as the Poincaré inequality (or the spectral gap condition) [4]. The inequality is equivalently expressed as

$$\int_{\mathbb{R}^d} (f - \hat{f}_\rho)^2 \rho \, \mathrm{d}x \le \frac{1}{\lambda_1} \int_{\mathbb{R}^d} |\nabla f|^2 \rho \, \mathrm{d}x, \qquad \forall\, f \in H^1(\rho),$$

where $\hat{f}_\rho = \int f \rho \, \mathrm{d}x$.

The Poincaré inequality is important to show that the Poisson equation is well-posed and a unique solution exists. The solution to the Poisson equation is defined using the weak formulation.

2.2.2. Weak formulation. A function $\phi \in H^1_0(\rho)$ is said to be a weak solution of (2.1) if

$$\int \nabla\phi(x) \cdot \nabla\psi(x) \rho(x) \, \mathrm{d}x = \int (h(x) - \hat{h}_\rho)\psi(x)\rho(x) \, \mathrm{d}x \qquad \forall\, \psi \in H^1(\rho). \tag{2.3}$$

Equation (2.3) is referred to as the weak-form of the Poisson's equation. The weak-form is expressed succinctly as $\langle \nabla\phi, \nabla\psi\rangle = \langle h - \hat{h}_\rho, \psi\rangle$ where $\langle \cdot, \cdot \rangle$ is the inner-product in $L^2(\rho)$. The existence and uniqueness of the solution to the weak-form of the Poisson equation is stated in the following Proposition.

Proposition 2.2. *[42, Thm. 2.2.] Suppose ρ satisfies Assumption (A1) and h satisfies Assumption (A2). Then there exists a unique function $\phi \in H^1_0(\rho)$ that satisfies the weak-form of the Poisson equation (2.3). The solution satisfies the bound:*

$$\int |\nabla\phi(x)|^2 \rho(x) \, \mathrm{d}x \le \frac{1}{\lambda_1} \int (h(x) - \hat{h}_\rho)^2 \rho(x) \, \mathrm{d}x.$$

Remark 2.3 (Constant gain approximation). The weak formulation (2.3) has led to the Galerkin algorithm presented in the original FPF papers [66]. A special case of the Galerkin solution is the constant gain approximation formula (1.8). The formula is obtained upon choosing the test functions in (2.3) to be the coordinate functions: $\psi_m(x) = x_m$ for $m = 1, 2, \dots, d$. Then,

$$\int \frac{\partial\phi}{\partial x_m}(x) \rho(x) \, \mathrm{d}x = \int (h(x) - \hat{h}_\rho) x_m \rho(x) \, \mathrm{d}x, \quad \text{for} \quad m = 1, \dots, d,$$

which yields the formula (1.8).

The diffusion map-based algorithm presented in this paper is based on the semigroup formulation of the Poisson equation.

2.2.3. Semigroup. Let $\{P_t\}_{t\geq 0}$ be the semigroup associated with the weighted Laplacian Δ_ρ. The semigroup allows for a probabilistic interpretation which is described next. Consider the following reversible Markov process $\{S_t\}_{t\geq 0}$ evolving in $\mathbb{R}^d$:

$$\mathrm{d}S_t = -\nabla V(S_t)\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}B_t,$$

where $V(x) := -\log(\rho(x))$ and $\{B_t\}_{t\geq 0}$ is a standard Weiner process in $\mathbb{R}^d$. Then

$$P_t f(x) = \mathsf{E}[f(S_t)|S_0 = x].$$

It is straightforward to verify that $P_t : L^2(\rho) \to L^2(\rho)$ is symmetric, i.e., $\langle P_t f, g\rangle = \langle f, P_t g\rangle$ for all $f, g \in L^2(\rho)$ and $\rho(x) = e^{-V(x)}$ is its invariant density. The semigroup also admits a kernel representation:

$$P_t f(x) = \sum_{m=1}^{\infty} e^{-t\lambda_m}\langle e_m, f\rangle e_m(x) = \int_{\mathbb{R}^d} \bar{k}_t(x,y) f(y)\rho(y)\,\mathrm{d}y,$$

where $\bar{k}_t(x,y) := \sum_{m=0}^{\infty} e^{-t\lambda_m} e_m(x)e_m(y)$.

The spectral gap implies that $\|P_t\|_{L^2_0(\rho)} = e^{-t\lambda_1} < 1$. Hence, P_t is a strict contraction on $L^2_0(\rho)$. For the special case of Gaussian density, the eigenfunctions are given by the Hermite polynomials. This leads to an explicit formula for the kernel $\bar{k}_t(x,y)$ in the Gaussian case, as described in section SM1.

By definition of the semigroup

$$\frac{\partial P_t f}{\partial t} = P_t \Delta_\rho f.$$

Integrating the equation from time 0 to t yields

$$P_t f - f = \int_0^t P_s \Delta_\rho f\,\mathrm{d}s.$$

Letting $f = \phi$ where ϕ solves the Poisson equation (2.1) concludes the following fixed-point equation for all $t > 0$:

$$\text{(exact fixed-point equation)} \quad \phi = P_t\phi + \int_0^t P_s(h - \hat{h}_\rho)\,\mathrm{d}s. \tag{2.4}$$

Equation (2.4) is referred to as the semigroup form of the Poisson equation (2.1).

The following Proposition shows that the weak form (2.3) and the semigroup form (2.4) are equivalent. The proof appears in the section SM2.

Proposition 2.4. *Suppose ρ satisfies Assumption (A1) and h satisfies Assumption (A2). Then the unique solution $\phi \in H^1_0(\rho)$ to the weak form (2.3) is also the unique solution to the fixed-point equation (2.4).*

The semigroup formulation has led to the diffusion-map based algorithm which is the main focus of the remainder of this paper.

3. Diffusion map-based Algorithm. The diffusion map-based algorithm is based on a numerical approximation of the fixed-point equation (2.4). The main technique is to approximate the semigroup in the following steps:

1. **Diffusion map approximation:** A family of Markov operators $\{T_\epsilon\}_{\epsilon>0}$ are defined as follows:

$$T_\epsilon f(x) := \frac{1}{n_\epsilon(x)} \int_{\mathbb{R}^d} k_\epsilon(x,y) f(y) \rho(y) \, \mathrm{d}y, \tag{3.1}$$

where $n_\epsilon(x) := \int k_\epsilon(x,y)\rho(y)\,\mathrm{d}y$ is the normalization factor,

$$k_\epsilon(x,y) := \frac{g_\epsilon(x,y)}{\sqrt{\int g_\epsilon(x,z)\rho(z)\,\mathrm{d}z}\sqrt{\int g_\epsilon(y,z)\rho(z)\,\mathrm{d}z}},$$

and $g_\epsilon(x,y) := e^{-\frac{|x-y|^2}{4\epsilon}}$ is the Gaussian kernel in $\mathbb{R}$. For small positive values of ϵ, the Markov operator T_ϵ is referred to as the *diffusion map* approximation of the exact semigroup P_ϵ [15, 32]. The precise statement of this approximation is contained in Proposition 3.4. For the special case of Gaussian density, an explicit formula for the diffusion map appears in the section SM1.

2. **Empirical approximation:** The operator T_ϵ is approximated empirically by the family of operators $\{T_\epsilon^{(N)}\}_{\epsilon>0,N\in\mathbb{N}}$ defined as follows:

$$T_\epsilon^{(N)} f(x) := \frac{1}{n_\epsilon^{(N)}(x)} \sum_{j=1}^{N} k_\epsilon^{(N)}(x, X^j) f(X^j), \tag{3.2}$$

where $n_\epsilon^{(N)}(x) := \sum_{i=1}^N k_\epsilon(x, X^i)$ is the normalization factor and

$$k_\epsilon^{(N)}(x,y) := \frac{g_\epsilon(x,y)}{\sqrt{\sum_{j=1}^N g_\epsilon(x,X^j)}\sqrt{\sum_{j=1}^N g_\epsilon(y,X^j)}}.$$

Recall that $X^i \overset{\text{i.i.d.}}{\sim} \rho$ for $i = 1,\dots,N$. So, by law of large numbers (LLN), $T_\epsilon^{(N)} f$ represents an empirical approximation of the diffusion map T_ϵ. The precise statement of the empirical approximation is contained in Proposition 3.5.

