A CONVERGENT SCHEME FOR THE BAYESIAN FILTERING PROBLEM BASED ON THE FOKKER–PLANCK EQUATION AND DEEP SPLITTING

KASPER BÅGMARK, ADAM ANDERSSON, STIG LARSSON, AND FILIP RYDIN

Abstract. A numerical scheme for approximating the nonlinear filtering density is introduced and its convergence rate is established, theoretically under a parabolic Hörmander condition, and empirically in numerical examples. In a prediction step, between the noisy and partial measurements at discrete times, the scheme approximates the Fokker–Planck equation with a deep splitting scheme, followed by an exact update through Bayes' formula. This results in a classical prediction-update filtering algorithm that operates online for new observation sequences post-training. The algorithm employs a sampling-based Feynman–Kac approach, designed to mitigate the curse of dimensionality. As a corollary we obtain the convergence rate for the approximation of the Fokker–Planck equation alone, disconnected from the filtering problem. The convergence analysis is complemented by a nonlinear 10-dimensional numerical example demonstrating the robustness of the method.

1. Introduction

Approximate nonlinear filters often rely on assumptions of unimodality (e.g., Kalman filters [37]) or computationally cheap tractable simulations. The latter mainly refers to sequential Monte Carlo methods, also known as particle filters. Despite their effectiveness in various settings and their asymptotic convergence to the true filter, these methods suffer from the curse of dimensionality with respect to the state dimension [45, 47, 52, 53]. There is currently a lot of progress within the Bayesian filtering community, e.g., [3], where sampling-based methods have improved substantially [17, 18, 28, 44, 50, 56]. However, the curse of dimensionality remains an open problem for the nonlinear case.

One motivation for developing methods suitable for a high-dimensional setting comes from applications. There exist numerous domains of applications for Bayesian filtering: target tracking [9, 20, 32], finance [22, 55], chemical engineering [48], and weather forecasts [13, 25, 30], to mention a few prominent ones. Many problems are inherently high-dimensional. An extreme example is [2], in which the authors discuss the challenge in weather forecasting with a corresponding dimension of 10^7 that arises from the spatial domain.

To address the challenge of high-dimensional applications, effective methods for Bayesian filtering must be developed to manage the complexity of the state space and accurately estimate the filtering probability distribution. In many settings, it is desirable to obtain the probability density of the filter and not only estimates of mean and covariance, or a finite empirical distribution. We refer to the filtering probability distribution as the probability of observing a hidden state S_k given noisy measurements $y_{0:k}$ up to time k. If such a density p_k exists, it satisfies, for a measurable set C in $\mathbb{R}^d$, the relation

$$\mathbb{P}(S_k \in C \mid y_{0:k}) = \int_C p_k(x \mid y_{0:k}) \, \mathrm{d}x. \tag{1}$$

Moreover, it can be shown that this density adheres to an evolution equation: for discrete observations, it satisfies a Fokker–Planck equation [40], and for continuous observations, it satisfies the Kushner–Stratonovich equation [41]. There are approximation techniques that work well in low dimensions to obtain the filtering density through solving these (Stochastic) Partial Differential Equations (PDEs). These classical techniques, such as finite elements [10] and finite differences, are real-time efficient only in one dimension and stop to be computationally feasible in state dimension $d \geq 4$, which is insufficient for most applications.

2020 *Mathematics Subject Classification.* 60G25, 60G35, 62F15, 62G07, 62M20, 65C30, 65M75, 68T07.

Key words and phrases. Filtering problem, Fokker–Planck equation, partial differential equation, numerical analysis, error estimate, convergence order, Hörmander condition, splitting scheme, deep learning.

In recent years with the explosion of machine learning in applied mathematics, there has been a lot of work going towards efficient deep learning solutions to PDEs. Direct approaches such as Physics-Informed Neural Networks (PINN) [46], deep Backward Stochastic Differential Equation (BSDE) methods [1, 26], deep neural operators [43], deep Ritz method [27], deep Picard iterations [35], and more, have shown excellent performance in handling higher dimensions to different degrees. Recently, there was success with a score-based PINN applied to a high-dimensional Fokker–Planck equation in [36]. However, in general, there are challenges with these methods with respect to modeling probability densities. Specifically, because the Fokker–Planck equation models a probability density function associated with Brownian motion and, since in very high dimensions, its values become extremely small, this poses challenges for existing methods in terms of numerical accuracy.

In this paper, we focus on further development of one such approximation technique, called Deep Splitting. It was demonstrated to perform very well for PDEs with strong symmetries in dimensions up to 10 000 [7], and has been extended to partial integro-differential equations [29]. For the Zakai equation [54], different deep splitting methods have been applied for offline filtering, when the solution is approximated for a single observation sequence [6, 19, 42]. In a subsequent work, deep splitting was combined with an energy-based approach on the Zakai equation and generalized to the online setting where it showed promising results for a state dimension up to $d = 20$ [4]. Here we extend it to a format more interesting for most applications in engineering and science, namely the setting where the underlying process S is continuous in time and space, but the observations O are discrete in time. This is done by propagating the Fokker–Planck equation for the prediction and by updating the solution with Bayes' formula at every measurement. The code for the implementation is publicly available[1], and in a concurrent work [5] we benchmark the method against other deep density methods and classical methods in a high-dimensional setting.

1.1. **Problem formulation.** Let B be a Brownian motion on some complete probability space, $T > 0$ denote a terminal time for the filtering, and $0 = t_0 < t_1 < \cdots < t_K = T$ be observation times. The filtering problem in question deals with time-continuous state and time-discrete observations, where the state and observation models are given by

$$\begin{aligned} \mathrm{d}S_t &= \mu(S_t)\,\mathrm{d}t + \sigma(S_t)\,\mathrm{d}B_t, \quad t \in (0, T], \\ S_0 &\sim q_0, \\ O_k &\sim g(O_k \mid S_{t_k}), \quad k = 0, \dots, K. \end{aligned} \tag{2}$$

In this setting we refer to the $\mathbb{R}^d$-valued process S, solving a Stochastic Differential Equation (SDE) with drift μ and diffusion coefficient σ, as the unknown state process and the $\mathbb{R}^{d'}$-valued variable $O = (O_k)_{k=0}^K$ as the observation process. We stress that (2) is a statistical model, only used in a distributional sense, meaning that pathwise values of S and O are irrelevant and therefore notation for the probability space is not introduced. In any sensible statistical procedure, the distribution of O, determined by the model (2) is approximately that of the data y, as otherwise the filter will output nonsense. The model implicitly defines a measurement likelihood $L(y_k, x) := p(O_k = y_k \mid S_{t_k} = x) = g(y_k \mid x)$, and in Section 2.2 we establish rigorous regularity conditions for the measurement likelihood $L(y, x)$. The standard measurement model, used in almost all publications on filtering with discrete observations, is the Gaussian model with $g(y_k \mid x) = N(y_k; h(x), Q)$, where h is a measurement function and Q is a covariance matrix. The filtering problem, i.e., the problem of obtaining the filtering density (1), assuming the state and observation model (2), and given data y, can be solved by recursively solving a Fokker–Planck equation and by updating using Bayes' formula. To formalize the unnormalized exact filter we initialize $p_0(0, x, y_0) = q_0(x)L(y_0, x)$, and recursively define

$$\begin{aligned} p_k(t, x, y_{0:k}) &= p_k(t_k, x, y_{0:k}) + \int_{t_k}^{t} A^* p_k(s, x, y_{0:k})\,\mathrm{d}s, \quad t \in (t_k, t_{k+1}],\ k = 0, \dots, K-1, \\ p_k(t_k, x, y_{0:k}) &= p_{k-1}(t_k, x, y_{0:k-1})L(y_k, x), \quad k = 1, \dots, K. \end{aligned} \tag{3}$$

Throughout this paper the filter, p_k, is denoted with three arguments (t, x, y), rather than the conventional $(t, x \mid y)$. This should not be confused with the joint distribution over (t, x, y), which we do not address here. The distribution is unnormalized, since the normalizing factor in the Bayes formula has been neglected, but is otherwise a classical Bayesian filtering system of recursive prediction-update steps.

[1] `https://github.com/bagmark/deep-density-filtering`

The operator A^*, that appears in the Fokker–Planck equation in (3), is the adjoint of the infinitesimal generator of the diffusion process S in (2). It remains to solve (3) to define $(p_k)_{k=0}^{K-1}$ including a final update at time t_K to define p_K. In Section 2 we introduce a Feynman–Kac based solution $\widetilde{\pi}$, that approximates the exact filtering density p. The approximation scheme, where $N \in \mathbb{N}$ denotes the number of discretization steps between consecutive observation times, is recursively defined by

$$
\begin{aligned}
(\widetilde{\pi}_{k,n+1}(x,y))_{(x,y)\in\mathbb{R}^d\times\operatorname{supp}(\mathbb{P}_Y)} &= \operatorname*{arg\,min}_{u\in L^\infty(\operatorname{supp}(\mathbb{P}_Y);C(\mathbb{R}^d;\mathbb{R}))} \mathbb{E}\left[\left|u(Z_{N-(n+1)},Y_{0:k}) - G_b\widetilde{\pi}_{k,n}(Z_{N-n},Y_{0:k})\right|^2\right], \\
&\qquad\qquad k=0,\dots,K-1,\ n=0,\dots,N-1, \\
\widetilde{\pi}_{0,0}(x,y_0) &= q_0(x)L(y_0,x), \\
\widetilde{\pi}_{k,0}(x,y_{0:k}) &= \widetilde{\pi}_{k-1,N}(x,y_{0:k-1})L(y_k,x), \quad k=1,\dots,K.
\end{aligned}
$$

Here Z is the Euler–Maruyama approximation of an auxiliary process X. The process X satisfies an SDE with drift b and diffusion σ, driven by a Brownian motion W on $(\Omega,\mathcal{A},\mathbb{P})$. The optimization problem is posed over $L^\infty(\operatorname{supp}(\mathbb{P}_Y);C(\mathbb{R}^d;\mathbb{R}))$, i.e., the space of (essentially) bounded functions on the support of the probability distribution $\mathbb{P}_Y$ with values in $C(\mathbb{R}^d;\mathbb{R})$. Here $C(\mathbb{R}^d;\mathbb{R})$ denotes the space of continuous functions $\mathbb{R}^d \to \mathbb{R}$. The operator G_b is a first order differential operator. By solving the optimization problem and finding $\widetilde{\pi}_k$ for all $k=0,\dots,K$, one has obtained a machine for approximating the unnormalized filtering density (and prediction density) for all possible measurement sequences $y \in \operatorname{supp}(\mathbb{P}_Y) \subset \mathbb{Y} := \mathbb{R}^{d'\times(K+1)}$ with probability distribution $\mathbb{P}_Y$. This makes inference in the online setting very efficient.

The main contribution of this paper is the introduction of this new scheme and its corresponding error analysis. We derive a strong convergence rate in $L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))$ of order 1 in time, assuming a parabolic Hörmander condition as in [14]. The convergence rate is independent of the distribution of the data y. Furthermore, we demonstrate the method on a nonlinear 10-dimensional example to complement the convergence study.

This paper is organized as follows: In Section 2 we present the notation and setting that we use and derive the approximations. In this section we also show regularity and uniqueness of the solution to (3), and of its approximations, and state the main theorem on strong convergence. Section 3 contains the proof of the strong convergence and some auxiliary lemmas that we need. Finally, Section 4 details the additional approximation steps needed to obtain a tractable method based on neural networks and Monte Carlo simulation, and presents the numerical experiments.

2. The online deep splitting filter and main results

2.1. **Notation.** We denote by $\langle x,y\rangle$ and $\|z\|$ the inner product and norm in the Euclidean space $\mathbb{R}^d$ if $x,y,z\in\mathbb{R}^d$ and the Frobenius norm if $z\in\mathbb{R}^{d\times d}$. The space of functions in $[0,T]\times\mathbb{R}^d\to\mathbb{R}$, which are k times continuously differentiable in the first variable and n times continuously differentiable in the second variable with no cross derivatives between the variables, is denoted $C^{k,n}([0,T]\times\mathbb{R}^d;\mathbb{R})$. Furthermore, functions in the space $C_{\mathrm{b}}^{k,n}([0,T]\times\mathbb{R}^d;\mathbb{R})$ have bounded derivatives. The space $C_{\mathrm{p}}^k(\mathbb{R}^d;\mathbb{R})$ consists of functions in $C^k(\mathbb{R}^d;\mathbb{R})$ which, together with their derivatives, are of most polynomial growth. Similarly, functions in the space $C_0^k(\mathbb{R}^d;\mathbb{R}) \subset C^k(\mathbb{R}^d;\mathbb{R})$ are such that they and their derivatives tend to 0 at infinity. For n or $k=\infty$, we adopt the usual modifications. Let $(A,\mathcal{B},\nu)$ be a measure space and U be a Banach space. By $\mathcal{L}^0(A;U)$, we denote the space of strongly measurable functions $f\colon A\to U$ and by $L^0(A;U)$ the equivalence classes of functions in $\mathcal{L}^0(A;U)$ that are equal ν-almost everywhere. The Bochner spaces $L^p(A;U)\subset L^0(A;U)$, $p\in[1,\infty]$, are defined by

$$
\begin{aligned}
\|f\|_{L^p(A;U)} &:= \Big(\int_A \|f(x)\|_U^p \,\mathrm{d}\nu(x)\Big)^{\frac{1}{p}} < \infty, \quad p\in[1,\infty), \\
\|f\|_{L^\infty(A;U)} &:= \sup_{x\in A}\|f(x)\|_U < \infty.
\end{aligned}
$$

Here and throughout the paper, we write sup to mean the essential supremum. For Banach spaces U and V, we denote by $\mathcal{L}(U;V)$ the space of bounded linear operators that map from U to V.

For smooth vector fields $V,W\in C^1(\mathbb{R}^d;\mathbb{R}^d)$ we define the Lie bracket of V and W by

$$
[V,W](x) = \mathrm{D}W(x)V(x) - \mathrm{D}V(x)W(x), \quad x\in\mathbb{R}^d.
$$

Here $\mathrm{D}V(x)$ is the Jacobi matrix of V with respect to $x \in \mathbb{R}^d$. We say that vector fields $V_0, \dots, V_n \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R}^d)$ satisfy a parabolic Hörmander condition if, for all $x \in \mathbb{R}^d$, we have

$$\begin{aligned} \operatorname{span}\Big\{ V_{j_0}(x), \big[V_{j_1}(x), V_{j_2}(x)\big], \Big[\big[V_{j_1}(x), V_{j_2}(x)\big], V_{j_3}(x)\Big], \dots; \\ j_0 \in \{1, \dots, n\},\, j_i \in \{0, \dots, n\} \text{ for all } i = 1, \dots, n \Big\} = \mathbb{R}^d. \end{aligned} \tag{4}$$

To improve clarity, we suppress $x \in \mathbb{R}^d$ in equations throughout the paper whenever appropriate. We define the index set $\mathcal{I}_{K,N} := \{0, \dots, K-1\} \times \{0, \dots, N\} \cup \{K\} \times \{0\}$, for $K, N \in \mathbb{Z}_+$. Finally, if R is a stochastic process, then we let $R^{t,x}$ denote the conditioned process R that starts in x at time t, so that $R_t^{t,x} = x$.

2.2. **Setting.** Throughout the rest of Sections 2 and 3 we assume that the following holds. Let $T > 0$, and $d, d', K \geq 1$ be integers. We consider $K+1$ uniformly distributed observation times, denoted $(t_k)_{k=0}^K$, satisfying

$$0 = t_0 < t_1 < \cdots < t_{K-1} < t_K = T,$$

with $t_{k+1} - t_k = \frac{T}{K}$ for $k = 0, \dots, K-1$. Moreover, for numerical approximations we use a family of finer uniform grids given by

$$t_k = t_{k,0} < t_{k,1} < \cdots < t_{k,N-1} < t_{k,N} = t_{k+1}, \quad k = 0, \dots, K-1,$$

with $t_{k,n+1} - t_{k,n} = \frac{T}{KN}$ for $k = 0, \dots, K-1$, $n = 0, \dots, N-1$. In our error analysis, we write $\tau := \frac{T}{KN}$, where thus K is fixed while N tends to infinity.

The underlying state process $(S_t)_{t\in[0,T]}$ is $\mathbb{R}^d$-valued and, for convenience, we define $\mathbb{Y} := \mathbb{R}^{d' \times (K+1)}$ to denote the space of data measurements. We denote by $y_{k:n}$, $0 \leq k \leq n$, the $d' \times (n-k+1)$-matrix $(y_k, y_{k+1}, \dots, y_n)$.

Throughout the paper, we use a filtered probability space $(\Omega, \mathcal{A}, (\mathcal{F}_t)_{t\in[0,T]}, \mathbb{P})$ equipped with a Brownian motion W adapted to $(\mathcal{F}_t)_{t\in[0,T]}$.

The assumptions on the functions in the statistical model (2) are listed next.

(i) The coefficients μ and σ, initial density q_0, and measurement likelihood L are bounded, infinitely smooth and with bounded derivatives, i.e., $\mu \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R}^d)$, $\sigma \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R}^{d\times d})$, $q_0 \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$, $L \in C_{\mathrm{b}}^\infty(\mathbb{R}^{d'} \times \mathbb{R}^d; \mathbb{R})$.

(ii) The coefficients μ and $\sigma = [\sigma_1, \dots, \sigma_d]$, where σ_i denotes the i'th column of σ, satisfy the parabolic Hörmander condition, i.e., the vector fields $V_0, \dots, V_d$ defined by

$$V_i = \sigma_i, \quad \text{for } i = 1, \dots, d, \qquad V_0 = \mu + \frac{1}{2}\sum_{j=1}^d \mathrm{D}V_j V_j,$$

satisfy (4) for all $x \in \mathbb{R}^d$.

We remark that it is usual to assume that the coefficients μ and σ have bounded derivatives and at most linear growth. This is sufficient for most of this paper also, but the proof of our convergence result seems to require that the coefficients are bounded.

2.3. **Solution to the filtering problem and deep splitting approximations.** The filtering method considered in this paper is a version of the method in [4, 6] but now adapted to the case of discrete observations. More precisely, we model the state, denoted S, with an SDE and the observation process, denoted O, with discrete random variables coupled to discrete time points of S, see (2). Under the conditions of Section 2.2, it is well known from the literature that the SDE in (2) has a unique solution S. That the observation process O is well defined is clear. We remind the reader that the problem under consideration is that of finding the conditional probability distribution of S_t given the measurements $y_{0:k}$, up to and including time t. More precisely, we are for $y \in \mathbb{Y}$ interested in the unnormalized conditional density $p(t, x \mid y_{0:k})$, $t \in [t_k, t_{k+1})$, satisfying for all measurable sets C in $\mathbb{R}^d$ the relation

$$\mathbb{P}(S_t \in C \mid y_{0:k}) = \frac{\int_C p(t, x \mid y_{0:k})\,\mathrm{d}x}{\int_{\mathbb{R}^d} p(t, x \mid y_{0:k})\,\mathrm{d}x}.$$

To this end, we introduce the Fokker–Planck equation. We recall that the state process S satisfying (2) has an associated infinitesimal generator A. This operator and its formal adjoint A^*, are defined, with $a := \sigma\sigma^\top$, for $\varphi \in C_0^\infty(\mathbb{R}^d;\mathbb{R})$, as

$$A\varphi = \frac{1}{2}\sum_{i,j=1}^{d} a_{ij}\,\frac{\partial^2\varphi}{\partial x_i \partial x_j} + \sum_{i=1}^{d}\mu_i\,\frac{\partial\varphi}{\partial x_i} \qquad \text{and} \qquad A^*\varphi = \frac{1}{2}\sum_{i,j=1}^{d}\frac{\partial^2}{\partial x_i\partial x_j}(a_{ij}\varphi) - \sum_{i=1}^{d}\frac{\partial}{\partial x_i}(\mu_i\varphi).$$

Similarly, by letting b define a new drift function, we obtain an alternative generator A_b given by

$$A_b\varphi = \frac{1}{2}\sum_{i,j=1}^{d} a_{ij}\,\frac{\partial^2\varphi}{\partial x_i \partial x_j} + \sum_{i=1}^{d} b_i\,\frac{\partial\varphi}{\partial x_i}.$$

To connect with the framework of the deep splitting method [4, 6, 7], we write $A^* = A_b + F_b$, where the first order differential operator F_b is, for $\varphi \in C^1(\mathbb{R}^d;\mathbb{R})$, defined by

$$F_b\varphi = \sum_{i=1}^{d}\Big(\sum_{j=1}^{d}\frac{\partial a_{ij}}{\partial x_j}\Big)\frac{\partial\varphi}{\partial x_i} + \frac{1}{2}\sum_{i,j=1}^{d}\frac{\partial^2 a_{ij}}{\partial x_i\partial x_j}\,\varphi - \sum_{i=1}^{d}\frac{\partial\mu_i}{\partial x_i}\,\varphi - \sum_{i=1}^{d}\mu_i\,\frac{\partial\varphi}{\partial x_i} - \sum_{i=1}^{d} b_i\,\frac{\partial\varphi}{\partial x_i}.$$

We remark that in the original derivation of the deep splitting method, the drift function b was defined as $b = \mu$, implying $A_b = A$. In our extended framework we allow for different choices satisfying $b \in C_{\mathrm{b}}^\infty$, and in Section 4.2 we discuss numerical impacts of this choice. One choice is

$$b_i = -\mu_i + \sum_{j=1}^{d}\frac{\partial a_{ij}}{\partial x_j}, \quad i = 1,\dots,d, \tag{5}$$

for which the first order terms in F_b disappear. This choice will facilitate our error analysis in Section 3. This is one reason why we assume that the coefficients are bounded; in fact, b as in (5) does not have bounded derivatives unless σ is bounded together with its derivatives.

The recursion (3), that defines $(p_k)_{k=0}^K$, then becomes

$$\begin{aligned} p_k(t,x,y_{0:k}) &= p_k(t_k,x,y_{0:k}) + \int_{t_k}^{t} A_b p_k(s,x,y_{0:k})\,\mathrm{d}s + \int_{t_k}^{t} F_b p_k(s,x,y_{0:k})\,\mathrm{d}s, \quad t \in (t_k,t_{k+1}], \\ &\qquad k = 0,\dots,K-1, \\ p_0(0,x,y_0) &= q_0(x)L(y_0,x), \\ p_k(t_k,x,y_{0:k}) &= p_{k-1}(t_k,x,y_{0:k-1})L(y_k,x), \quad k = 1,\dots,K. \end{aligned} \tag{6}$$

The solution $(p_k)_{k=0}^K$, obtained by solving (6) for every path $y_{0:K} \in \mathbb{Y}$, involves both prediction, when $t \in (t_k,t_{k+1}]$, and filtering, when $t = t_k$. We remark that, since $A_b + F_b = A^*$, the solution of (6) does not depend on the choice of b. The following proposition summarizes the regularity of the solution.

Proposition 2.1. *There exists a unique $p_k \in L^\infty(\mathbb{Y}; C([t_k,t_{k+1}]\times\mathbb{R}^d,\mathbb{R}))$, $k = 0,\dots,K$, satisfying* (6). *Moreover, $p_k(y) \in C_{\mathrm{b}}^{1,\infty}([t_k,t_{k+1}]\times\mathbb{R}^d;\mathbb{R})$ for all $k = 0,\dots,K$, $y \in \mathbb{Y}$.*

Proof. We fix $y \in \mathbb{Y}$ and begin by noting that if $p_{k-1}(t_k,y_{0:k-1}) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$, then $p_k(t_k,y_{0:k}) = p_{k-1}(t_k,y_{0:k-1})L(y_k) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$, since $C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$ is an algebra. In particular, this holds for $p_0(0,y_0) = q_0L(y_0)$. It remains to prove that the Fokker–Planck equation is regularity preserving and that p_k is continuously differentiable in time. For this purpose it is enough to consider one step and to prove that $p_0(y_0) \in C_{\mathrm{b}}^{1,\infty}([0,t_1]\times\mathbb{R}^d;\mathbb{R})$. The conditions of [14, Theorem 4.3] are satisfied under (i) and (ii) of Section 2.2, which tells us that the probability density of S is infinitely smooth and bounded. More precisely, for all $t \in [0,t_1]$ we have $p_0(t,y_0) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$. It remains to show the time regularity. In [21, Chapter 9] the semigroup $(P(t))_{t\geq 0}$ of bounded linear operators on $C_{\mathrm{b}}(\mathbb{R}^d;\mathbb{R})$ is defined as the solution operator to the Kolmogorov backward equation. Likewise, one can define the adjoint semigroup $R = P^*$ on $C_{\mathrm{b}}(\mathbb{R}^d;\mathbb{R})$, associated with the forward equation (6). From this we have for $t \in [0,t_1]$ that $p_0(t,y_0) = R(t)\phi_0$, where $\phi_0 := p_0(0,y_0) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$. Following [21, Proposition 9.9] one can analogously show that for all $\psi \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$ it holds $A^*R(t)\psi = R(t)A^*\psi$. In particular, this holds for $\psi = \phi_0$ and therefore

$$\frac{\mathrm{d}}{\mathrm{d}t}p_0(t,y_0) = \frac{\mathrm{d}}{\mathrm{d}t}R(t)\phi_0 = A^*R(t)\phi_0 = R(t)A^*\phi_0, \quad t \in [0,t_1].$$

From the regularity of ϕ_0 we have that $A^*\phi_0 \in C_\mathrm{b}^\infty(\mathbb{R}^d;\mathbb{R})$ and thus $\frac{\mathrm{d}}{\mathrm{d}t}p_0(t,y_0) \in C_\mathrm{b}^\infty(\mathbb{R}^d;\mathbb{R})$ for all $t \in [0,t_1]$. This shows that $p_0(y_0) \in C_\mathrm{b}^{1,\infty}([0,t_1]\times\mathbb{R}^d;\mathbb{R})$. Lastly, the uniqueness of $p_0(y_0)$ follows analogously to [21, Theorem 9.11] and uniqueness of $p_0 \in L^\infty(\mathbb{Y};C([0,t_1]\times\mathbb{R}^d,\mathbb{R})$ follows since L is bounded and thus so is $\phi_0 = L(y)q_0$ over $\mathbb{Y}$. □

Remark 2.1. *The solution to* (6) *gives unnormalized densities. Instead, in* [15, 23] *the normalized update step is used, replacing the third row of* (6) *with*

$$p_k(t_k,x,y_{0:k}) = \frac{p_{k-1}(t_k,x,y_{0:k-1})L(y_k,x)}{\int_{\mathbb{R}^d} p_{k-1}(t_k,z,y_{0:k-1})L(y_k,z)\,\mathrm{d}z}.$$

The benefit of directly obtaining a normalized density has to be compared to the additional computational cost of evaluating or approximating this integral. In this work, we consider the unnormalized version of the filtering density, but we note that the same methodology holds for the normalized version. This should be seen as a generalization allowing a more flexible framework where normalization is not necessarily required. In practice, as we discuss in Section 4.2 we perform an approximative normalization for numerical stability.