3. **Approximation as Markov matrix:** An $N \times N$ Markov matrix T is defined with (i,j)-th element given by

$$\mathsf{T}_{ij} = \frac{1}{n_\epsilon^{(N)}(X^i)} \mathsf{K}_\epsilon^{(N)}(X^i, X^j). \tag{3.3}$$

Finite-dimensional fixed-point equation: Using the three steps described above, the original infinite-dimensional fixed-point equation (2.4) is approximated as a finite dimensional fixed-point equation

$$\Phi = \mathsf{T}\Phi + \epsilon(\mathsf{h} - \pi(h)), \tag{3.4}$$

where $\mathsf{h} := (h(X^1), \ldots, h(X^N))$ is a $N \times 1$ column vector, and $\pi(h) = \sum_{i=1}^N \pi_i h(X^i)$ where the probability vector $\pi_i = \frac{n_\epsilon^{(N)}(X^i)}{\sum_{j=1}^N n_\epsilon^{(N)}(X^j)}$ is the unique stationary distribution of the Markov matrix T. The solution Φ is used to define an approximation to the solution of the Poisson equation as follows:

$$\phi_\epsilon^{(N)}(x) := \frac{1}{n_\epsilon^{(N)}(x)} \sum_{j=1}^N k_\epsilon^{(N)}(x, X^j)\Phi_j + \epsilon(h(x) - \pi(h)). \tag{3.5}$$

The approximation for the gain function is as follows (see also Remark 4.8):

$$\mathsf{K}_\epsilon^{(N)}(x) = \nabla \left[\frac{1}{n_\epsilon^{(N)}(x)} \sum_{j=1}^N k_\epsilon^{(N)}(x, X^j)(\Phi_j + \epsilon \mathsf{h}_j) \right]. \tag{3.6}$$

Upon evaluating the gradient in closed-form, the following linear formula results for the gain function evaluated at particle locations:

$$\mathsf{K}^i := \mathsf{K}_\epsilon^{(N)}(X^i) = \sum_{j=1}^N s_{ij} X^j, \tag{3.7}$$

where

$$s_{ij} := \frac{1}{2\epsilon} \mathsf{T}_{ij}(r_j - \sum_{k=1}^N \mathsf{T}_{ik} r_k), \quad r_j := \Phi_j + \epsilon \mathsf{h}_j. \tag{3.8}$$

The details of the calculation leading to the linear formula appear in the section SM3.

Remark 3.1 (Numerical procedure). The fixed-point problem (3.4) is solved in an iterative manner. The vector Φ is initialized to $\Phi_0 = (0, \ldots, 0) \in \mathbb{R}^N$ and updated according to

$$\Phi_{n+1} = \mathsf{T}\Phi_n + \epsilon(\mathsf{h} - \pi(h)), \tag{3.9}$$

for $n = 1, \ldots, L$ for a finite number of L iterations. The procedure is guaranteed to converge, with a geometric convergence rate, because T is a strict contraction on $L_0^2(\pi)$ (Proposition 4.1-(ii)). The overall algorithm is presented in Algorithm 3.1.

The proposed iterative procedure (3.9) is preferred to other numerical procedures because (i) it is straightforward to implement and does not require matrix inversion; (ii) it may be numerically more efficient than solving a system of N linear equations; and (iii) it allows one to use the solution obtained from the previous filter step, as initialization for the iterative procedure (3.9), resulting in quick convergence – typically in a few iterations. The reason for quick convergence is that the change in the solution of the fixed point equation (3.4) is (typically) small from one filtering step to the next. This is because the change in particle locations is (typically) small for a small choice of time increment.

Algorithm 3.1 diffusion-map based algorithm for gain function approximation

Input: $\{X^i\}_{i=1}^N$, $\{h(X^i)\}_{i=1}^N$, Φ_{prev}, ϵ, L
Output: $\{\mathsf{K}^i\}_{i=1}^N$

Calculate $g_{ij} := e^{-\frac{|X^i - X^j|^2}{4\epsilon}}$ for $i, j = 1$ to N
Calculate $k_{ij} := \frac{g_{ij}}{\sqrt{\sum_l g_{il}}\sqrt{\sum_l g_{jl}}}$ for $i, j = 1$ to N
Calculate $d_i = \sum_j k_{ij}$ for $i = 1$ to N
Calculate $\mathsf{T}_{ij} := \frac{k_{ij}}{d_i}$ for $i, j = 1$ to N
Calculate $\pi_i = \frac{d_i}{\sum_j d_j}$ for $i = 1$ to N
Calculate $\hat{\mathsf{h}} = \sum_{i=1}^N \pi_j h(X^i)$
Initialize $\Phi = \Phi_{\text{prev}}$
for $t = 1$ to L **do**
    $\Phi_i = \sum_{j=1}^N \mathsf{T}_{ij}\Phi_j + \epsilon(\mathsf{h} - \hat{\mathsf{h}})$ for $i = 1$ to N
end for
Calculate $r_i = \Phi_i + \epsilon \mathsf{h}_i$ for $i = 1$ to N
Calculate $s_{ij} = \frac{1}{2\epsilon}\mathsf{T}_{ij}(r_j - \sum_{k=1}^N \mathsf{T}_{ik} r_k)$ for $i, j = 1$ to N
Calculate $\mathsf{K}^i = \sum_j s_{ij} X^j$ for $i = 1$ to N

Remark 3.2. The computational complexity of the Algorithm 3.1 is $O(N^2)$ because of the need to assemble the $N \times N$ matrix T. The computational complexity may be reduced by employing sub-sampling techniques and exploiting the approximate sparsity structure of the matrix T. It is noted that the matrix can be made sparse, e.g., by setting $g_\epsilon(X^i, X^j) = 0$ whenever $|X^i - X^j|^2 \gg \epsilon$. Compared to the Galerkin algorithm with computational complexity of $O(Nd^3)$, the diffusion-map algorithm is advantageous in high-dimensional problems where $d \gg N$.

Remark 3.3 (Extension to vector-valued observation). Consider the continuous-time filtering problem (1.1a) and (1.1b) with vector-valued observation process $Z_t = (Z_t^1, \dots, Z_t^m)$, observation function $h = (h^1, \dots, h^m)$, and a standard w.p. model for the observation noise W. In this case, the update law (1.2) for the mean-field process of the FPF algorithm takes the form

$$\mathrm{d}\bar{X}_t = a(\bar{X}_t)\,\mathrm{d}t + \mathrm{d}\bar{B}_t + \sum_{j=1}^m \mathsf{K}_t^j(\bar{X}_t) \circ (\,\mathrm{d}Z_t^j - \frac{h^j(\bar{X}_t) + \hat{h}_t^j}{2}\,\mathrm{d}t),$$

where $\hat{h}_t^j = \mathsf{E}[h^j(\bar{X}_t)|\mathcal{Z}_t]$, and $\mathsf{K}_t^j(x) = \nabla\phi^j(x)$ for $j = 1, \dots, m$. The function ϕ^j is the solution of the Poisson equation (1.3) with the function h replaced by the function h^j, for $j = 1, \dots, m$. In order to approximate the gain function with the diffusion map-based algorithm, one first forms the Markov matrix T according to (3.3), and then solves the finite-dimensional fixed-point problem (4.4) for each observation function $(h^1, \dots, h^m)$. The overall computational

cost is of order $O(mN^2)$ scaling linearly with the number of observations.

3.1. Approximation results. The notation $G_\epsilon(f)(x) := \int g_\epsilon(x,y)f(y)\,\mathrm{d}y$ is used to denote the heat semigroup with a Gaussian kernel $g_\epsilon(x,y)$, and

$$U_\epsilon := \frac{1}{2}\log(\frac{G_\epsilon(\rho)}{\rho^2}), \quad U := -\frac{1}{2}\log(\rho), \tag{3.10a}$$

$$W_\epsilon := \frac{1}{\epsilon}\log(e^{U_\epsilon}G_\epsilon(e^{-U_\epsilon})), \quad W := |\nabla U|^2 - \Delta U. \tag{3.10b}$$

The proof of the following proposition appears in section SM5.

Proposition 3.4. *Consider the family of Markov operators $\{T_\epsilon\}_{\epsilon>0}$ defined according to (3.1). Let $n \in \mathbb{N}$, $t \in (0, t_0)$ with $t_0 < \infty$, and $\epsilon = \frac{t}{n}$. Then,*

(i) *The semigroup P_t and the operator T_ϵ^n admit the following representations:*

$$P_t f(x) = e^{U(x)}\mathsf{E}[e^{-\int_0^t W(B_{2s}^x)\,\mathrm{d}s}e^{-U(B_{2t}^x)}f(B_{2t}^x)], \tag{3.11}$$

$$T_\epsilon^n f(x) = e^{U_\epsilon(x)}\mathsf{E}[e^{-\epsilon\sum_{k=0}^{n-1}W_\epsilon(B_{2k\epsilon}^x)}e^{-U_\epsilon(B_{2n\epsilon}^x)}f(B_{2n\epsilon}^x)], \tag{3.12}$$

for all $x \in \mathbb{R}^d$ where B_t^x is the Brownian motion with initial condition $B_0^x = x$.

(ii) *In the asymptotic limit as $\epsilon \to 0$:*

$$U_\epsilon(x) = U(x) + 2\epsilon W(x) + \epsilon\Delta V(x) + \epsilon^2 r_\epsilon^{(1)}(x), \tag{3.13a}$$

$$W_\epsilon(x) = W(x) + \epsilon r_\epsilon^{(2)}(x), \tag{3.13b}$$

where $|r_\epsilon^{(1)}(x)|, |r_\epsilon^{(2)}(x)| = O(|x|^2)$ and $|\nabla r_\epsilon^{(1)}(x)| = O(|x|)$ as $|x| \to \infty$.

(iii) *For all functions f such that $f, \nabla f \in L^4(\rho)$:*

$$\|(T_{\frac{t}{n}}^n - P_t)f\|_{L^2(\rho)} \le \frac{t\sqrt{t}}{n}C(\|f\|_{L^4(\rho)} + \|\nabla f\|_{L^4(\rho)}), \tag{3.14}$$

where the constant C only depends on t_0 and ρ.

The proof of the following proposition appears in section SM8.

Proposition 3.5. *Consider the diffusion map kernel $\{T_\epsilon\}_{\epsilon>0}$, and its empirical approximation $\{T_\epsilon^{(N)}\}_{\epsilon>0,N\in\mathbb{N}}$. Then for any bounded continuous function $f \in C_b(\mathbb{R}^d)$:*

(i) *(Almost sure convergence) For all $x \in \mathbb{R}^d$*

$$\lim_{N\to\infty} T_\epsilon^{(N)}f(x) = T_\epsilon f(x), \quad a.s.$$

(ii) *(Convergence rate) For any $\delta \in (0,1)$, in the asymptotic limit as $N \to \infty$,*

$$\int |T_\epsilon^{(N)}f(x) - T_\epsilon f(x)|^2\rho(x)\,\mathrm{d}x \le O(\frac{\log(\frac{N}{\delta})}{N\epsilon^d}), \tag{3.15}$$

with probability higher than $1-\delta$.

Remark 3.6. The key idea in the proof of the Proposition 3.4 is the Feynman-Kac representation of the semigroup (3.11). To the best of our knowledge, this representation has not been used before in the analysis of the diffusion map approximation. Most of the existing results concerning the convergence of the diffusion map are based on a Taylor series expansion that would lead to a convergence of the form $\lim_{\epsilon\to 0} \frac{f(x)-T_\epsilon f(x)}{\epsilon} = \Delta_\rho f(x)$ for each $x \in \mathbb{R}^d$ [32, 15, 28]. Convergence results of the form $\lim_{n\to\infty} \|T_{\frac{t}{n}}^n f - P_t f\|_{L^2(\rho)} = 0$ appear in [15, 61], based on functional analytic arguments. The Taylor series type arguments typically require the distribution to be supported on a compact manifold which is not assumed here.

4. Convergence and error analysis. The analysis of the diffusion-map algorithm involves the consideration of the following four fixed point problems:

$$\phi = P_t\phi + \int_0^t P_s(h - \hat{h}_\rho)\,\mathrm{d}s, \tag{4.1}$$

(exact)

$$\phi_\epsilon = T_\epsilon\phi_\epsilon + \epsilon(h - \hat{h}_{\rho_\epsilon}), \tag{4.2}$$

(diffusion-map approx.)

$$\phi_\epsilon^{(N)} = T_\epsilon^{(N)}\phi_\epsilon^{(N)} + \epsilon(h - \pi(h)), \tag{4.3}$$

(empirical approx.)

$$\mathsf{\Phi} = \mathsf{T}\mathsf{\Phi} + \epsilon(\mathsf{h} - \pi(\mathsf{h})), \tag{4.4}$$

(finite-dim.)

where $\hat{h}_{\rho_\epsilon} := \int h(x)\rho_\epsilon(x)\,\mathrm{d}x$ and $\rho_\epsilon(x) := \frac{n_\epsilon(x)\rho(x)}{\int n_\epsilon(x)\rho(x)\,\mathrm{d}x}$ is the density of the invariant probability distribution associated with the Markov operator T_ϵ.

In practice, the finite-dimensional problem (4.4) is solved. The existence and uniqueness of the solution for this problem is the subject of the following proposition whose proof appears in section SM4.

Proposition 4.1. *Consider the finite-dimensional fixed point equation (4.4). Then almost surely*

(i) T *is a reversible Markov matrix with a unique stationary distribution*

$$\pi_i := \frac{n_\epsilon^{(N)}(X^i)}{\sum_{j=1}^N n_\epsilon^{(N)}(X^j)}, \tag{4.5}$$

for $i = 1, \dots, N$.

(ii) T *is a strict contraction on* $L_0^2(\pi) = \{v \in \mathbb{R}^N; \sum \pi_i v_i = 0\}$. *Hence the fixed point equation (4.4) has a unique solution* $\mathsf{\Phi} \in L_0^2(\pi)$.

(iii) *The (empirical approx.) fixed point equation (4.3) has a unique solution given by (see (3.5))*

$$\phi_\epsilon^{(N)}(x) = \frac{1}{n_\epsilon^{(N)}(x)} \sum_{j=1}^N k_\epsilon^{(N)}(x, X^j)\mathsf{\Phi}_j + \epsilon(h(x) - \pi(h)).$$

Based on the results in Proposition 2.4 and Proposition 4.1, the exact solution ϕ and the numerical solution $\phi_\epsilon^{(N)}$ are both well-defined. The remaining task is to show the convergence of $\phi_\epsilon^{(N)} \to \phi$ as $N \to \infty$ and $\epsilon \to 0$. We break the convergence analysis into two parts, bias and variance:

$$\phi_\epsilon^{(N)} \underset{\text{(variance)}}{\overset{N\uparrow\infty}{\longrightarrow}} \phi_\epsilon \underset{\text{(bias)}}{\overset{\epsilon\downarrow 0}{\longrightarrow}} \phi.$$

Before describing the general result, it is useful to first introduce an example that helps illustrate the bias-variance trade-off in this problem.

4.1. Example - the scalar case. In the scalar case (where $d = 1$), the Poisson equation is:

$$-\frac{1}{\rho(x)}\frac{\mathrm{d}}{\mathrm{d}x}(\rho(x)\frac{\mathrm{d}\phi}{\mathrm{d}x}(x)) = h(x) - \hat{h}.$$

Integrating twice yields the solution explicitly

$$\mathsf{K}_{\text{exact}}(x) = \frac{\mathrm{d}\phi}{\mathrm{d}x}(x) = -\frac{1}{\rho(x)}\int_{-\infty}^{x} \rho(z)(h(z) - \hat{h})\,\mathrm{d}z. \tag{4.6}$$

For the choice of ρ as the sum of two Gaussians $\mathcal{N}(-1, \sigma^2)$ and $\mathcal{N}(+1, \sigma^2)$ with $\sigma^2 = 0.2$ and $h(x) = x$, the solution obtained using (4.6) is depicted in Figure 1 (a). Also depicted is the approximate solution obtained using the diffusion-map algorithm with $N = 200$, for different values of ϵ. The constant gain approximation is evaluated according to the explicit integral formula (1.8). As $\epsilon \to \infty$ the approximate gain converges to the constant gain approximation. As ϵ becomes smaller, the approximation becomes more accurate. However, for very small values of ϵ the approximation is poor due to the variance error.

The bias-variance trade-off while varying the parameter ϵ is depicted in Figure 1 (b). The L^2 error is computed as a Monte-Carlo average:

$$\text{m.s.e} = \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}|\mathsf{K}^{(m)}(X^i) - \mathsf{K}_{\text{exact}}(X^i)|^2. \tag{4.7}$$

Figure 1 (b) depicts the error obtained from averaging over $M = 1000$ simulations as a function of the parameter ϵ. It is observed that for a fixed number of particles N, there is an optimal value of ϵ that minimizes the error.

The vector counterpart of this example appears in subsection 5.1.

4.2. Bias. The analysis of bias has two parts:

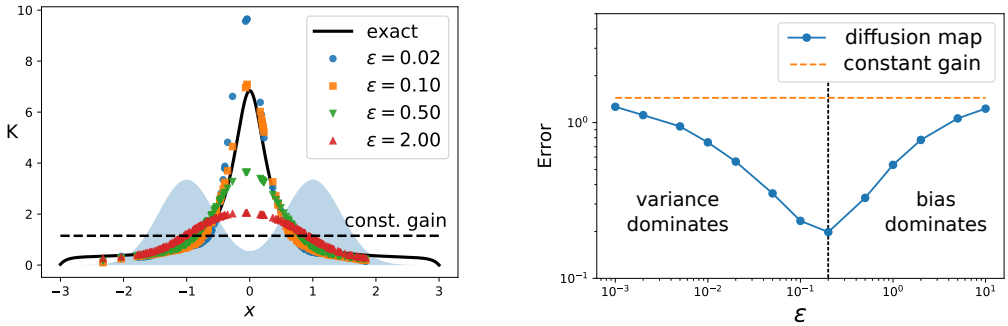
1. To show that the (diffusion-map) fixed-point equation (4.2) admits a unique solution ϕ_ϵ for *all* positive choices of ϵ;
2. To show that $\phi_\epsilon \to \phi$ as $\epsilon \downarrow 0$.

For $n \in \mathbb{N}$, iterate the fixed-point equation (4.2) n times to obtain:

$$\phi_\epsilon = T_\epsilon^n \phi_\epsilon + \sum_{k=0}^{n-1} \epsilon T_\epsilon^k (h - \hat{h}_{\rho_\epsilon}). \tag{4.8}$$

We let $\epsilon = \frac{t}{n}$ for some $t > 0$ and study the solution of this fixed-point equation as $n \to \infty$. Note that the solution to the iterated fixed-point equation (4.8) is identical to the solution to the fixed-point equation (4.2). In the context of equation (4.8), the parameter $\epsilon = \frac{t}{n}$ is interpreted as a small time step-size.

The fixed-point equation (4.8) is the (discrete) Poisson equation that appears in the theory of Markov chain simulation [29, 46] and stochastic control [45, Ch. 9]. Theory presented in



(a) (b)

Figure 1. *Simulation results for the diffusion-map algorithm for the scalar bimodal example: (a) Approximate gain function for different choices of ϵ compared to the exact gain function (solid line). The shaded area in the background is the bimodal probability density function ρ. The dashed line is the constant gain approximation solution; (b) Gain function approximation error of the diffusion-map algorithm as a function of the parameter ϵ. All the results are with $N = 200$ particles.*

these references illustrates how bounds on the solution are obtained under a Foster-Lyapunov drift condition. A similar strategy is adopted here.