The scene is now set for applying the deep splitting methodology between each of the measurement updates. However, the derivation in this paper is done in a slightly different way that avoids the explicit splitting equations seen in [4, 7]. This leads to the same approximation scheme in the end, but is beneficial for our error analysis. To prepare for Feynman–Kac representations, used in the the deep splitting scheme and the error analysis, we introduce $X\colon [0,T]\times\Omega\to\mathbb{R}^d$, that for all $t\in[0,T]$, $\mathbb{P}$-a.s., satisfies

$$X_t = X_0 + \int_0^t b(X_s)\,\mathrm{d}s + \int_0^t \sigma(X_s)\,\mathrm{d}W_s, \quad X_0 \sim \widetilde{q}_0.$$

Here $\widetilde{q}_0$ is a probability density with finite moments. A natural choice is $\widetilde{q}_0 = q_0$, but to emphasize that this is not necessary, and since it gives increased flexibility in the resulting optimization problem, we have a general $\widetilde{q}_0$ here. From this point, with a slight abuse of notation, we denote by $(t_n)_{n=0}^N$ the time partition between the first two observation times $t_{0,0}$ and $t_{1,0}$, i.e., $t_n := t_{0,n}$, $n=0,\dots,N$, and the previously defined observation time points $(t_k)_{k=0}^K$ will consistently be denoted by $(t_{k,0})_{k=0}^K$ to clarify the distinction. We do this to simplify the notation in the formulas and the following proofs. We have the following Feynman–Kac representation for the solution p_k. We remark that here it is sufficient that μ and σ have bounded derivatives.

Proposition 2.2 (Feynman–Kac representation formula)**.** *The solutions p_k, $k=0,\dots,K-1$, to* (6)*, in the time points $t_{k,n+1}$, $n=0,\dots,N-1$, and $x\in\mathbb{R}^d$, satisfy*

$$p_k(t_{k,n+1},x,y_{0:k}) = \mathbb{E}\Big[p_k(t_{k,n},X_{t_N-t_n}^{t_N-t_{n+1},x},y_{0:k}) + \int_{t_N-t_{n+1}}^{t_N-t_n} F_b p_k(t_{k,N}-s,X_s^{t_N-t_{n+1},x},y_{0:k})\,\mathrm{d}s\Big]. \tag{7}$$

Proof. The proof follows a standard derivation; see, e.g., [38]. We start by fixing $y\in\mathbb{Y}$. Furthermore, we simplify the notation by omitting the dependence on $y_{0:k}$, where it is possible to do so without confusion. We start by noting that the solution p_k to (6), for $n=0,\dots,N-1$ and $t\in(t_{k,n},t_{k,n+1}]$, satisfies

$$\frac{\partial}{\partial t}p_k(t) = A_b p_k(t) + F_b p_k(t).$$

Reparametrization of time $t\mapsto t_{k,N}-t$, so that $t_{k,N}-t\in(t_{k,n},t_{k,n+1}]$, yields

$$\frac{\partial}{\partial t}p_k(t_{k,N}-t) + A_b p_k(t_{k,N}-t) = -F_b p_k(t_{k,N}-t), \quad t\in[t_{k,N}-t_{k,n+1},t_{k,N}-t_{k,n}). \tag{8}$$

Noting that X has the infinitesimal generator A_b, and the fact that $p_k\in C^{1,2}([t_{k,0},t_{k+1,0}]\times\mathbb{R}^d;\mathbb{R})$, allows us to use Itô's formula for $p_k(t_{k,N}-t,X_t)$. This gives, $\mathbb{P}$-a.s.,

$$\begin{aligned} p_k(t_{k,N}-t,X_t) = p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}}) &+ \int_{t_{k,N}-t_{k,n+1}}^t \big\langle \nabla p_k(t_{k,N}-s,X_s),\sigma(X_s)\,\mathrm{d}W_s\big\rangle \\ &+ \int_{t_{k,N}-t_{k,n+1}}^t \Big(\frac{\partial}{\partial s}p_k(t_{k,N}-s,X_s) + A_b p_k(t_{k,N}-s,X_s)\Big)\,\mathrm{d}s. \end{aligned}$$

Inserting (8) into the third term we get

$$
\begin{aligned}
p_k(t_{k,N}-t,X_t) = p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}}) &+ \int_{t_{k,N}-t_{k,n+1}}^{t} \big\langle \nabla p_k(t_{k,N}-s,X_s),\sigma(X_s)\,\mathrm{d}W_s\big\rangle \\
&- \int_{t_{k,N}-t_{k,n+1}}^{t} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s.
\end{aligned} \tag{9}
$$

We recall that $p_k(t_{k,N}-s,y_{0:k}) \in C_{\mathrm{b}}^2(\mathbb{R}^d;\mathbb{R})$ for all $s \in [t_{k,N}-t_{k,n+1}, t_{k,N}-t_{k,n})$, and by the assumptions in Section 2.2 we have that σ satisfies a linear growth bound and X has finite second moments on $[0,T]$. This guarantees that

$$
\int_{t_{k,N}-t_{k,n+1}}^{t_{k,N}-t_{k,n}} \mathbb{E}\Big[\big\|\sigma(X_s)^\top \nabla p_k(t_{k,N}-s,X_s)\big\|^2\Big]\,\mathrm{d}s < \infty
$$

and hence the Itô integral in (9) is a square integrable martingale with respect to $\mathcal{F}_{t_{k,N}-t_{k,n+1}}$. Taking the conditional expectation in (9) we obtain

$$
\begin{aligned}
&\mathbb{E}\Big[p_k(t_{k,N}-t,X_t) \mid \mathcal{F}_{t_{k,N}-t_{k,n+1}}\Big] \\
&\quad = \mathbb{E}\Big[p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}}) - \int_{t_{k,N}-t_{k,n+1}}^{t} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s \mid \mathcal{F}_{t_{k,N}-t_{k,n+1}}\Big].
\end{aligned}
$$

By reordering the terms and using the fact that $p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}})$ is $\mathcal{F}_{t_{k,N}-t_{k,n+1}}$-measurable, we get

$$
p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}}) = \mathbb{E}\Big[p_k(t_{k,N}-t,X_t) + \int_{t_{k,N}-t_{k,n+1}}^{t} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s \mid \mathcal{F}_{t_{k,N}-t_{k,n+1}}\Big]. \tag{10}
$$

We note that the right endpoint limit in $t \in [t_{k,N}-t_{k,n+1}, t_{k,N}-t_{k,n})$, for $x \in \mathbb{R}^d$, satisfies

$$
p_k(t_{k,N}-t,x) \to p_k(t_{k,n},x) \quad \text{as} \quad t \to (t_{k,N}-t_{k,n}),
$$

and $\mathbb{P}$-a.s.

$$
X_t \to X_{t_{k,N}-t_{k,n}} \quad \text{as} \quad t \to (t_{k,N}-t_{k,n}).
$$

Combining these we see that the L^2-limit of the right hand side of (10) satisfies

$$
\begin{aligned}
\lim_{t\to(t_{k,N}-t_{k,n})} \mathbb{E}\Big[\Big| p_k(t_{k,N}-t,X_t) &+ \int_{t_{k,N}-t_{k,n+1}}^{t} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s \\
&- \Big(p_k(t_{k,n},X_{t_{k,N}-t_{k,n}}) + \int_{t_{k,N}-t_{k,n+1}}^{t_{k,N}-t_{k,n}} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s\Big)\Big|^2\Big] = 0.
\end{aligned} \tag{11}
$$

Inserting the limit from (11) in (10) we obtain

$$
p_k(t_{k,n+1},X_{t_{k,N}-t_{k,n+1}}) = \mathbb{E}\Big[p_k(t_{k,n},X_{t_{k,N}-t_{k,n}}) + \int_{t_{k,N}-t_{k,n+1}}^{t_{k,N}-t_{k,n}} F_b p_k(t_{k,N}-s,X_s)\,\mathrm{d}s \mid \mathcal{F}_{t_{k,N}-t_{k,n+1}}\Big].
$$

Rewriting the conditional expectation with respect to a conditioned process $(X_t^{s,x})_{t\geq s}$ (starting in $x \in \mathbb{R}^d$ at time s), and making the $y_{0:k}$-dependence explicit in the notation, we obtain

$$
p_k(t_{k,n+1},x,y_{0:k}) = \mathbb{E}\Big[p_k(t_{k,n},X_{t_{k,N}-t_{k,n}}^{t_{k,N}-t_{k,n+1},x},y_{0:k}) + \int_{t_{k,N}-t_{k,n+1}}^{t_{k,N}-t_{k,n}} F_b p_k(t_{k,N}-s,X_s^{t_{k,N}-t_{k,n+1},x},y_{0:k})\,\mathrm{d}s\Big].
$$

Here we note that $t_{k,N}-t_{k,n} = t_N - t_n$ for all k and n. The proved identity holds for all $y \in \mathbb{Y}$. This completes the proof. □

In the next step, we consider a forward Euler approximation of the integral term in (7). First, we introduce the first order differential operator G_b acting on $\phi \in C^1(\mathbb{R}^d;\mathbb{R})$ according to

$$
(G_b\phi)(x) = \phi(x) + \tau(F_b\phi)(x), \quad x \in \mathbb{R}^d. \tag{12}
$$

We define the approximations $\pi_{k,n+1}$, $(k,n) \in \mathcal{I}_{K,N-1}$, at each time step $t_{k,n+1}$, of (7) by the recursive formula

$$\begin{aligned} \pi_{k,n+1}(x, y_{0:k}) &= \mathbb{E}\Big[G_b \pi_{k,n}\big(X_{t_N - t_n}^{t_N - t_{n+1}, x}, y_{0:k}\big)\Big], \\ &\qquad x \in \mathbb{R}^d,\ y \in \mathbb{Y},\ k = 0, \dots, K-1,\ n = 0, \dots, N-1, \end{aligned} \tag{13}$$

satisfying

$$\begin{aligned} \pi_{0,0}(x, y_0) &= q_0(x) L(y_0, x), \\ \pi_{k,0}(x, y_{0:k}) &= \pi_{k-1,N}(x, y_{0:k-1}) L(y_k, x), \quad k = 1, \dots, K. \end{aligned} \tag{14}$$

Notice that we have $n = 0$ in the final update for $k = K$. Thus, $\pi_{k,n}$ is defined for $(k,n) \in \mathcal{I}_{K,N} = \{0, \dots, K-1\} \times \{0, \dots, N\} \cup \{K\} \times \{0\}$.

Proposition 2.3. *There exists a unique $\pi_{k,n} \in L^\infty(\mathbb{Y}; C(\mathbb{R}^d; \mathbb{R}))$, $(k,n) \in \mathcal{I}_{K,N}$, satisfying* (13) *and* (14). *Furthermore, $\pi_{k,n}(y) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$ for $(k,n) \in \mathcal{I}_{K,N}$, $y \in \mathbb{Y}$.*

Proof. We fix $y \in \mathbb{Y}$ and suppress it from the notation whenever suitable. The lemma is proved by induction. For the base case, with $k = 0$ and $n = 0$, from the assumption (i) of Section 2.2 we have $\pi_{0,0}(y_0) = q_0 L(y_0) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$. For the induction step we take $k \in \{0, 1, \dots, K\}$, $n \in \{1, \dots, N\}$, and assume that $\pi_{k,n-1} \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$. We use the fact that $\pi_{k,n}(x) = \mathbb{E}[G_b \pi_{k,n-1}(X_{t_N - t_{n-1}}^{t_N - t_n, x})]$, is the Feynman–Kac solution $u \colon [t_{k,n-1}, t_{k,n}] \times \mathbb{R}^d \to \mathbb{R}$, at the left endpoint $t_{k,n-1}$, to the Kolmogorov backward equation

$$\frac{\partial}{\partial t} u(t,x) + A_b u(t,x) = 0, \quad t \in [t_{k,n-1}, t_{k,n}); \quad u(t_{k,n}, x) = (G_b \pi_{k,n-1})(x).$$

As a consequence of the assumption, we have $G_b \pi_{k,n-1} \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$ since μ, b, and σ are infinitely smooth. By [40, Theorem 4.8.6] or [34], $u(t) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$ for all $t \in [t_{k,n-1}, t_{k,n}]$ and this shows that $\pi_{k,n} \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$. For the case $n = 1$ we stress that, by the definition of π, we take $\pi_{k,0} = \pi_{k-1,N} L$ and notice that it belongs to $C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$. The uniqueness of the solution follows by [21, Theorem 9.11]. From the boundedness of L it follows that $\pi_{k,n} \in L^\infty(\mathbb{Y}; C(\mathbb{R}^d; \mathbb{R}))$ and this completes the proof. □

The convergence of this approximation is stated in the following lemma, which we prove in Section 3.3. We remark that, while the solution $(p_k)_{k=0}^K$ to (6) is independent of the auxiliary drift b, the discrete approximations $\pi_{k,n+1}$, $(k,n) \in \mathcal{I}_{K,N-1}$, do depend on b.