In the following proposition, an existence-uniqueness result is described for the fixed-point equation (4.8). The technical step in the proof involves a Foster-Lyapunov condition known as DV(3) [39]. The proof appears in section SM6.

Proposition 4.2. *Consider the family of Markov operators $\{T_\epsilon\}_{\epsilon>0}$ defined in (3.1). Let $n \in \mathbb{N}$, $t \in (0, t_0)$, and $\epsilon = \frac{t}{n}$, with $t_0 < \infty$. Then there exists positive constants a, b, R, δ, a probability measure ν, and a number $n_0 \in \mathbb{N}$ such that for all $n > n_0$:*

$$\log(e^{-U_\epsilon} T_\epsilon^n e^{U_\epsilon}) \leq -atU_\epsilon + bt, \tag{4.9a}$$

$$T_\epsilon^n \mathbb{1}_A(x) \geq \delta \nu(A) \mathbb{1}_{\|x\| \leq R} \quad \forall A \ \in \mathcal{B}(\mathbb{R}^d). \tag{4.9b}$$

Consequently,

(i) *The chain with transition kernel T_ϵ^n is geometrically ergodic with invariant density*

$$\rho_\epsilon(x) := \frac{n_\epsilon(x)\rho(x)}{\int n_\epsilon(x)\rho(x)\,\mathrm{d}x}. \tag{4.10}$$

(ii) *T_ϵ^n is reversible with respect to the density ρ_ϵ It admits a spectral gap as a linear operator $T_\epsilon^n : L_0^2(\rho_\epsilon) \to L_0^2(\rho_\epsilon)$ that is uniform with respect to ϵ. The spectral gap is denoted as λ.*

(iii) *There exists a solution to (4.8) with the bound*

$$\|\phi_\epsilon\|_{L^2(\rho_\epsilon)} \leq \frac{t\|h\|_{L^2(\rho_\epsilon)}}{\lambda}.$$

The proof of the following main result appears in section SM7.

Theorem 4.3. *Suppose the assumptions (A1)-(A2) hold for the density ρ and the function h, and ϕ denotes the exact solution of (4.1). Consider the approximation of this problem defined by the (diffusion-map) fixed-point equation (4.2). For the approximate problem:*

(i) ***Existence-Uniqueness:*** *For each fixed $\epsilon > 0$, there exists a unique solution ϕ_ϵ.*

(ii) ***Convergence:*** *In the asymptotic limit as $\epsilon \to 0$*

$$\|\phi_\epsilon - \phi\|_{L^2(\rho_\epsilon)} \leq O(\epsilon). \tag{4.11}$$

4.3. Variance. The analysis of the variance concerns the (empirical) fixed-point equation (4.3) whose solution is denoted as $\phi_\epsilon^{(N)}$. The parameter ϵ is assumed to be positive and fixed and N is assumed to be finite but large.

The existence-uniqueness of $\phi_\epsilon^{(N)}$ has already been shown as part of Proposition 4.1. The convergence has only been shown below only for the case where the density has a compact support.

Assumption A3: The distribution ρ has compact support given by $\Omega \subset \mathbb{R}^d$.

Theorem 4.4. *Suppose the assumptions (A2)-(A3) hold for the density ρ and the function h, and ϕ_ϵ denotes the solution of the (kernel) fixed-point equation (4.2) for a fixed positive parameter ϵ. Consider the approximation of this problem defined by the (empirical) fixed-point equation (4.3). For the approximate problem:*

(i) ***Existence-Uniqueness:*** *For each finite N, there exists (almost surely) a unique solution $\phi_\epsilon^{(N)}$.*

(ii) ***Convergence:*** *The approximate solution $\phi_\epsilon^{(N)}$ converges to the kernel solution ϕ_ϵ*

$$\lim_{N\to\infty} \|\phi_\epsilon^{(N)} - \phi_\epsilon\|_{L^\infty(\Omega)} = 0, \quad a.s. \tag{4.12}$$

The proof of the convergence $\phi_\epsilon^{(N)} \to \phi_\epsilon$ is based on classical results in the numerical analysis of integral equations on a grid [1, 2]. It relies on the verification of the following three conditions:

(i) The family of operators $\{T_\epsilon^{(N)}\}_{N=1}^\infty$ is collectively compact as linear operators on $C_b(\Omega)$.

(ii) For any function $f \in C_b(\Omega)$,

$$\lim_{N\to\infty} \|T_\epsilon^{(N)} f - T_\epsilon f\|_{L^\infty(\Omega)} = 0, \quad \text{a.s.} \tag{4.13}$$

(iii) The inverse $(I - T_\epsilon)^{-1}$ exists and it is a bounded on $C_0(\Omega) := \{f \in C_b(\Omega); \hat{f}_\rho = 0\}$.

Once these three conditions have been verified, the convergence result (4.12) follows from a standard result in the approximation theory of the numerical solutions of integral equations [34, Thm. 7.6.6]. The proof appears in section SM9.

Remark 4.5 (Convergence rate). The result in Theorem 4.4 establishes asymptotic convergence of the variance error to zero. However, it does not provide an explicit form for the convergence rate. It is possible to obtain an explicit form based upon a convergence rate estimate for the uniform convergence (4.13). The latter is difficult because the existing result in [28] holds only under rather strong regularity conditions on f and assumes that the distribution ρ is uniform.

Based upon the approximation result Proposition 3.5, suppose a convergence rate holds for (4.13) with order $O(\frac{1}{N^{1/2}\epsilon^{d/2}})$. In this case, it is straightforward to derive the following explicit form of the convergence rate for the variance:

$$\|\phi_\epsilon - \phi_\epsilon^{(N)}\|_{L^\infty(\Omega)} \le O(\frac{1}{N^{1/2}\epsilon^{1+d/2}}).$$

The validity and tightness of this bound is studied using numerical experiments in section 5.

Remark 4.6. (Unbounded domain) Analysis of the variance error for the case where the support of ρ is unbounded has proved to be difficult. In the unbounded case, it is more appropriate to consider T_ϵ and $T_\epsilon^{(N)}$ as linear operators on $L^2(\rho)$. Following the same approach as used in the proof of Theorem 4.4, one would need to verify the three conditions noted above. However, for the unbounded case, we could not verify the condition (i) that $\{T_\epsilon^{(N)}\}_{N=1}^\infty$ is collectively compact on $L^2(\rho)$. An alternative approach is to follow the spectral method as outlined in [38]. In this approach, one examines the convergence of empirical matrix $[k(X^i, X^j)]_{i,j=1}^N$ where $k(\cdot,\cdot)$ is a given symmetric kernel. However, this approach does not directly apply to the analysis of the empirical operator $T_\epsilon^{(N)}$. This is because the form of the kernel $k_\epsilon^{(N)}(\cdot,\cdot)$, as it is used in the definition of $T_\epsilon^{(N)}$, is not explicitly given. It too must be empirically approximated as a ratio whose convergence analysis has proved to be rather challenging.

4.4. Relationship to the constant gain approximation. Although the convergence and error analysis pertains to the $\epsilon \downarrow 0$ limit, an important property of the diffusion-map approximation is that the numerical procedure yields a unique solution for arbitrary values of ϵ (see Proposition 4.1). In fact, more can be said: one recovers the constant gain approximation formula in the $\epsilon \to \infty$ limit.

Before stating the result, it is useful to recall the three formulae for the gain:

(i) **Exact formula:** $\mathsf{K} = \nabla\phi$ is defined using the exact solution ϕ;

(ii) **Kernel formula:** K_ϵ is defined using the solution ϕ_ϵ to the (diffusion-map) approximation fixed-point equation:

$$\mathsf{K}_\epsilon(x) := \nabla_x \left[\frac{1}{n_\epsilon(x)} \int k_\epsilon(x,y)(\phi_\epsilon(y) + \epsilon h(y))\rho_\epsilon(y)\,\mathrm{d}y \right]. \tag{4.14}$$

(iii) **Empirical formula:** $\mathsf{K}_\epsilon^{(N)}$ is the empirical version of the kernel formula. It was defined in (3.6) using the solution Φ of the finite-dimensional fixed-point problem.

The proof of the following Proposition appears in the section SM10.

Proposition 4.7. *Consider the fixed-point problems (4.2) and (4.3) in the limit as $\epsilon \to \infty$.*

(i) *The kernel formula of the gain is given by*

$$\lim_{\epsilon\to\infty} \mathsf{K}_\epsilon = \int (h(x) - \hat{h}_\rho) x \rho(x)\,\mathrm{d}x.$$

(ii) *For any finite N, the empirical formula of the gain is given by*

$$\lim_{\epsilon\to\infty} \mathsf{K}_\epsilon^{(N)} = \frac{1}{N}\sum_{i=1}^N (h(X^i) - \hat{h}_\rho^{(N)}) X^i \quad a.s.$$

This result serves to highlight the connection between the FPF and the EnKF: With the diffusion map approximation of the gain, the FPF approaches EnKF in the limit of large ϵ. The parameter ϵ can then be regarded as the tuning parameter to "improve" the gain. Of course, for any finite value of N, this can only be done up to a point – where variance becomes dominant (see Figure 1).