Lemma 2.1. *Let p_k, $k = 0, \dots, K$, be the solution to* (6) *and $\pi_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (13) *and* (14). *If b satisfies* (5), *then for $\tau := \frac{T}{KN} \le 1$, there exists $C := C(T, \mu, \sigma, L, K) > 0$, such that*

$$\sup_{(k,n) \in \mathcal{I}_{K,N}} \big\| p_k(t_{k,n}) - \pi_{k,n} \big\|_{L^\infty(\mathbb{Y}; L^\infty(\mathbb{R}^d; \mathbb{R}))} \le C\tau.$$

Since the equation for the process X lacks analytical solution in general, we must consider approximations. On the finer time mesh $t_{k,n}$, $(k,n) \in \mathcal{I}_{K,N-1}$, we define an Euler–Maruyama approximation Z, with $Z_{0,0} \sim \widetilde{q}_0$, by

$$Z_{k,n+1} = Z_{k,n} + b(Z_{k,n})(t_{k,n+1} - t_{k,n}) + \sigma(Z_{k,n})(W_{t_{k,n+1}} - W_{t_{k,n}}). \tag{15}$$

For convenience, we let $Z_n := Z_{0,n}$, $n = 0, \dots, N$. Now, we define the approximations $\overline{\pi}_{k,n+1}$, of $\pi_{k,n+1}$, by replacing X with Z in (13), i.e., by the recursion

$$\begin{aligned} \overline{\pi}_{k,n+1}(x, y_{0:k}) &= \mathbb{E}\Big[G_b \overline{\pi}_{k,n}\big(Z_{N-n}^{t_N - t_{n+1}, x}, y_{0:k}\big)\Big], \\ &\qquad x \in \mathbb{R}^d,\ y \in \mathbb{Y},\ k = 0, \dots, K-1,\ n = 0, \dots, N-1, \end{aligned} \tag{16}$$

satisfying

$$\begin{aligned} \overline{\pi}_{0,0}(x, y_0) &= q_0(x) L(y_0, x), \\ \overline{\pi}_{k,0}(x, y_{0:k}) &= \overline{\pi}_{k-1,N}(x, y_{0:k-1}) L(y_k, x), \quad k = 1, \dots, K. \end{aligned} \tag{17}$$

The following proposition can be proved identically to Proposition 2.3, adapting arguments with Z satisfying an SDE with piecewise constant coefficients on each time interval $[t_{k,n}, t_{k,n+1})$.

Proposition 2.4. *There exists a unique $\overline{\pi}_{k,n} \in L^\infty(\mathbb{Y}; C(\mathbb{R}^d; \mathbb{R}))$, $(k,n) \in \mathcal{I}_{K,N}$, satisfying* (16) *and* (17). *Moreover, $\overline{\pi}_{k,n}(y) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d; \mathbb{R})$ for $(k,n) \in \mathcal{I}_{K,N}$, $y \in \mathbb{Y}$.*

The following lemma states the convergence of this second approximation step, with the proof being addressed in Section 3.3.

Lemma 2.2. *Let $\pi_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (13) *and* (14)*, and $\overline{\pi}_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (16) *and* (17)*. If b satisfies* (5)*, then for $\tau := \frac{T}{KN} \leq 1$, there exists $C := C(T, \mu, \sigma, L, K) > 0$, such that*

$$\sup_{(k,n)\in\mathcal{I}_{K,N}} \left\|\pi_{k,n} - \overline{\pi}_{k,n}\right\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))} \leq C\tau.$$

Finally, we reach our first main result, namely the convergence of the approximation $\overline{\pi}$ of p.

Theorem 2.1. *Let p_k, $k = 0, \dots, K$, be the solution to* (6) *and $\overline{\pi}_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (16) *and* (17)*. If b satisfies* (5)*, then for $\tau := \frac{T}{KN} \leq 1$, there exists $C := C(T, \mu, \sigma, L, K) > 0$, such that*

$$\sup_{(k,n)\in\mathcal{I}_{K,N}} \left\|p_k(t_{k,n}) - \overline{\pi}_{k,n}\right\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))} \leq C\tau.$$

Proof. This follows directly from Lemmas 2.1 and 2.2 together with the triangle inequality. □

2.4. **Optimization based formulation of the deep splitting scheme.** Here we reformulate the conditional expectation, formulated as a conditioned process in the proof of Proposition 2.2, as the optimum from a recursive minimization problem. This will allow us to approximate the solution by means of a neural network and stochastic gradient descent. We discuss this approximation further in Section 4.3, in the context of numerical examples. The first optimization problem defined below gives a solution to the recursion (16) for a fixed $y \in \mathbb{Y}$, and was first introduced in [6] for stochastic PDE trajectories. In [19] a similar approach was introduced for the Fokker–Planck equation. A scheme based on this formulation requires retraining of the neural network for every observation path and works in an offline setting, where one is only interested in the inference of a single path. We prove this proposition below and an alternative proof can be found in [8, Proposition 2.7].

Proposition 2.5. *The functions $\overline{\pi}_{k,n+1}$, $k = 0, \dots, K-1$, $n = 0, \dots, N-1$, defined in* (16) *and* (17) *satisfy the minimization problems*

$$(\overline{\pi}_{k,n+1}(x, y_{0:k}))_{x\in\mathbb{R}^d} = \operatorname*{arg\,min}_{u\in C(\mathbb{R}^d;\mathbb{R})} \mathbb{E}\Big[\big|u(Z_{N-(n+1)}) - G_b\overline{\pi}_{k,n}(Z_{N-n}, y_{0:k})\big|^2\Big], \quad n = 0, \dots, N-1,\ y \in \mathbb{Y}.$$

Proof. We start by fixing $y \in \mathbb{Y}$, $k = 0, \dots, K-1$, $n = 0, \dots, N-1$, and introduce the random variables

$$\Phi_{k,n+1} = \overline{\pi}_{k,n+1}(Z_{N-(n+1)}, y_{0:k}),$$
$$\Xi_{k,n} = G_b\overline{\pi}_{k,n}(Z_{N-n}, y_{0:k}).$$

By (16) we have $\Phi_{k,n+1} = \mathbb{E}[\Xi_{k,n} \mid \mathfrak{S}(Z_{N-(n+1)})]$. and by the interpretation of the conditional expectation as an orthogonal L^2-projection, see [39, Corollary 8.17], we thus have that $\Phi_{k,n+1}$ is the unique minimizer in $L^2(\Omega; \mathfrak{S}(Z_{N-(n+1)}))$ of $\Phi \mapsto \mathbb{E}[|\Phi - \Xi_{k,n}|^2]$. Moreover, we have $\Phi_{k,n+1} = u^*(Z_{N-(n+1)})$, where

$$u^* = \operatorname*{arg\,min}_{u\in\mathcal{L}^0(\mathbb{R}^d;\mathbb{R})} \mathbb{E}\Big[\big|u(Z_{N-(n+1)}) - \Xi_{k,n}\big|^2\Big]. \tag{18}$$

We know that $u^* = \overline{\pi}_{k,n+1}$ from the definition of $\Phi_{k,n+1}$ and conclude that u^* is continuous, since $\overline{\pi}_{k,n+1} \in C(\mathbb{R}^d;\mathbb{R})$ by Proposition 2.4. We can thus minimize over continuous functions in (18). □

We generalize the minimization problem obtained in Proposition 2.5 by making a change of probability measure. Instead of fixing y we consider the expectation with respect to a probability measure $\mathbb{P}_Z \times \mathbb{P}_Y$, where $\mathbb{P}_Z$ is the distribution of (15), and $\mathbb{P}_Y$ can be chosen arbitrarily. However, it is natural to use the measurement model g in the statistical model (2). The obtained solution becomes a deterministic function over $\mathbb{R}^d \times \operatorname{supp}(\mathbb{P}_Y)$, allowing for instantaneous evaluation of new observation sequences, suitable for an online setting. In Section 4 we use neural networks as function approximators and the new optimization problem drops the need of retraining the networks.

Proposition 2.6. *Let $Y \in L^\infty(\Omega;\mathbb{Y})$ be a random variable distributed with $\mathbb{P}_Y$, $\overline{\pi}_{k,n+1}$, $(k,n) \in \mathcal{I}_{K,N-1}$ be the solutions to* (16)–(17) *and the functions $\widetilde{\pi}_{k,n+1}\colon \mathbb{R}^d \times \operatorname{supp}(\mathbb{P}_Y) \to \mathbb{R}$, $(k,n) \in \mathcal{I}_{K,N-1}$, be the solutions to the recursive optimization problems*

$$
\begin{aligned}
(\widetilde{\pi}_{k,n+1}(x,y))_{(x,y)\in\mathbb{R}^d\times\operatorname{supp}(\mathbb{P}_Y)} &= \operatorname*{arg\,min}_{u\in L^\infty(\operatorname{supp}(\mathbb{P}_Y);C(\mathbb{R}^d;\mathbb{R}))} \mathbb{E}\left[\left|u(Z_{N-(n+1)},Y_{0:k}) - G_b\widetilde{\pi}_{k,n}(Z_{N-n},Y_{0:k})\right|^2\right], \\
&\qquad k=0,\dots,K-1,\ n=0,1,\dots,N-1, \qquad (19)\\
\widetilde{\pi}_{0,0}(x,y_0) &= q_0(x)L(y_0,x), \\
\widetilde{\pi}_{k,0}(x,y_{0:k}) &= \widetilde{\pi}_{k-1,N}(x,y_{0:k-1})L(y_k,x), \quad k=1,\dots,K.
\end{aligned}
$$

Then $\widetilde{\pi}$ is well defined and moreover, for all $(k,n) \in \mathcal{I}_{K,N}$, and $y \in \operatorname{supp}(\mathbb{P}_Y)$, we have $\overline{\pi}_{k,n}(y_{0:k}) = \widetilde{\pi}_{k,n}(y_{0:k})$.

Proof. We remind the reader that Y now represents a random variable with distribution $\mathbb{P}_Y$, and hence it is enough to show that $\overline{\pi}_{k,n}(\cdot,Y_{0:k}) = \widetilde{\pi}_{k,n}(\cdot,Y_{0:k})$ in $L^\infty(\Omega;\mathbb{R})$. We proceed with the proof by using an induction argument. The base case $k=0$ and $n=0$ is trivially valid, since $\overline{\pi}_{0,0}(Y_0) = q_0L(Y_0) = \widetilde{\pi}_{0,0}(Y_0)$ by definition, and moreover $x \mapsto q_0(x)L(y,x)$ is continuous for every $y \in \mathbb{Y}$ and $(x,y) \mapsto q_0(x)L(y,x)$ is bounded. For the induction step, we fix $k \in \{0,\dots,K-1\}$, $n \in \{1,\dots,N\}$ and assume that $\widetilde{\pi}_{k,n-1}(Z_{N-(n-1)},Y_{0:k}) = \overline{\pi}_{k,n-1}(Z_{N-(n-1)},Y_{0:k}) \in L^\infty(\Omega;\mathbb{R})$. By definition, for a fixed $Y_{0:k}(\omega) = y_{0:k}$, we have

$$
\overline{\pi}_{k,n}(Z_{N-n},y_{0:k}) = \mathbb{E}\Big[G_b\overline{\pi}_{k,n-1}(Z_{N-(n-1)},y_{0:k}) \mid \mathfrak{S}(Z_{N-n})\Big].
$$

The minimization problem (19) is solved by the conditional expectation

$$
\widetilde{\pi}_{k,n}(Z_{N-n},Y_{0:k}) = \mathbb{E}\Big[G_b\widetilde{\pi}_{k,n-1}(Z_{N-(n-1)},Y_{0:k}) \mid \mathfrak{S}(Z_{N-n},Y_{0:k})\Big].
$$

By the inductive assumption we have

$$
\begin{aligned}
\widetilde{\pi}_{k,n}(Z_{N-n},Y_{0:k}) &= \mathbb{E}\Big[G_b\overline{\pi}_{k,n-1}(Z_{N-(n-1)},Y_{0:k}) \mid \mathfrak{S}(Z_{N-n},Y_{0:k})\Big] \\
&= \mathbb{E}\Big[G_b\overline{\pi}_{k,n-1}(Z_{N-(n-1)},y_{0:k}) \mid \mathfrak{S}(Z_{N-n})\Big]\Big|_{y_{0:k}=Y_{0:k}} \\
&= \overline{\pi}_{k,n}(Z_{N-n},Y_{0:k}).
\end{aligned}
$$

Since $x \mapsto \overline{\pi}_{k,n}(x,y)$ is continuous for all $y \in \mathbb{Y}$, then so is $\widetilde{\pi}_{k,n}$, as we have shown that they coincide for $y \in \operatorname{supp}(\mathbb{P}_Y) \subset \mathbb{Y}$. □

The next result is our second main result. It is a direct consequence of Theorem 2.1 and Proposition 2.6.

Corollary 2.1. Let p_k, $k=0,\dots,K$, be the solution to (6) and $\widetilde{\pi}_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to (19). If b satisfies (5), then for $\tau := \frac{T}{KN} \le 1$, there exists $C := C(T,\mu,\sigma,L,K) > 0$, such that

$$
\sup_{\substack{k=0,\dots,K\\ n=0,\dots,N}} \left\|p_k(t_{k,n}) - \widetilde{\pi}_{k,n}\right\|_{L^\infty(\operatorname{supp}(\mathbb{P}_Y);L^\infty(\mathbb{R}^d;\mathbb{R}))} \le C\tau. \qquad (20)
$$

2.5. **Convergence of deep splitting approximation of the Fokker–Planck equation.** In this section we harvest the fruits of our analysis for the Fokker–Planck equation alone, disconnected from any filtering problem. For this purpose we take $K=1$, denote by a slight abuse of notation, $t_{k,n}$ by t_n and notice that $0 = t_0 < t_1 < \cdots < t_N = T$ with uniform time step $\tau = T/N$. The reader can verify that all our proofs are valid for $L \equiv 1$, corresponding to no update and only prediction. From Theorem 2.1, Proposition 2.4, and Propositions 2.1 and 2.5 we conclude the following theorem.

Theorem 2.2. *Let $p \in C_{\mathrm{b}}^{1,\infty}([0,T]\times\mathbb{R}^d;\mathbb{R})$ be the solution to the Fokker–Planck equation*

$$
p(t) = q_0 + \int_0^t A^*p(s)\,\mathrm{d}s, \quad t \in [0,T],
$$

and $\overline{\pi}_n \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$, $n=0,\dots,N-1$, defined by the minimization problems

$$
(\overline{\pi}_{n+1}(x))_{x\in\mathbb{R}^d} = \operatorname*{arg\,min}_{u\in C(\mathbb{R}^d;\mathbb{R})} \mathbb{E}\left[\left|u(Z_{N-(n+1)}) - G_b\overline{\pi}_n(Z_{N-n})\right|^2\right], \quad n=0,\dots,N-1,
$$

where Z_n, $n = 0, \dots, N$, is the Euler–Maruyama approximation

$$Z_{n+1} = Z_n + b(Z_n)(t_{n+1} - t_n) + \sigma(Z_n)(W_{t_{n+1}} - W_{t_n}), \quad n = 0, \dots, N-1,$$

with $Z_0 \sim \widetilde{q}_0$. If b satisfies (5), then for $\tau := \frac{T}{N} \leq 1$, there exists $C := C(T, \mu, \sigma) > 0$ such that

$$\sup_{n=0,\dots,N} \left\| p(t_n) - \overline{\pi}_n \right\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} \leq C\tau.$$

3. Convergence proofs

This section is devoted to the error analysis of the approximation scheme obtained in Proposition 2.6. Section 3.1 contains some preliminaries used for the analysis. In Section 3.2 we state and prove two auxiliary lemmas which are used in Section 3.3 to prove Lemma 2.1 and Lemma 2.2.

3.1. **Preliminaries for the analysis.** We remind the reader that throughout this section we assume the setting of Section 2.2 and we let b be defined by (5). We recall that this choice of b eliminates the first order term in F_b. For convenience in the later proofs we rewrite the operator F_b by defining $f_0 \colon \mathbb{R}^d \to \mathbb{R}$

$$f_0(x) = \frac{1}{2} \sum_{i,j=1}^{d} \frac{\partial^2 a_{ij}(x)}{\partial x_i \partial x_j} - \sum_{i=1}^{d} \frac{\partial \mu_i(x)}{\partial x_i}.$$

With this function, we can write F_b, acting on a function $\phi \in C^1(\mathbb{R}^d;\mathbb{R})$, in the form

$$(F_b\phi)(x) = f_0(x)\phi(x), \quad x \in \mathbb{R}^d. \tag{21}$$

From Assumption (i) of Section 2.2, Proposition 2.1, and Proposition 2.4, we define bounding constants $C_\phi = \max_{|\alpha|\leq 4} \|\mathrm{D}^\alpha \phi\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}$ for $\phi = f_0$, L, and $\overline{\pi}_{k,n}$, for $(k,n) \in \mathcal{I}_{K,N}$. Since $p_k(y) \in C_{\mathrm{b}}^{1,\infty}([t_{k,0}, t_{k+1,0}] \times \mathbb{R}^d;\mathbb{R}))$ for all $y \in \mathbb{Y}$ and $k = 0, \dots, K-1$, p_k is Lipschitz in time, i.e., there exists a constant $C_p > 0$ such that

$$\left\| p_k(t, y_{0:k}) - p_k(s, y_{0:k}) \right\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} \leq C_p|t-s|, \quad s,t \in [t_{k,0}, t_{k+1,0}], \ \ k = 0, \dots, K-1, \ \ y \in \mathbb{Y}, \tag{22}$$

where we note that $\|\phi\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} = \|\phi\|_{C(\mathbb{R}^d;\mathbb{R})}$ for $\phi \in C_{\mathrm{b}}(\mathbb{R}^d;\mathbb{R})$.

3.2. **Local convergence.** In preparation for the proofs of Lemma 2.1 and Lemma 2.2, we first state and prove two lemmas. All the estimations in the proofs hold uniformly with respect to $y \in \mathbb{Y}$, and when we need to emphasize the spatial dependence we write $p_k(t)(x)$ instead of $p_k(t, x, y)$.

Lemma 3.1. *Let p_k, $k = 0, \dots, K$, be the solution to (6) and $\pi_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to (13) and (14). There exist $C_3, C_4 > 0$, such that for all $(k,n) \in \mathcal{I}_{K,N-1}$, we have*

$$\|p_k(t_{k,n+1}) - \pi_{k,n+1}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))} \leq C_3\tau^2 + (1 + C_4\tau)\|p_k(t_{k,n}) - \pi_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}. \tag{23}$$

Proof. Let $y \in \mathbb{Y}$, and $(k,n) \in \mathcal{I}_{K,N-1}$ be fixed, and let y be hidden in the notation. We begin by recalling, from Proposition 2.2 and (13), the Feynman–Kac representations

$$p_k(t_{k,n+1})(x) = \mathbb{E}\Big[p_k(t_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) + \int_{t_N - t_{n+1}}^{t_N - t_n} F_b p_k(t_{k,N} - s)(X_s^{t_N - t_{n+1},x})\,\mathrm{d}s\Big]$$

and

$$\pi_{k,n+1}(x) = \mathbb{E}\Big[G_b \pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x})\Big] = \mathbb{E}\Big[\pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x}) + \tau F_b \pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x})\Big].$$

Using these expressions and the triangle inequality we get

$$\begin{aligned}\|p_k(t_{k,n+1}) - \pi_{k,n+1}\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} \leq{}& \sup_{x\in\mathbb{R}^d} \left|\mathbb{E}\Big[p_k(t_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) - \pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x})\Big]\right| \\ &+ \sup_{x\in\mathbb{R}^d} \left|\mathbb{E}\Big[\int_{t_N - t_{n+1}}^{t_N - t_n} F_b p_k(t_{k,N} - s)(X_s^{t_N - t_{n+1},x})\,\mathrm{d}s - \tau F_b \pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x})\Big]\right| = \mathrm{I} + \mathrm{II}.\end{aligned}$$

For the first term we use the triangle inequality and take a supremum over $\mathbb{R}^d$ inside the expectation to obtain

$$\begin{aligned}\mathrm{I} &\leq \sup_{x\in\mathbb{R}^d} \mathbb{E}\Big[\Big|p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x}) - \pi_{k,n}(X_{t_N-t_n}^{t_N-t_{n+1},x})\Big|\Big] \leq \sup_{x\in\mathbb{R}^d} \mathbb{E}\Big[\sup_{z\in\mathbb{R}^d}\Big|p_k(t_{k,n})(z) - \pi_{k,n}(z)\Big|\Big] \\ &= \sup_{x\in\mathbb{R}^d}\Big|p_k(t_{k,n})(x) - \pi_{k,n}(x)\Big| \leq \big\|p_k(t_{k,n}) - \pi_{k,n}\big\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.\end{aligned} \tag{24}$$

This fits the form of the recursive bound in Lemma 3.1. The second term is handled by adding and subtracting $F_b p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x})\tau$, and applying the triangle inequality to obtain

$$\begin{aligned}\mathrm{II} \leq \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\int_{t_N-t_{n+1}}^{t_N-t_n} F_b p_k(t_{k,N}-s)(X_s^{t_N-t_{n+1},x})\,\mathrm{d}s - \tau F_b p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x})\Big]\Big| \\ + \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\tau F_b p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x}) - \tau F_b \pi_{k,n}(X_{t_N-t_n}^{t_N-t_{n+1},x})\Big]\Big| = \mathrm{II}_1 + \mathrm{II}_2.\end{aligned}$$

By substituting F_b as in (21), applying Hölder's inequality, and using the bound C_{f_0}, we see that

$$\begin{aligned}\mathrm{II}_2 &= \tau \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[f_0(X_{t_N-t_n}^{t_N-t_{n+1},x})(p_k(t_{k,n}) - \pi_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x})\Big]\Big| \\ &\leq C_{f_0}\tau \sup_{x\in\mathbb{R}^d}\mathbb{E}\Big[\Big|p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x}) - \pi_{k,n}(X_{t_N-t_n}^{t_N-t_{n+1},x})\Big|\Big].\end{aligned}$$

We follow the steps in (24) and obtain

$$\mathrm{II}_2 \leq C_{f_0}\tau\big\|p_k(t_{k,n}) - \pi_{k,n}\big\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.$$

This is of the form required in Lemma 3.1.

It remains to show $\mathrm{II}_1 \leq C\tau^2$. By the linearity of the integral we have

$$\mathrm{II}_1 = \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\int_{t_N-t_{n+1}}^{t_N-t_n}\big(F_b p_k(t_{k,N}-s)(X_s^{t_N-t_{n+1},x}) - F_b p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x})\big)\,\mathrm{d}s\Big]\Big|.$$

By adding and subtracting $F_b p_k(t_{k,n})(X_s^{t_N-t_{n+1},x})$ and applying the triangle inequality we obtain

$$\begin{aligned}\mathrm{II}_1 \leq \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\int_{t_N-t_{n+1}}^{t_N-t_n}\big(F_b p_k(t_{k,N}-s)(X_s^{t_N-t_{n+1},x}) - F_b p_k(t_{k,n})(X_s^{t_N-t_{n+1},x})\big)\,\mathrm{d}s\Big]\Big| \\ + \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\int_{t_N-t_{n+1}}^{t_N-t_n}\big(F_b p_k(t_{k,n})(X_s^{t_N-t_{n+1},x}) - F_b p_k(t_{k,n})(X_{t_N-t_n}^{t_N-t_{n+1},x})\big)\,\mathrm{d}s\Big]\Big| = \mathrm{II}_{1,1} + \mathrm{II}_{1,2}.\end{aligned}$$

We substitute F_b as in (21) and use Fubini–Tonelli theorem to observe that

$$\begin{aligned}\mathrm{II}_{1,1} &= \sup_{x\in\mathbb{R}^d}\Big|\mathbb{E}\Big[\int_{t_N-t_{n+1}}^{t_N-t_n}\big(f_0(X_s^{t_N-t_{n+1},x})(p_k(t_{k,N}-s) - p_k(t_{k,n}))(X_s^{t_N-t_{n+1},x})\big)\,\mathrm{d}s\Big]\Big| \\ &= \sup_{x\in\mathbb{R}^d}\Big|\int_{t_N-t_{n+1}}^{t_N-t_n}\mathbb{E}\Big[f_0(X_s^{t_N-t_{n+1},x})(p_k(t_{k,N}-s) - p_k(t_{k,n}))(X_s^{t_N-t_{n+1},x})\Big]\,\mathrm{d}s\Big|.\end{aligned}$$

The condition of absolute integrability of the integrand to use Fubini–Tonelli can be verified by using the Cauchy–Schwarz inequality, bound on f_0, and Lipschitz property of p. In the next step we use the bound C_{f_0}. Hence, we obtain

$$\mathrm{II}_{1,1} \leq C_{f_0}\sup_{x\in\mathbb{R}^d}\Big(\int_{t_N-t_{n+1}}^{t_N-t_n}\mathbb{E}\Big[\big|(p_k(t_{k,N}-s) - p_k(t_{k,n}))(X_s^{t_N-t_{n+1},x})\big|^2\Big]^{\frac{1}{2}}\,\mathrm{d}s\Big). \tag{25}$$

By a uniform bound and substitution in the time variable we get

$$\begin{aligned}&\int_{t_N-t_{n+1}}^{t_N-t_n}\mathbb{E}\Big[\big|(p_k(t_{k,N}-s) - p_k(t_{k,n}))(X_s^{t_N-t_{n+1},x})\big|^2\Big]^{\frac{1}{2}}\,\mathrm{d}s \\ &\quad\leq \int_{t_N-t_{n+1}}^{t_N-t_n}\big\|p_k(t_{k,N}-s) - p_k(t_{k,n})\big\|_{L^\infty(\mathbb{R}^d;\mathbb{R})}\,\mathrm{d}s = \int_{t_{k,n}}^{t_{k,n+1}}\big\|p_k(s) - p_k(t_{k,n})\big\|_{L^\infty(\mathbb{R}^d;\mathbb{R})}\,\mathrm{d}s.\end{aligned}$$

Recalling the Lipschitz continuity in time (22) we obtain

$$\int_{t_{k,n}}^{t_{k,n+1}} \left\| p_k(s) - p_k(t_{k,n}) \right\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} \mathrm{d}s \leq C_p \int_{t_{k,n}}^{t_{k,n+1}} (s - t_{k,n}) \, \mathrm{d}s = \frac{1}{2} C_p \tau^2.$$