Remark 4.8 (Justification of the approximation (3.4) and (3.6)). The exact fixed-point equation (2.4) may be approximated empirically in terms of $T_\epsilon^{(N)}$ in the following two ways:

$$\phi_\epsilon^{(N)} = T_\epsilon^{(N)}\phi_\epsilon^{(N)} + \epsilon(h - \pi(h)), \tag{4.15a}$$

$$\phi_\epsilon^{(N)} = T_\epsilon^{(N)}\phi_\epsilon^{(N)} + \epsilon T_\epsilon^{(N)}(h - \pi(h)). \tag{4.15b}$$

The (4.15a) and (4.15b) follow from two ways to approximate the integral term

$$\int_0^\epsilon P_s(h - \hat{h}_\rho) \approx \epsilon(h - \hat{h}_\rho),$$
$$\int_0^\epsilon P_s(h - \hat{h}_\rho) \approx \epsilon P_\epsilon(h - \hat{h}_\rho),$$

upon approximating P_ϵ with $T_\epsilon^{(N)}$. In the limit as $\epsilon \to 0$, both (4.15a) and (4.15b) are valid approximations of exact equation (2.4). However, the two approximations exhibit significantly different asymptotic behaviour as $\epsilon \to \infty$. In the limit as $\epsilon \to \infty$, the solutions to (4.15a) and (4.15b) are given by asymptotic formulae

$$\phi_\epsilon^{(N)} = \epsilon(h - \hat{h}) + O(1),$$
$$\phi_\epsilon^{(N)} = O(1),$$

respectively. Similarly, the empirically approximated gain function may be defined in two ways:

$$\mathsf{K}_\epsilon^{(N)} = \nabla\left[T_\epsilon^{(N)}\phi_\epsilon^{(N)} + \epsilon(h - \pi(h))\right], \tag{4.16a}$$

$$\mathsf{K}_\epsilon^{(N)} = \nabla\left[T_\epsilon^{(N)}\phi_\epsilon^{(N)} + \epsilon T_\epsilon^{(N)}(h - \pi(h))\right], \tag{4.16b}$$

where $\phi_\epsilon^{(N)}$ is either given by (4.15a) or (4.15b). The two approximations yield the same limit as $\epsilon \to 0$, however exhibit different asymptotic behaviour as $\epsilon \to \infty$. Given two ways to define $\phi_\epsilon^{(N)}$ according to (4.15a) and (4.15b), and two ways to define $\mathsf{K}_\epsilon^{(N)}$ according to (4.16a) and (4.16b), there are four different asymptotic behaviour for $\mathsf{K}_\epsilon^{(N)}$ as $\epsilon \to \infty$, that are tabulated in Table 1.

In the paper, we choose to define $\phi_\epsilon^{(N)}$ according to (4.15a) and define the gain $\mathsf{K}_\epsilon^{(N)}$ according to (4.16b) . With these choices, the gain converges to the constant gain approximation formula (1.8), in the limit as $\epsilon \to \infty$.

5. Numerics.

$\phi_\epsilon^{(N)}$ def. \ $\mathsf{K}_\epsilon^{(N)}$ def.	(4.16a)	(4.16b)
(4.15a)	$\frac{\epsilon}{2}\nabla h + \frac{1}{2}\mathsf{K}_{\text{cst}}$	K_{cst}
(4.15b)	$\frac{\epsilon}{2}\nabla h$	$\frac{1}{2}\mathsf{K}_{\text{cst}}$

Table 1

Asymptotic behaviour of the gain $\mathsf{K}_\epsilon^{(N)}$ as $\epsilon \to \infty$, corresponding to definitions (4.16a) or (4.16b) of $\mathsf{K}_\epsilon^{(N)}$, where $\phi_\epsilon^{(N)}$ is defined by either (4.15a) or (4.15b). Here K_{cst} denotes the constant gain approximation (1.8).

5.1. Example - the vector case. A vector generalization of the scalar example in subsection 4.1 is obtained by considering the following form of the probability density function in d-dimensions:

$$\rho(x) = \rho_b(x_1) \prod_{n=2}^{d} \rho_g(x_n), \quad \text{for} \quad x = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d,$$

where ρ_b is the bimodal distribution $\frac{1}{2}\mathcal{N}(-1, \sigma^2) + \frac{1}{2}\mathcal{N}(+1, \sigma^2)$ introduced in subsection 4.1, and ρ_g is the Gaussian distribution $\mathcal{N}(0, \sigma^2)$. Also suppose the function $h(x) = x_1$. The simple example is illustrative of realistic application scenarios where the density has non-Gaussian features along certain (not necessarily apriori known) low-dimensional subspace. The directions orthogonal to this subspace are modelled here as Gaussian noise.

For this problem, the exact gain function is easily obtained as

$$\mathsf{K}_{\text{exact}}(x) = (\mathsf{K}_{\text{exact}}(x_1), 0, \ldots, 0),$$

where the function $\mathsf{K}_{\text{exact}}(x_1)$ is given by the formula (4.6) in subsection 4.1. The exact solution is used to compute error properties as dimension increases.

The diffusion-map algorithm (Algorithm 3.1) is simulated to approximate the gain function for this problem. The number of iterations in Algorithm 3.1 set to $L = 10^3$. For each particle $X^i = (X_1^i, \ldots, X_d^i)$, the first coordinate $X_1^i \overset{\text{i.i.d.}}{\sim} \frac{1}{2}\mathcal{N}(-1, \sigma^2) + \frac{1}{2}\mathcal{N}(+1, \sigma^2)$ and other the coordinates $X_n^i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$ for $n = 2, \ldots, d$. The constant gain approximation is evaluated according to the explicit integral formula (1.8).

Figure 2 depicts the m.s.e (4.7) computed from running $M = 100$ simulations. A summary of these results is as follows:

1. Figure 2-(a) depicts the error as a function of the parameters ϵ and d for a fixed number of particles $N = 1000$. Also depicted is the error with the constant gain approximation. The constant gain error serves here as baseline.
   For large values of ϵ, the bias error is dominant, and as $\epsilon \to \infty$ the error asymptotes to the error for the constant-gain approximation. This is because (see Proposition 4.7) the diffusion map gain approaches the constant gain as $\epsilon \to \infty$. For small values of ϵ, the variance error dominates. According to Remark 4.5, the upper-bound for m.s.e is expected te be of the order $O(\frac{1}{N\epsilon^{d+2}})$. However, the numerical error in Figure 2-(a) is observed to be $O(\frac{1}{N\epsilon^{0.16d+0.3}})$. Therefore, the upper-bound in Remark 4.5 is not tight for this specific problem.

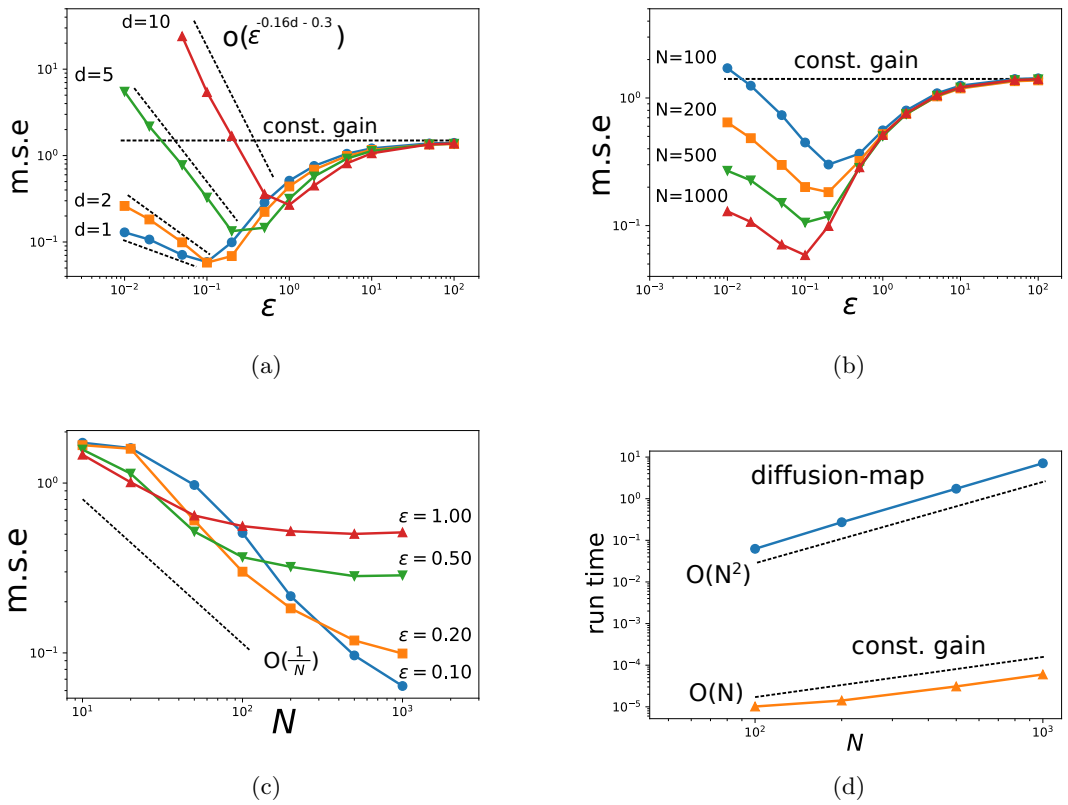


Figure 2. *Simulation results for the diffusion-map algorithm for the vector bimodal example: (a) Gain function approximation error as a function of ϵ for $d \in \{1, 2, 5, 10\}$. (b) Error as a function of ϵ for $N \in \{100, 200, 500, 1000\}$. (c) Error as a function of N for $\epsilon \in \{0.1, 0.2, 0.5, 1.0\}$; (d) Comparison of the run-time.*

2. Figure 2-(b) depicts the bias-variance trade-off as a function of number of particles N for the fixed $d = 1$. It is not a surprise that the error gets better, for all choices of ϵ, as the number of particles increase. However, the optimal value of ϵ – at which the error is the smallest – is relatively insensitive to changes in N.
3. Figure 2-(c) depicts the error as function of N for different values of ϵ. The dimension $d = 1$ is fixed. The error goes down as $O(\frac{1}{N})$ and asymptotes to the $O(\epsilon)$ bias. The $O(\frac{1}{N})$ is due to the variance error obtained in Proposition 3.5 and $O(\epsilon)$ bias error is consistent with the conclusion of the Theorem 4.3.
4. Figure 2-(d) depicts the run time comparison between the diffusion-map algorithm and the constant gain algorithm. The scaling for the diffusion-map algorithm is $O(N^2)$ which is significantly more expensive than the $O(N)$ scaling of the constant gain approximation.

Remark 5.1 (Selection of ϵ). The numerical results in Figure 2 suggest that there is an optimal value of ϵ such that the error is smallest. Given the fact that the constant gain approximation results in the limit as $\epsilon \to \infty$, an optimal choice of ϵ may be possible more

generally. At the optimal value, one optimally trades-off the errors due to variance and bias. The difficulty, of course, is that the formula for this optimal choice is not known and may not even be possible in general settings. Instead, in the literature involving kernel methods, a popular heuristic is to set $\epsilon = \frac{4(\text{med})^2}{\log(N)}$ where (med) is the median value of all pairwise distances $\{|X^i - X^j|\}_{i\neq j}$ [12]. The justification is that, with such a choice, the matrix $[g_\epsilon(X^i, X^j)]_{i,j=1}^N$ is not close to the identity matrix (which represents the degenerate case).

Remark 5.2. It is worthwhile to also examine the limit as $\epsilon \to 0$ while N is fixed at a finite value. In this limit, the Markov matrix T converges to the identity matrix. As a result, the solution Φ to the fixed-point problem (4.4) is unbounded. However, in practice, value of Φ is large but finite, because the equation (4.4) is solved in an iterative manner with finite number of iterations. With a finite value of Φ and T equal to identity, the gain function given by the formula (3.7) is zero. Consequently, the feedback correction for each particle is zero.

5.2. Filtering example. Consider the following filtering problem:

$$\begin{aligned} \mathrm{d}X_t &= 0, \quad X_0 \sim p_0, \\ \mathrm{d}Z_t &= h(X_t)\,\mathrm{d}t + \sigma_w\,\mathrm{d}W_t, \end{aligned}$$

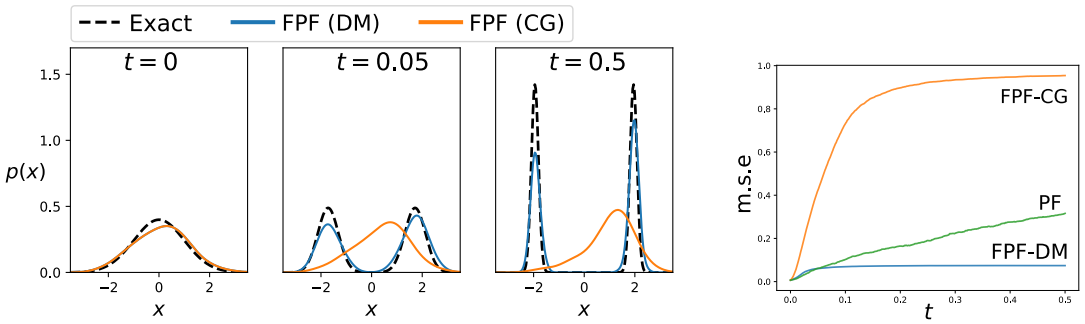
where $X_t \in \mathbb{R}$, $Z_t \in \mathbb{R}$, $\sigma_W > 0$, and $\{W_t\}$ is standard Brownian motion, independent of X_t. The prior distribution p_0 is Gaussian $\mathcal{N}(0,1)$ and the observation function $h(x) = |x|$. For the static filtering problem, the posterior distribution is explicitly given by:

$$p_t^\star(x) = (\text{const.})p_0(x)e^{\frac{1}{\sigma_w^2}(h(x)Z_t - \frac{t}{2}h^2(x))}.$$

Three filtering algorithms are implemented for this problem: (i) the FPF algorithm with the diffusion-map gain approximation; (ii) the FPF algorithm with the the constant gain approximation (similar to EnKF); (iii) a sequential importance resampling (SIR) particle filter [25]. The simulation parameters are as follows: The measurement noise $\sigma_w = 0.1$. The simulation is carried out for $T = 500$ time-steps with step-size $\Delta t = 0.001$. Both the algorithms use $N = 200$ particles with identical initialization. For the diffusion-map approximation, the kernel bandwidth was set to $\epsilon = 0.1$, and number of iterations in Algorithm 3.1 is set to $L = 100$.

The numerical results are depicted in Figure 3. The distribution of the particles along with the exact posterior distribution are depicted in Figure 3-(a). It is observed that the FPF algorithm with the diffusion map approximation provides a more accurate approximation of the posterior distribution. In contrast, the constant-gain approximation fails to reproduce the bimodal nature of the posterior distribution. The fact that the approximated distribution, using the constant-gain approximation, tends to the right is a random artifact due to the initialization of the particles.

A quantitative estimate of the performance is provided in terms of a mean squared error (m.s.e.). in estimating the conditional expectation of the function $\psi(x) = x\mathbb{1}_{x\leq 0}$. A Monte Carlo estimate of the m.s.e. is depicted in Figure 3-(b) with $M = 100$ runs. At time t, it is



(a) (b)

Figure 3. *Simulation results for the FPF algorithm for the filtering example: (a) The distribution of the particles obtained using the diffusion-map approximation and the constant gain approximation as compared to the exact distribution (dashed line); (b) Plot of the mean squared error in estimating the conditional expectation of the function* $\psi(x) = x\mathbb{1}_{x<0}$.

calculated according to

$$\text{m.s.e.}_t = \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{N}\sum_{i=1}^{N}\psi(X_t^{m,i}) - \int \psi(x)p_t^\star(x)\,\mathrm{d}x\right)^2.$$

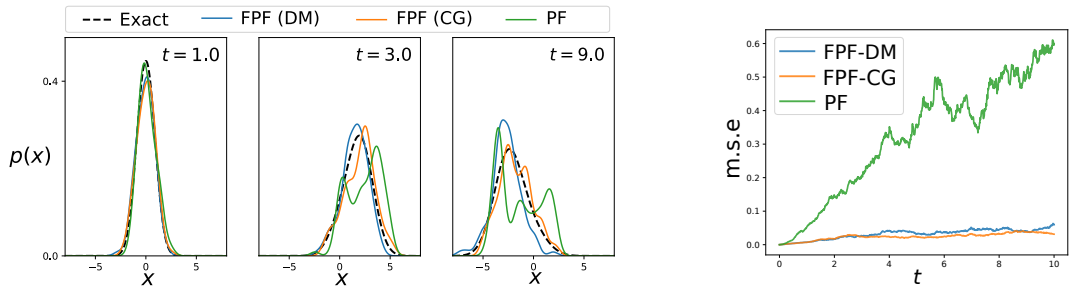
At time $t = 0$, the empirical distribution of the particles is an accurate approximation of the prior distribution, because the particles are sampled i.i.d.. from the prior distribution. Therefore, the m.s.e at $t = 0$ is small. As time progress, the difference between the empirical distribution and the exact posterior becomes larger because the filter update is not exact. For FPF, as the time-step Δt is small, the main source of the m.s.e. error is due to the error in the gain function approximation. Therefore, the diffusion map FPF with its more accurate approximation of the gain yields better m.s.e., compared to the EnKF using the constant gain approximation. The particle filter, like FPF with diffusion map approximation, is able to capture the bi-modal distribution. However, due to the stochastic noise, introduced from the resampling step, it admits larger error.