Inserting this in (25) we get

$$\mathrm{II}_{1,1} \leq \frac{1}{2} C_{f_0} C_p \tau^2.$$

For the term $\mathrm{II}_{1,2}$, we apply Itô's formula to obtain

$$\begin{aligned} F_b p_k(t_{k,n})(X_s^{t_N - t_{n+1},x}) - F_b p_k(t_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) = &- \int_s^{t_N - t_n} A_b(F_b p_k(t_{k,n}))(X_r^{t_N - t_{n+1},x}) \, \mathrm{d}r \\ &- \int_s^{t_N - t_n} \langle \nabla (F_b p_k(t_{k,n}))(X_r^{t_N - t_{n+1},x}), \sigma(X_r^{t_N - t_{n+1},x}) \, \mathrm{d}W_r \rangle. \end{aligned} \tag{26}$$

We note that the second term is a square integrable martingale with respect to $\mathcal{F}_s$, since the integrand is uniformly bounded, and hence it vanishes under the conditional expectation. We apply the Fubini–Tonelli theorem to change the order of the conditional expectation and time integral in $\mathrm{II}_{1,2}$, and then insert the right hand side of (26) to get

$$\begin{aligned} \mathrm{II}_{1,2} &= \sup_{x \in \mathbb{R}^d} \left| \int_{t_N - t_{n+1}}^{t_N - t_n} \mathbb{E}\Big[\big(F_b p_k(t_{k,n})(X_s^{t_N - t_{n+1},x}) - F_b p_k(t_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) \big) \Big] \, \mathrm{d}s \right| \\ &= \sup_{x \in \mathbb{R}^d} \left| \int_{t_N - t_{n+1}}^{t_N - t_n} \mathbb{E}\Big[\int_s^{t_N - t_n} A_b(F_b p_k(t_{k,n}))(X_r^{t_N - t_{n+1},x}) \, \mathrm{d}r \Big] \, \mathrm{d}s \right|. \end{aligned}$$

We recall that $p_k(t_{k,n})(y) \in C_{\mathrm{b}}^\infty(\mathbb{R}^d;\mathbb{R})$ uniformly with respect to $y \in \mathbb{Y}$, hence $A_b(F_b p_k(t_{k,n}))$ is bounded by some constant $C > 0$ depending on μ, σ and derivatives of p of order up to two, since F_b is of order zero. We use this bounding constant to obtain

$$\mathrm{II}_{1,2} \leq \frac{1}{2} C \tau^2.$$

Thus, we have shown that $\mathrm{II}_{1,2}$, and hence II_1, is bounded as required. Since the obtained bounds are uniform in y, we can finish by taking supremum over $y \in \mathbb{Y}$ on the left hand side to obtain (23). □

Lemma 3.2. *Let $\pi_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (13) *and* (14)*, and $\overline{\pi}_{k,n}$, $(k,n) \in \mathcal{I}_{K,N}$, be the solution to* (16) *and* (17)*. There exist $C_5, C_6 > 0$, such that for all $(k,n) \in \mathcal{I}_{K,N-1}$, we have*

$$\| \pi_{k,n+1} - \overline{\pi}_{k,n+1} \|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))} \leq C_5 \tau^2 + (1 + C_6 \tau) \| \pi_{k,n} - \overline{\pi}_{k,n} \|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.$$

Proof. We start by fixing $y \in \mathbb{Y}$, $(k,n) \in \mathcal{I}_{K,N-1}$ and show that the convergence rate is independent of y. Recalling that the solution π to (13) and the solution $\overline{\pi}$ to (16) satisfy the recursive relations

$$\begin{aligned} \pi_{k,n+1}(x) &= \mathbb{E}\big[(G_b \pi_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) \big], \\ \overline{\pi}_{k,n+1}(x) &= \mathbb{E}\big[(G_b \overline{\pi}_{k,n})(Z_{N-n}^{t_N - t_{n+1},x}) \big], \end{aligned}$$

we obtain

$$\sup_{x \in \mathbb{R}^d} \left| \pi_{k,n+1}(x) - \overline{\pi}_{k,n+1}(x) \right| = \sup_{x \in \mathbb{R}^d} \left| \mathbb{E}\Big[(G_b \pi_{k,n})\big(X_{t_N - t_n}^{t_N - t_{n+1},x}\big) - (G_b \overline{\pi}_{k,n+1})\big(Z_{N-n}^{t_N - t_{n+1},x}\big) \Big] \right|.$$

We add and subtract $(G_b \overline{\pi}_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x})$ and apply the triangle inequality to get

$$\begin{aligned} \sup_{x \in \mathbb{R}^d} \left| \pi_{k,n+1}(x) - \overline{\pi}_{k,n+1}(x) \right| \leq &\sup_{x \in \mathbb{R}^d} \left| \mathbb{E}\big[(G_b \pi_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) - (G_b \overline{\pi}_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) \big] \right| \\ &+ \sup_{x \in \mathbb{R}^d} \left| \mathbb{E}\big[(G_b \overline{\pi}_{k,n})(X_{t_N - t_n}^{t_N - t_{n+1},x}) - (G_b \overline{\pi}_{k,n})(Z_{N-n}^{t_N - t_{n+1},x}) \big] \right| = \mathrm{I} + \mathrm{II}. \end{aligned}$$

By the definition (12) of G_b and the triangle inequality we obtain

$$\begin{aligned} \mathrm{I} \leq &\sup_{x \in \mathbb{R}^d} \left| \mathbb{E}\big[\pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x}) - \overline{\pi}_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x}) \big] \right| \\ &+ \sup_{x \in \mathbb{R}^d} \left| \mathbb{E}\big[\tau \big(F_b \pi_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x}) - F_b \overline{\pi}_{k,n}(X_{t_N - t_n}^{t_N - t_{n+1},x}) \big) \big] \right| = \mathrm{I}_1 + \mathrm{I}_2. \end{aligned}$$

Applying steps analogous to (24), we obtain

$$\mathrm{I}_1 \leq \sup_{x\in\mathbb{R}^d} \mathbb{E}\Big[\big|\pi_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n}) - \overline{\pi}_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n})\big|\Big] \leq \|\pi_{k,n} - \overline{\pi}_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}. \tag{27}$$

This is one of the desired terms in the bound in Lemma 3.2. For the second term I_2, we want to show

$$\mathrm{I}_2 \leq C\tau\|\pi_{k,n} - \overline{\pi}_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.$$

We proceed by substituting F_b as in (21), applying the Cauchy–Schwarz inequality, and using the bound C_{f_0}. This gives

$$\begin{aligned}\mathrm{I}_2 &\leq \tau \sup_{x\in\mathbb{R}^d} \mathbb{E}\Big[\big|f_0(X^{t_N-t_{n+1},x}_{t_N-t_n})\big|^2\Big]^{\frac{1}{2}} \mathbb{E}\Big[\big|\pi_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n}) - \overline{\pi}_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n})\big|^2\Big]^{\frac{1}{2}} \\ &\leq C_{f_0}\tau \sup_{x\in\mathbb{R}^d} \mathbb{E}\Big[\big|\pi_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n}) - \overline{\pi}_{k,n}(X^{t_N-t_{n+1},x}_{t_N-t_n})\big|^2\Big]^{\frac{1}{2}}.\end{aligned}$$

Finally, using the inequality in (27), we get

$$\mathrm{I}_2 \leq C_{f_0}\tau\|\pi_{k,n} - \overline{\pi}_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.$$

Thus, we have shown

$$I \leq \big(1 + C_{f_0}\tau\big)\|\pi_{k,n} - \overline{\pi}_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}.$$

The term II is the weak error for the Euler–Maruyama approximation of X with test function $G_b\overline{\pi}_{k,n}$. Compared to the standard setting the error is now only over one time step (from $t_N - t_{n+1}$ to $t_N - t_n$) and it is therefore $O(\tau^2)$ instead of $O(\tau)$. We omit the proof which can be found in, e.g., [31, Theorem 5.3.1]. For a constant $C > 0$ that depends on μ, σ and $C_{\overline{\pi}}$ we get

$$\mathrm{II} \leq C\tau^2.$$

We conclude the proof by taking supremum over $y \in \mathbb{Y}$ on the left hand side. □

3.3. **Global convergence.** This section is devoted to proving the strong convergence in Lemma 2.1. The arguments in the following proof can be adapted, using Lemma 3.2, to prove Lemma 2.2.

Proof of Lemma 2.1. We define $e_{k,n} := \|p_k(t_{k,n}) - \pi_{k,n}\|_{L^\infty(\mathbb{Y};L^\infty(\mathbb{R}^d;\mathbb{R}))}$, so that Lemma 3.1 can be summarized, with this notation, as

$$e_{k,n} \leq C_3\tau^2 + (1 + C_4\tau)e_{k,n-1}, \quad k = 0,\dots,K-1,\ n = 1,\dots,N. \tag{28}$$

We next prove a bound at the update time $t_{k,0}$, where we have

$$p_k(t_{k,0}, y_{0:k}) - \pi_{k,0}(y_{0:k}) = \big(p_{k-1}(t_{k,0}, y_{0:k-1}) - \pi_{k-1,N}(y_{0:k-1})\big)L(y_k).$$

Using the fact that $t_{k,0} = t_{k-1,N}$ we get

$$\|p_k(t_{k,0}, y_{0:k}) - \pi_{k,0}(y_{0:k})\|_{L^\infty(\mathbb{R}^d;\mathbb{R})} \leq \|L(y_k)\|_{L^\infty(\mathbb{R}^d;\mathbb{R})}\|p_{k-1}(t_{k-1,N}, y_{0:k-1}) - \pi_{k-1,N}(y_{0:k-1})\|_{L^\infty(\mathbb{R}^d;\mathbb{R})}.$$

We take supremum over $\mathbb{Y}$ and use the uniform bound on L, to obtain

$$e_{k,0} \leq C_L e_{k-1,N}. \tag{29}$$

Inequalities (28) and (29) are next used to complete the proof and we begin by fixing $(k,n) \in \mathcal{I}_{K,N}$. Repeating (28) over n steps, we get

$$e_{k,n} \leq C_3\tau^2 \sum_{\ell=0}^{n-1}(1 + C_4\tau)^\ell + (1 + C_4\tau)^n e_{k,0}. \tag{30}$$

By recalling $\tau = \frac{T}{KN}$, we note that, for $N \geq 1$,

$$(1 + C_4\tau)^{N-1} \leq \exp(C_4TK^{-1}). \tag{31}$$

Applying the formula for geometric sums and (31), we obtain

$$\begin{aligned}C_3\tau^2 \sum_{\ell=0}^{n-1}(1 + C_4\tau)^\ell &\leq C_3\tau^2 \sum_{\ell=0}^{N-1}(1 + C_4\tau)^\ell = C_3\tau^2 \frac{(1 + C_4\tau)^{N-1} - 1}{(1 + C_4\tau) - 1} \\ &= C_3C_4^{-1}\big((1 + C_4\tau)^{N-1} - 1\big)\tau \leq C_3C_4^{-1}\big(\exp(C_4TK^{-1}) - 1\big)\tau = c_1\tau.\end{aligned}$$

This gives a bound for the first term of (30). For the second term we use (29), (31) to get

$$(1 + C_4\tau)^n e_{k,0} \leq (1 + C_4\tau)^N e_{k,0} \leq C_L \exp(C_4 T K^{-1}) e_{k-1,N} = c_2 e_{k-1,N}.$$

To sum up, we have

$$e_{k,n} \leq c_1\tau + c_2 e_{k-1,N}.$$

Repeating this procedure k times we obtain

$$e_{k,n} \leq c_1\tau \sum_{\ell=0}^{k-1} c_2^\ell + c_2^k e_{0,0}.$$

Since $\pi_0 = q_0$ the second term vanishes. We recall that K is finite, while N tends to infinity. Introducing $C = c_1 \sum_{\ell=0}^{K} c_2^\ell$, we obtain

$$e_{k,n} \leq c_1\tau \sum_{\ell=0}^{K} c_2^\ell = C\tau.$$

This completes the proof of Lemma 2.1. □

4. Numerical experiments

In this section we examine the strong convergence order of $\widetilde{\pi}$ numerically. We begin by reminding that the convergence in Corollary 2.1 holds for all $y \in \mathrm{supp}(\mathbb{P}_Y)$, and in this section we fix $Y = O$, defined in the statistical model (2), and let $\widetilde{q}_0 = q_0$. First, in Section 4.1, we detail the final approximation steps needed to perform the numerical experiments. Section 4.2 contains a discussion of numerical design choices. In Section 4.3 we describe the approximation of the left hand side in Corollary 2.1. Section 4.4 contains the empirical convergence study on two one-dimensional examples. Finally, in Section 4.5 we demonstrate the method on a nonlinear 10-dimensional process and compare the performance to classical filter approximations.

4.1. Energy-based approximation.

In the previous work [4], the Energy-Based Deep Splitting (EBDS) method was employed on the Zakai equation, solving the filtering problem with continuous in time observations. Here we adapt the same approximation procedures. The initial step involves approximating the expectation in (19) using a Monte Carlo average. This consists of sampling M independent pairs $(Z^m_{0:N}, Y^m_{0:K})_{m=1}^M$ from (15) and (2) respectively. The updated problem reads

$$\begin{aligned}
(\widetilde{\pi}^M_{k,n+1}(x,y))_{(x,y)\in\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)}} &= \operatorname*{arg\,min}_{u\in C(\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)};\mathbb{R})} \frac{1}{M}\sum_{m=1}^{M} \left| u(Z^m_{N-(n+1)}, Y^m_{0:k}) - G_b\widetilde{\pi}^M_{k,n}(Z^m_{N-n}, Y^m_{0:k}) \right|^2, \\
&\qquad k = 0,\dots,K-1, \quad n = 0,\dots,N-1, \\
\widetilde{\pi}^M_{0,0}(x,y_0) &= q_0(x)L(y_0,x), \\
\widetilde{\pi}^M_{k,0}(x,y_{0:k}) &= \widetilde{\pi}^M_{k-1,N}(x,y_{0:k-1})L(y_k,x), \quad k = 1,\dots,K.
\end{aligned}$$

Approximating expectations with Monte Carlo averages is a classical technique and by the central limit theorem one can, for well behaved integrands, expect a convergence of order $\frac{1}{2}$ in M.

The second step is to approximate $C(\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)};\mathbb{R})$ by a neural network, more precisely, by the space $\{\mathcal{N}_\theta : \theta \in \Theta_k\} \subset C(\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)};\mathbb{R})$. Here every θ in the space of parameters $\Theta_k \subset \mathbb{R}^{(d+d'\times(k+1))\times J^{\mathfrak{L}}\times 1}$ defines a continuous function $\mathcal{N}_\theta$, a Fully Connected Neural Network (FCNN) with $\mathfrak{L}$ hidden layers and a width of J neurons, mapping $\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)}$ to $\mathbb{R}$. In this work we use ReLU activation functions between each hidden layer except the final one, for which we instead use an energy-based activation described below. Other constructions with more advanced (or simpler) architectures could be of interest in future work, but here the focus lies on examining the convergence order numerically rather than on optimized

architectures. The new optimization problem reads

$$
\begin{aligned}
(\widetilde{\pi}^{M,\mathfrak{L}}_{k,n+1}(x,y))_{(x,y)\in\mathbb{R}^d\times\mathbb{R}^{d'\times(k+1)}} &= \operatorname*{arg\,min}_{u\in\{\mathcal{N}_\theta:\theta\in\Theta_k\}} \frac{1}{M}\sum_{m=1}^{M}\left|u(Z^m_{N-(n+1)},Y^m_{0:k}) - G_b\widetilde{\pi}^{M,\mathfrak{L}}_{k,n}(Z^m_{N-n},Y^m_{0:k})\right|^2, \\
&\qquad k=0,\dots,K-1, \quad n=0,\dots,N-1, \\
\widetilde{\pi}^{M,\mathfrak{L}}_{0,0}(x,y_0) &= q_0(x)L(y_0,x), \\
\widetilde{\pi}^{M,\mathfrak{L}}_{k,0}(x,y_{0:k}) &= \widetilde{\pi}^{M,\mathfrak{L}}_{k-1,N}(x,y_{0:k-1})L(y_k,x), \quad k=1,\dots,K.
\end{aligned} \tag{32}
$$

In [51] it is shown how well FCNNs approximate functions in Sobolev spaces. Specifically, with a fixed width J, it can be shown that the approximation converges with order $\mathfrak{L}^{-\gamma}$, where J and $\gamma > 0$ depend on the underlying dimension d and the regularity of the Sobolev space.

The energy-based activation function mentioned above consists of passing a scalar output $f_\theta(x, y_{0:k})$ through the exponential function. That is, we let the normalized conditional density be approximated, for each $\theta \in \Theta_k$ and input pair $(x, y_{0:k}) \in \mathbb{R}^d \times \mathbb{R}^{d'\times(k+1)}$, by

$$
p(x \mid y_{0:k}) \approx \frac{\mathcal{N}_\theta(x, y_{0:k})}{Z_\theta(y_{0:k})}
$$

where

$$
\mathcal{N}_\theta(x, y_{0:k}) = \mathrm{e}^{-f_\theta(x,y_{0:k})}, \quad Z_\theta(y_{0:k}) = \int_{\mathbb{R}^d} \mathrm{e}^{-f_\theta(z,y_{0:k})}\,\mathrm{d}z.
$$

There are two main ideas behind the use of energy-based activation for probabilistic modeling. The first is that we do not have to evaluate the normalizing constant Z_θ, if we are only concerned with the unnormalized density. This speeds up training and still learns the density. The second concerns the scalar energy f_θ. The idea is that it should assign low energy (resulting in a high probability) to pairs $(x, y_{0:k})$, that are more likely to occur and high energy (resulting in low probability) to pairs that are less likely to occur. This fits very well with solving (32), where it is enough to approximate an unnormalized solution of the Fokker–Planck equation (6) and to do the normalization in the inference stage. Trivially, this activation guarantees positive values, which is crucial when approximating probabilities. One can also construct f_θ in such a way that it guarantees that Z_θ is finite, e.g., by learning Gaussian tails [4]. Finally, we employ the time reparametrization that was done in [4, Section 3.4], which defines an equivalent optimization problem that is more suitable for computations.

4.2. **Discussion on numerical approximations.** In this subsection we collect a few remarks on numerical design choices in the proposed scheme. The first concerns the choice of the auxiliary drift function b in the process X. The second concerns how to pass from the unnormalized surrogate $\widetilde{\pi}$ produced by the algorithm to a numerically stable approximation of the normalized filtering density, including practical strategies for approximating the corresponding normalizing constants in moderate and high dimensions.

4.2.1. *Choosing the auxiliary drift function.* In Section 2 we introduced the auxiliary process X with drift function b. The natural candidate, which was the only choice in the original derivation [7], is to set $b = \mu$. In this case, if $\widetilde{q}_0 = q_0$, then S and X have the same distribution. This choice is natural from a sampling perspective, since the auxiliary process X then explores the same region of the state space as the state S. The second choice that we consider is given by (5), which eliminates the first order terms in F_b and therefore allows us to prove strong convergence of order one. A third, not explored, option would be to use a drift coefficient that more efficiently explores the tails of the distribution. For illustration, consider the one-dimensional case where S is a mean-reverting process such that $S_T \sim \mathcal{N}(0, 1)$. If we choose an auxiliary drift b with smaller magnitude, thereby weakening the mean reversion, the process X will have heavier tails. Heuristically, this may allow the method to explore the tail behaviour of the desired distribution more efficiently, providing more information about these regions for a fixed number of Monte Carlo samples M. A systematic design and analysis of such tail-exploring drifts is beyond the scope of this work, but appears to be an interesting direction for future research.

4.2.2. *Normalization.* In Section 2 we introduced the unnormalized filtering density, whose approximation is analyzed in Section 3. The analysis is valid for both the normalized and unnormalized versions: given an approximation $\widetilde{\pi}^{M,\mathfrak{L}}_{k,0}(x, Y^m_{0:k})$ of the unnormalized density, an approximation of the corresponding normalized filtering density is obtained by dividing by an approximation of its normalizing constant.

We emphasize that the unnormalized density is also of practical interest. In particular, if one is primarily interested in generating samples from the filtering density, it is often sufficient to know the density up to a multiplicative constant. Since our surrogate $\widetilde{\pi}$ is differentiable in x, it can be used directly as a potential in gradient-based Markov chain Monte Carlo methods. For example, Hamiltonian Monte Carlo [24, 11] can efficiently explore the support of the density by combining Hamiltonian dynamics with an accept-reject step, without ever requiring explicit evaluation of the normalizing constant.

For numerical stability in the training procedure, we nevertheless incorporate an explicit normalization of $\widetilde{\pi}^{M,\mathfrak{L}}_{k,0}(Y^m_{0:k})$, for $k = 0, \dots, K$, and $m = 1, \dots, M$, before training each subsequent network. In Section 4.4 we evaluate the strong error in two one-dimensional examples, where the normalizing constants can be accurately computed by quadrature.

In the high-dimensional setting we instead approximate the normalizing constant by importance sampling. Let q be an importance distribution and write the normalizing constant as

$$C^{m,k} = \int_{\mathbb{R}^d} \widetilde{\pi}^{M,\mathfrak{L}}_{k,0}(x, Y^m_{0:k}) \, \mathrm{d}x = \int_{\mathbb{R}^d} \frac{\widetilde{\pi}^{M,\mathfrak{L}}_{k,0}(x, Y^m_{0:k})}{q(x \mid Y^m_{0:k})} q(x \mid Y^m_{0:k}) \, \mathrm{d}x, \quad m = 1, \dots, M, \quad k = 0, \dots, K.$$

To avoid the curse of dimensionality, while still allowing q to depend on $Y^m_{0:k}$, we construct q from an Extended Kalman Filter (EKF) approximation of the filtering distribution, see, e.g., [49]. More precisely, the EKF provides a Gaussian approximation with mean and covariance (m^m_k, P^m_k), and we take

$$q(x \mid Y^m_{0:k}) = \mathcal{N}\big(x;\, m^m_k,\, \lambda P^m_k\big),$$

with an inflation factor $\lambda \geq 1$ to ensure that the support of q covers the main mass of $\widetilde{\pi}^{M,\mathfrak{L}}_{k,0}$. The dominant numerical cost in this construction is the inversion of the $d \times d$ covariance matrices, which is at most of order d^3 per observation time and therefore avoids the exponential growth associated with grid-based quadrature in high dimensions.

4.3. **Evaluation.** The objective of Section 4 is to numerically examine the convergence order of the approximation $\widetilde{\pi}$. To do this, we need to approximate the left hand side of (20). Subsection 4.1 introduces the approximation $\widetilde{\pi}^{M,\mathfrak{L}}_{k,n}$ of $\widetilde{\pi}_{k,n}$. Now we measure the error between $p_k(t_{k,n})$ and $\widetilde{\pi}^{M,\mathfrak{L}}_{k,n}$ in $L^2(\Omega; L^\infty(\mathbb{R}^d; \mathbb{R}))$ for fixed $(k, n) \in \mathcal{I}_{K,N}$. In the numerical experiments we thus measure the error in the weaker norm $L^2(\Omega; L^\infty(\mathbb{R}^d; \mathbb{R}))$ rather than in $L^\infty(\mathrm{supp}(\mathbb{P}_Y); L^\infty(\mathbb{R}^d; \mathbb{R}))$. The latter would require a numerical approximation of a supremum over the full support of the observation process; in our examples in Section 4.4 we have $\mathrm{supp}(\mathbb{P}_Y) = \mathbb{R}^{10}$, which makes such a computation prohibitively expensive and unstable. By contrast, the L^2-norm admits a straightforward Monte Carlo approximation which is much more accurate. By the triangle inequality and by estimating the L^2-norm by the L^∞-norm, we have

$$\begin{aligned} &\big\|p_k(t_{k,n}) - \widetilde{\pi}^{M,\mathfrak{L}}_{k,n}\big\|_{L^2(\Omega; L^\infty(\mathbb{R}^d;\mathbb{R}))} \\ &\quad \leq \big\|p_k(t_{k,n}) - \widetilde{\pi}_{k,n}\big\|_{L^\infty(\mathrm{supp}(\mathbb{P}_Y); L^\infty(\mathbb{R}^d;\mathbb{R}))} + \big\|\widetilde{\pi}_{k,n} - \widetilde{\pi}^{M,\mathfrak{L}}_{k,n}\big\|_{L^\infty(\mathrm{supp}(\mathbb{P}_Y); L^\infty(\mathbb{R}^d;\mathbb{R}))}. \end{aligned} \tag{33}$$

We see that the first term on the right hand side of (33) is the left hand side in Corollary 2.1. The second term is the error that occurs due to the statistical approximation depending on M and the approximation error by the neural network depending on $\mathfrak{L}$. In Subsection 4.4 we set M and $\mathfrak{L}$ sufficiently large such that the first term on the right hand side of (33) is the dominating error.

We need further approximations to evaluate the left hand side of (33), which is, for $(k, n) \in \mathcal{I}_{K,N}$, defined by

$$\begin{aligned} \big\|p_k(t_{k,n}) - \widetilde{\pi}^{M,\mathfrak{L}}_{k,n}\big\|_{L^2(\Omega; L^\infty(\mathbb{R}^d;\mathbb{R}))} &= \mathbb{E}\Big[\big\|p_k(t_{k,n}, Y_{0:k}) - \widetilde{\pi}^{M,\mathfrak{L}}_{k,n}(Y_{0:k})\big\|^2_{L^\infty(\mathbb{R}^d;\mathbb{R})}\Big]^{\frac{1}{2}} \\ &= \mathbb{E}\Big[\sup_{x \in \mathbb{R}^d} \big|p_k(t_{k,n}, x, Y_{0:k}) - \widetilde{\pi}^{M,\mathfrak{L}}_{k,n}(x, Y_{0:k})\big|^2\Big]^{\frac{1}{2}}. \end{aligned} \tag{34}$$

To evaluate (34), we first approximate the expectation with M_e Monte Carlo samples from (2). Second, since we are only evaluating this in $d = 1$, we approximate the supremum by taking the supremum over a uniform grid B in a bounded domain of $\mathbb{R}$, where the solutions have most of their mass. Combining

the approximations, we get

$$\mathbb{E}\Big[\sup_{x\in\mathbb{R}^d}\big|p_k(t_{k,n},x,Y_{0:k})-\widetilde{\pi}^{M,\mathfrak{L}}_{k,n}(x,Y_{0:k})\big|^2\Big]^{\frac{1}{2}} \approx \Big(\frac{1}{M_\mathrm{e}}\sum_{m_\mathrm{e}=1}^{M_\mathrm{e}}\sup_{x\in B}\big|p_k(t_{k,n},x,Y^{m_\mathrm{e}}_{0:k})-\widetilde{\pi}^{M,\mathfrak{L}}_{k,n}(x,Y^{m_\mathrm{e}}_{0:k})\big|^2\Big)^{\frac{1}{2}}. \tag{35}$$

The approximation error becomes arbitrarily small by choosing $|B|$ and M_e large enough.