5.3. Benes filter. Consider the following filtering problem:

$$\begin{aligned}\mathrm{d}X_t &= \mu\sigma_B \tanh\left(\frac{\mu}{\sigma_B}X_t\right)\mathrm{d}t + \sigma_B\,\mathrm{d}B_t, \quad X_0 = x_0,\\ \mathrm{d}Z_t &= (h_1X_t + h_1h_2)\,\mathrm{d}t + \mathrm{d}W_t,\end{aligned}$$

where $\{X_t\}, \{Z_t\} \in \mathbb{R}$ are one-dimensional stochastic processes, $\{B_t\}$ and $\{W_t\}$ are one-dimensional, independent, Brownian motions, x_0 is a known initial condition, and the constants $\mu, \sigma_B, h_1, h_2 \in \mathbb{R}$. This filtering problem has a finite-dimensional analytical solution given by a mixture of two Gaussians [3]:

$$w_t\mathcal{N}(a_t - b_t, \sigma_t^2) + (1 - w_t)\mathcal{N}(a_t + b_t, \sigma_t^2),$$



(a) (b)

Figure 4. *Simulation results for the FPF algorithm for the Benes filter example: (a) The distribution of the particles obtained using the diffusion-map approximation and the constant gain approximation as compared to the exact distribution (dashed line); (b) Plot of the mean squared error in estimating the conditional expectation of the function $\psi(x) = x$.*

where

$$a_t = \sigma_B \Psi_t \tanh(h_1 \sigma_B t) + \frac{h_2 + x_0}{\cosh(h_1 \sigma_B t)} - h_2, \quad b_t = \frac{\mu}{h_1} \tanh(h_1 \sigma_B t),$$

$$\sigma_t^2 = \frac{\sigma_B}{h_1} \tanh(h_1 \sigma_B t), \quad \Psi_t = \int_0^t \frac{\sinh(h_1 \sigma_B s)}{\sinh(h_1 \sigma_B t)} \, \mathrm{d}Z_s, \quad w_t = \frac{1}{1 + e^{\frac{2 a_t b_t}{\sigma_B} \coth(h_1 \sigma_B t)}}.$$

The three filtering algorithms, as in the previous example, are also implemented and evaluated for this problem. The simulation parameters are chosen according to the values used in [16]: $\mu = 0.5$, $h_1 = 0.4$, $h_2 = 0$, $\sigma_B = 0.8$, $x_0 = 1.0$. The simulations are carried out over the time horizon $T = 10$. The stochastic integrals are approximated with a first-order Euler scheme using the discretization step-size $\Delta t = 0.01$. For FPF with DM gain approximation, the kernel bandwidth ϵ is selected according to the rule described in Remark 5.1 and number of iterations in Algorithm 3.1 is $L = 100$.

The numerical results are depicted in Figure 4. It is observed that the FPF with DM and constant gain approximations admit almost the same accuracy. The reason is that the exact bimodal posterior distribution quickly converges to an almost uni-modal distribution. This is because the weight of one of the mixture modes converges to zero. The accuracy of the SIR particle filter is poor because of the stochastic noise introduced from resampling step.

6. Conclusions and Directions for Future Work. In this paper, the diffusion map (DM) algorithm was presented for the problem of gain function approximation in the FPF. It was shown that the approximation error converges to zero in the limit as the number of particles $N \to \infty$ and the kernel bandwidth parameter $\epsilon \to 0$ (Theorems 4.3 and 4.4). In the limit as $\epsilon \to \infty$, the gain obtained using the DM algorithm was shown to converge to the constant gain approximation (Proposition 4.7). Consequently, in this limit, the FPF using the DM algorithm reduces to an EnKF. This is an important property because it suggests a path to improve the performance of an EnKF algorithm by choosing an appropriate (finite) value of the parameter ϵ. The bounds, scalings and the numerical experiments described in this paper

provide guidance on how to choose the parameter ϵ for large but finite N. Some directions for future work are as follows:

1. Relaxing the assumptions: The analysis is based on Assumption A1 which is restrictive because it does not include the mixture of Gaussians. Relaxing this assumption, possibly as suggested in Remark 2.1, is one possible avenue of future work.
2. Error analysis for the FPF: The error analysis in this paper concerns primarily the convergence of function $\phi_\epsilon^{(N)}$ to the exact solution ϕ. Extending these results to include the convergence analysis of the gain $\mathsf{K}_\epsilon^{(N)} = \nabla\phi_\epsilon^{(N)}$ to the exact gain $\mathsf{K} = \nabla\phi$ is important for the complete error analysis of the FPF with finitely many particles.

REFERENCES

[1] P. M. Anselone, *Collectively compact operator approximation theory and applications to integral equations*, Prentice Hall, 1971.

[2] K. Atkinson, *A survey of numerical methods for the solution of Fredholm integral equations of the second kind*, Soc. for Industrial and Applied Mathematics, Philadelphia, PA, 1976, https://cds.cern.ch/record/107092.

[3] A. Bain and D. Crisan, *Fundamentals of stochastic filtering*, vol. 3, Springer, 2009, https://doi.org/10.1007/978-0-387-76896-0.

[4] D. Bakry, F. Barthe, P. Cattiaux, and A. Guillin, *A simple proof of the Poincaré inequality for a large class of probability measures including the log-concave case*, Electron. Commun. Probab, 13 (2008), pp. 60–66, https://doi.org/10.1214/ECP.v13-1352.

[5] D. Bakry, I. Gentil, and M. Ledoux, *Analysis and geometry of Markov diffusion operators*, vol. 348, Springer Science & Business Media, 2013, https://doi.org/10.1007/978-3-319-00227-9_3.

[6] M. Belkin, *Problems of learning on manifolds*, PhD thesis, The University of Chicago, 2003. AAI3097083.

[7] M. Belkin and P. Niyogi, *Convergence of Laplacian eigenmaps*, in Advances in Neural Information Processing Systems, 2007, pp. 129–136, https://doi.org/10.7551/mitpress/7503.003.0021.

[8] K. Bergemann and S. Reich, *An ensemble Kalman-Bucy filter for continuous data assimilation*, Meteorologische Zeitschrift, 21 (2012), pp. 213–219, https://doi.org/10.1127/0941-2948/2012/0307.

[9] K. Berntorp, *Feedback particle filter: Application and evaluation*, in 18th Int. Conf. Information Fusion, Washington, DC, 2015, https://ieeexplore.ieee.org/document/7266752.

[10] K. Berntorp, *Comparison of gain function approximation methods in the feedback particle filter*, in 2018 21st International Conference on Information Fusion (FUSION), IEEE, 2018, pp. 123–130, https://doi.org/10.23919/ICIF.2018.8455574.

[11] K. Berntorp and P. Grover, *Data-driven gain computation in the feedback particle filter*, in 2016 American Control Conference (ACC), 2016, pp. 2711–2716, https://doi.org/10.1109/ACC.2016.7525328.

[12] A. Chaudhuri, D. Kakde, C. Sadek, L. Gonzalez, and S. Kong, *The mean and median criteria for kernel bandwidth selection for support vector data description*, in 2017 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2017, pp. 842–849, https://doi.org/10.1109/ICDMW.2017.116.

[13] Y. Chen, T. Georgiou, and M. Pavon, *Optimal steering of a linear stochastic system to a final probability distribution, part I*, IEEE Trans. Autom. Control, 61 (2016), pp. 1158–1169, https://doi.org/10.1109/TAC.2015.2457784.

[14] Y. Cheng and S. Reich, *A McKean optimal transportation perspective on Feynman-Kac formulae with application to data assimilation*, arXiv preprint arXiv:1311.6300, (2013), https://arxiv.org/abs/1311.6300.

[15] R. R. Coifman and S. Lafon, *Diffusion maps*, Applied and computational harmonic analysis, 21 (2006), pp. 5–30, https://doi.org/10.1016/j.acha.2006.04.006.

[16] D. Crisan and S. Ortiz-Latorre, *A Kusuoka–Lyons–Victoir particle filter*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 469 (2013), p. 20130076, https://doi.org/10.1098/rspa.2013.0076.

[17] D. Crisan and J. Xiong, *Numerical solutions for a class of SPDEs over bounded domains*, ESAIM: Proc., 19 (2007), pp. 121–125, https://doi.org/10.1051/proc:071916.

[18] D. Crisan and J. Xiong, *Approximate McKean-Vlasov representations for a class of SPDEs*, Stochastics, 82 (2010), pp. 53–68, https://doi.org/10.1080/17442500902723575.

[19] F. Daum, J. Huang, and A. Noushin, *Exact particle flow for nonlinear filters*, in SPIE Defense, Security, and Sensing, 2010, pp. 769704–769704, https://doi.org/10.1117/12.839590.

[20] F. Daum, J. Huang, and A. Noushin, *Generalized Gromov method for stochastic particle flow filters*, in SPIE Defense+ Security, International Society for Optics and Photonics, 2017, pp. 102000I–102000I, https://doi.org/10.1117/12.2248723.

[21] J. de Wiljes, S. Reich, and W. Stannat, *Long-time stability and accuracy of the ensemble Kalman–Bucy filter for fully observed processes and small measurement noise*, SIAM Journal on Applied Dynamical Systems, 17 (2018), pp. 1152–1181, https://doi.org/10.1137/17m1119056.

[22] P. Del Moral, *Feynman-Kac formulae*, in Feynman-Kac Formulae, Springer, 2004, pp. 47–93, https://doi.org/10.1007/978-1-4684-9393-1_2.

[23] P. Del Moral, A. Kurtzmann, and J. Tugaut, *On the stability and the uniform propagation of chaos of a class of extended ensemble Kalman–Bucy filters*, SIAM Journal on Control and Optimization, 55 (2017), pp. 119–155, https://doi.org/10.1137/16M1087497.

[24] P. Del Moral and J. Tugaut, *On the stability and the uniform propagation of chaos properties of ensemble KalmanBucy filters*, Ann. Appl. Probab., 28 (2018), pp. 790–850, https://doi.org/10.1214/17-AAP1317, https://doi.org/10.1214/17-AAP1317.

[25] A. M. Doucet, A.and Johansen, *A tutorial on particle filtering and smoothing: Fifteen years later*, Handbook of Nonlinear Filtering, 12 (2009), pp. 656–704, https://www.cs.ubc.ca/~arnaud/doucet_johansen_tutorialPF.pdf.

[26] T. A. El Moselhy and Y. M. Marzouk, *Bayesian inference with optimal maps*, Journal of Computational Physics, 231 (2012), pp. 7815–7850, https://doi.org/10.1016/j.jcp.2012.07.022.

[27] G. Evensen, *Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics*, Journal of Geophysical Research: Oceans, 99 (1994), pp. 10143–10162, https://doi.org/10.1029/94JC00572.

[28] E. Giné, V. Koltchinskii, et al., *Empirical graph Laplacian approximation of Laplace–Beltrami operators: Large sample results*, in High dimensional probability, Institute of Mathematical Statistics, 2006, pp. 238–259, https://doi.org/10.1214/074921706000000888.

[29] P. W. Glynn and S. P. Meyn, *A Liapunov bound for solutions of the Poisson equation*, The Annals of Probability, (1996), pp. 916–931, https://doi.org/10.1214/aop/1039639370.

[30] N. J. Gordon, D. J. Salmond, and A. F. Smith, *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*, in IEE Proceedings F (Radar and Signal Processing), vol. 140, 1993, pp. 107–113, https://doi.org/10.1049/ip-f-2.1993.0015.

[31] M. Hein, J. Audibert, and U. Luxburg, *Graph Laplacians and their convergence on random neighborhood graphs*, J. Mach. Learn. Res., 8 (2007), pp. 1325–1368, https://doi.org/10.1007/11503415_32.

[32] M. Hein, J. Audibert, and U. Von Luxburg, *From graphs to manifolds–weak and strong pointwise consistency of graph Laplacians*, in Learning theory, Springer, 2005, pp. 470–485.

[33] J. Heng, A. Doucet, and Y. Pokern, *Gibbs flow for approximate transport with applications to Bayesian computation*, arXiv preprint arXiv:1509.08787, (2015), https://arxiv.org/abs/1509.08787.

[34] V. Hutson, J. Pym, and M. Cloud, *Applications of functional analysis and operator theory*, vol. 200, Elsevier, 2005.

[35] R. E. Kalman and R. S. Bucy, *New results in linear filtering and prediction theory*, Journal of basic engineering, 83 (1961), pp. 95–108, https://doi.org/10.1115/1.3658902.

[36] E. Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge University Press, 2002, https://doi.org/10.1017/CBO9780511802270.

[37] D. Kelly, K. J. Law, and A. M. Stuart, *Well-posedness and accuracy of the ensemble Kalman filter in discrete and continuous time*, Nonlinearity, 27 (2014), p. 2579, https://doi.org/10.1088/0951-7715/27/10/2579.

[38] V. Koltchinskii and E. Giné, *Random matrix approximation of spectra of integral operators*, Bernoulli, (2000), pp. 113–167.

[39] I. Kontoyiannis, S. P. Meyn, et al., *Large deviations asymptotics and the spectral theory of multi-*

plicatively regular Markov processes, Electron. J. Probab, 10 (2005), pp. 61–123, https://doi.org/10.1214/EJP.v10-231.

[40] H. Kwakernaak and R. Sivan, *Linear optimal control systems*, vol. 1, Wiley-interscience New York, 1972.

[41] E. Kwiatkowski and J. Mandel, *Convergence of the square root ensemble Kalman filter in the large ensemble limit*, SIAM/ASA Journal on Uncertainty Quantification, 3 (2015), pp. 1–17, https://doi.org/10.1137/140965363.

[42] R. S. Laugesen, P. G. Mehta, S. P. Meyn, and M. Raginsky, *Poisson's equation in nonlinear filtering*, SIAM Journal on Control and Optimization, 53 (2015), pp. 501–525, 10.1137/13094743X.

[43] F. Le Gland, V. Monbet, and V. Tran, *Large sample asymptotics for the ensemble Kalman filter*, PhD thesis, INRIA, 2009.

[44] Y. Matsuura, R. Ohata, K. Nakakuki, and R. Hirokawa, *Suboptimal gain functions of feedback particle filter derived from continuation method*, in AIAA Guidance, Navigation, and Control Conference, 2016, p. 1620, https://doi.org/10.2514/6.2016-1620.

[45] S. Meyn, *Control techniques for complex networks*, Cambridge University Press, 2008, https://doi.org/10.1017/CBO9780511804410.

[46] S. Meyn and R. Tweedie, *Markov chains and stochastic stability, cambridge*, 2009, https://doi.org/10.1017/CBO9780511626630.

[47] D. Oliver, A. Reynolds, and N. Liu, *Inverse theory for petroleum reservoir characterization and history matching*, Cambridge University Press, Cambridge, 2008, https://doi.org/10.1017/CBO9780511535642.

[48] A. Radhakrishnan, A. Devraj, and S. Meyn, *Learning techniques for feedback particle filter design*, in Conference on Decision and COntrol (CDC), 2016, IEEE, 2016, pp. 648–653, https://doi.org/10.1109/CDC.2016.7799106.

[49] A. Radhakrishnan and S. Meyn, *Feedback particle filter design using a differential-loss reproducing kernel Hilbert space*, in 2018 Annual American Control Conference (ACC), IEEE, 2018, pp. 329–336, https://doi.org/10.23919/ACC.2018.8431689.

[50] S. Reich, *A dynamical systems framework for intermittent data assimilation*, BIT Numerical Analysis, 51 (2011), pp. 235–249, https://doi.org/10.1007/s10543-010-0302-4.

[51] S. Reich and C. Cotter, *Probabilistic Forecasting and Bayesian Data Assimilation*, Cambridge University Press, Cambridge, UK, 2015.

[52] A. Singer, *From graph to manifold Laplacian: The convergence rate*, Applied and Computational Harmonic Analysis, 21 (2006), pp. 128–134, https://doi.org/10.1016/j.acha.2006.03.004.

[53] P. M. Stano, *Nonlinear State and Parameter Estimation for Hopper Dredgers*, PhD thesis, Ph. D. dissertation). Delft University of Technology, 2013.

[54] P. M. Stano, A. K. Tilton, and R. Babuska, *Estimation of the soil-dependent time-varying parameters of the hopper sedimentation model: The FPF versus the BPF*, Control Engineering Practice, 24 (2014), pp. 67–78, https://doi.org/10.1016/j.conengprac.2013.11.005.

[55] S. C. Surace, A. Kutschireiter, and J.-P. Pfister, *How to avoid the curse of dimensionality: scalability of particle filters with and without importance weights*, Siam Review, 61 (2019), pp. 79–91, https://doi.org/10.1137/17M1125340.

[56] A. Sznitman, *Topics in propagation of chaos*, Ecole d'Eté de Probabilités de Saint-Flour XIX1989, (1991), pp. 165–251, https://doi.org/10.1007/BFb0085169.

[57] A. Taghvaei, J. De Wiljes, P. G. Mehta, and S. Reich, *Kalman filter and its modern extensions for the continuous-time nonlinear filtering problem*, Journal of Dynamic Systems, Measurement, and Control, 140 (2018), p. 030904, https://doi.org/10.1115/1.4037780.

[58] A. Taghvaei and P. G. Mehta, *Gain function approximation in the feedback particle filter*, in Decision and Control (CDC), 2016 IEEE 55th Conference on, IEEE, 2016, pp. 5446–5452, https://doi.org/10.1109/CDC.2016.7799105.

[59] A. Taghvaei and P. G. Mehta, *An optimal transport formulation of the linear feedback particle filter*, in American Control Conference (ACC), 2016, IEEE, 2016, pp. 3614–3619, https://doi.org/10.1109/acc.2016.7525474.

[60] A. Taghvaei, P. G. Mehta, and S. P. Meyn, *Error estimates for the kernel gain function approximation in the feedback particle filter*, in American Control Conference (ACC), 2017, IEEE, 2017, pp. 4576–

4582, https://doi.org/10.23919/ACC.2017.7963661.

[61] D. Ting, L. Huang, and M. Jordan, *An analysis of the convergence of graph Laplacians*, arXiv preprint arXiv:1101.5435, (2011), https://arxiv.org/abs/1101.5435.

[62] C. Villani, *Topics in Optimal Transportation*, vol. 58, American Mathematical Soc., 2003, https://doi.org/10.1090/gsm/058/09.

[63] U. Von Luxburg, *A tutorial on spectral clustering*, Statistics and computing, 17 (2007), pp. 395–416, https://doi.org/10.1007/s11222-007-9033-z.

[64] U. Von Luxburg, M. Belkin, and O. Bousquet, *Consistency of spectral clustering*, The Annals of Statistics, (2008), pp. 555–586, https://doi.org/10.1214/009053607000000640.

[65] J. Xiong, *An introduction to stochastic filtering theory*, vol. 18 of Oxford Graduate Texts in Mathematics, Oxford University Press, 2008.

[66] T. Yang, R. S. Laugesen, P. G. Mehta, and S. P. Meyn, *Multivariable feedback particle filter*, Automatica, 71 (2016), pp. 10–23, https://doi.org/10.1016/j.automatica.2016.04.019.

[67] T. Yang, P. G. Mehta, and S. P. Meyn, *Feedback particle filter*, IEEE Transactions on Automatic Control, 58 (2013), pp. 2465–2480, https://doi.org/10.1109/TAC.2013.2258825.