4.4. **Numerical convergence.** In this section we empirically examine the convergence order of the EBDS method applied to two one-dimensional examples. The first one, which we refer to as the tanh process, has the tanh function as drift coefficient and satisfies the assumptions in Section 2.2. The other is a bistable process, characterized by two modes in its corresponding probability distribution, and does not satisfy the assumptions. In both examples, we have $q_0 = \mathcal{N}(0,1)$, $\sigma(x) = 1$, Gaussian measurements $Y_k \sim \mathcal{N}(h(S_{t_{k,0}}), R)$ with measurement function $h(x) = x$ and variance $R = 1$. It is worth mentioning that this corresponds to a rather low signal-to-noise ratio, which some would argue is a harder problem. The tanh process has the drift coefficient $\mu(x) = \tanh(x)$, while the bistable process has a drift coefficient $\mu(x) = \frac{2}{5}(5x - x^3)$. In both cases, we use $K = 20$ with equidistant measurements with a final time of $T = 2$. This decision was made based on the observation that the $L^2(L^\infty)$-error reaches a relatively stable value before $t = 2$ for both examples. Furthermore, we utilize two different choices of b. In the first example we fulfill the assumptions of Corollary 2.1 by picking b as in (5), and in the second example we use $b = \mu$.

To evaluate (35), we need the reference solution $p_k(t_{k,n})$. In both examples we lack analytical solutions and instead employ a bootstrap particle filter with 10^5 particles, with 128 auxiliary time steps, to find a good approximation of the true solution. Finally, the constants for the numerical evaluation in the tanh process example and the bistable example are, respectively: $M_\mathrm{e} = \{4000, 500\}$, $J = 128$, $M = 10^6$, $\mathfrak{L} = 3$ and $|B| = 2\,000$.

In Figure 1 we see the error (35) with 5 different discretizations $N = 1, 2, 4, 8, 16$ over time. Each error trajectory corresponds to the average over 10 runs of the method to better illustrate the performance. It is easy to see how the error decreases by an increasing number of time steps N in both examples. On the dashed lines, at time steps $(t_{k,0})_{k=0}^K$, the solutions are updated with new measurements. We note that the overall error level is substantially lower for the tanh process, and in both examples the finer discretizations lead to an earlier plateau of the error.

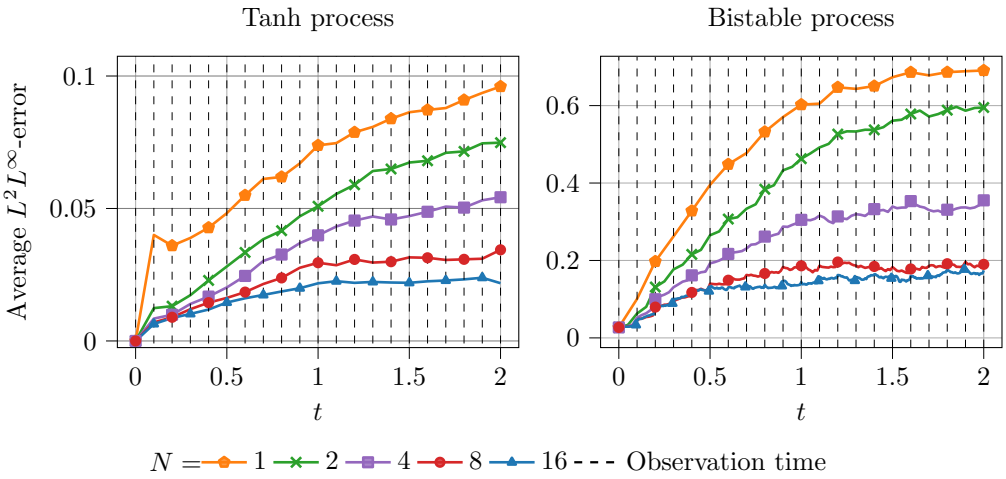

Figure 1. The figure illustrates the $L^2(\Omega; L^\infty(\mathbb{R}^d; \mathbb{R}))$-error over time for five different discretizations averaged over 10 instances. To the left we see the error for the tanh process and to the right we see the error for the example with the bistable process.

Looking at the final time $T = 2$ for the different discretizations, we see the convergence in Figure 2. In the figure, we see 10 instances of the performed algorithm, obtaining 10 slightly different approximations due to stochastic gradient descent, illustrated in red. The figure has a logarithmic scale and shows that the average error, illustrated in blue, decreases with order slightly less than 1 for the tanh process and with about order $\frac{1}{2}$ for the bistable example compared to the reference lines of $O(N^{-1})$ and $O(N^{-\frac{1}{2}})$, where $N = \frac{T}{K\tau}$.

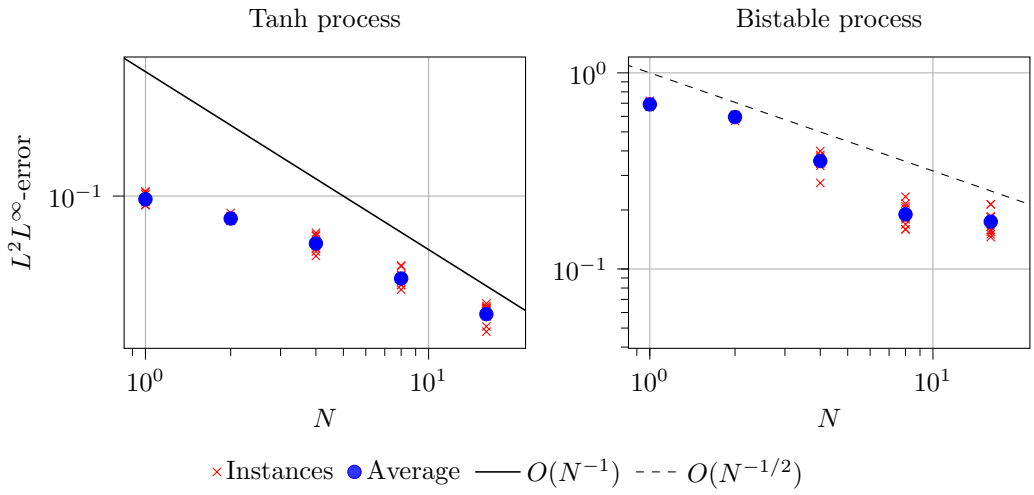

Figure 2. The figure presents the convergence for the numerical scheme for 10 individual instances of the scheme in red, their average in blue, and in black we see reference lines of order 1 and $\frac{1}{2}$, respectively. To the left we have the errors corresponding to the tanh process and to the right the example with the bistable process.

This indicates that the assumptions in Section 2.2 are crucial for the numerical convergence order and that order $\frac{1}{2}$ might be possible under weaker assumptions.

For transparency, we also remark that if we let M and $\mathfrak{L}$ stay constant and keep increasing N, the error will stop to decrease after some threshold as the statistical error and neural network approximation error starts to dominate. To examine the convergence we set a sufficiently large sample size $M = 10^6$ and layer depth $\mathfrak{L} = 3$ for all the discretizations.

4.5. **High-dimensional demonstration.** In our final example we consider a high-dimensional version of the tanh process of Section 4.4. More precisely, we let C be a $d \times d$-matrix, with entries $C_{i,j}$, $i, j \in \{1, \dots, d\}$, uniformly distributed in $[-0.5, 0.5]$, and let the drift be defined by $\mu_i(x) = \sum_{j=1}^d C_{i,j} \tanh(x_j)$ for $i = 1, \dots, d$. Similarly, we let the diffusion coefficient be constant $\sigma(x) = \Sigma$, with a $d \times d$-matrix Σ, with entries $\Sigma_{i,j}$, $i, j \in \{1, \dots, d\}$, uniformly distributed in $[-0.5, 0.5]$. We let $\widetilde{q}_0 = q_0 = \mathcal{N}(0, I)$, and define the observation process by $Y_k \sim \mathcal{N}(h(S_{t_{k,0}}), R)$ with $h(x) = x$ and $R = I$. In this example we let $T = 1$, $K = 10$ and choose $b = \mu$. Furthermore, in this high-dimensional framework with increasingly small probability values we need to work in log-scale. Hence, we employ the logarithmic version of the method, defined in [5]. To train the method we use $M = 10^7$, $J = 512$, $\mathfrak{L} = 3$.

In this high-dimensional nonlinear setting we lack obvious candidates for reference solutions. Instead we consider two metrics, where we do not need a reference solution, but where it suffices to use samples of S. For $k = 0, \dots, K$, given an approximate mean $\widehat{m}_k$, and an approximate density $\widehat{p}_k$, we consider a Mean Absolute Error (MAE) and a Negative Log-Likelihood (NLL), which are defined, and approximated, by

$$\mathrm{MAE}(\widehat{m}_k) = \mathbb{E}\Big[\big\|S_{t_k} - \widehat{m}_k\big\|\Big] \approx \frac{1}{M_{\mathrm{e}}} \sum_{m_{\mathrm{e}}=1}^{M_{\mathrm{e}}} \big\|S_{t_k}^{(m_{\mathrm{e}})} - m_k^{(m_{\mathrm{e}})}\big\|,$$

$$\mathrm{NLL}(\widehat{p}_k) = \mathbb{E}\Big[-\log\big(\widehat{p}_k(S_{t_k} \mid Y_{0:k})\big)\Big] \approx -\frac{1}{M_{\mathrm{e}}} \sum_{m_{\mathrm{e}}=1}^{M_{\mathrm{e}}} \log\Big(\widehat{p}_k\big(S_{t_k}^{(m_{\mathrm{e}})} \mid Y_{0:k}^{(m_{\mathrm{e}})}\big)\Big).$$

By comparing our method to classical filters, such as an Ensemble Kalman Filter (EnKF) [12] and a bootstrap Particle Filter (PF) [33], with these metrics, we quantify how well the method is performing. Neither of these metrics are expected to yield 0 but rather be as small as possible, while the exact filter is the true minimizer of both.

In Figure 3 we report the errors evaluated with $M_{\mathrm{e}} = 10^4$ samples. The figure compares a range of approximative filters, including our proposed method. For the latter we show results for $N = 16$ and $N = 32$ to highlight the dependence on the auxiliary time discretization within the method. We observe that the method is robust and achieves satisfactory accuracy, with a clear improvement when refining the discretization. Moreover, the Kalman-based methods perform best overall, which is consistent with the fact that the posterior is expected to be close to Gaussian, since the tanh-function is approximately linear on $[-0.5, 0.5]$. The particle filter, on the other hand, is asymptotically exact as the number of particles tends to infinity [16], but already in this fully interacting 10-dimensional example we require on the order of 10^6 particles to outperform our proposed method in terms of the NLL metric. Finally, we note that our method performs comparatively better in NLL than in MAE, indicating that it captures the overall shape of the filtering density more accurately than its mean.

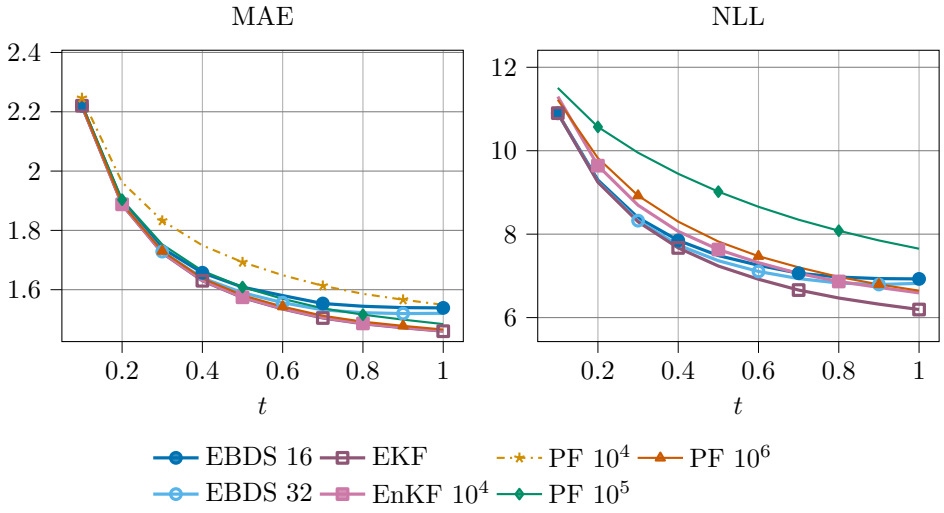

Figure 3. The figure presents the two errors, mean absolute error to the left, and the negative log-likelihood to the right, for the different approximative filters when applied to the 10-dimensional Tanh process example.

Acknowledgements

The work of K.B. and S.L. was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The work of F.R. was funded by the Swedish Electromobility Centre (SEC) and partially supported by WASP. The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at Chalmers e-Commons partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

[1] K. Andersson, A. Andersson, and C. W. Oosterlee. Convergence of a robust deep FBSDE method for stochastic control. *SIAM J. Sci. Comput.*, 45:A226–A255, 2023.

[2] A. Apte, C. K. R. T. Jones, A. M. Stuart, and J. Voss. Data assimilation: Mathematical and statistical perspectives. *Int. J. Numer. Methods Fluids*, 56:1033–1046, 2008.

[3] E. Bach, R. Baptista, E. Luk, and A. Stuart. Learning optimal filters using variational inference. *arXiv:2406.18066*, 2024.

[4] K. Bågmark, A. Andersson, and S. Larsson. An energy-based deep splitting method for the nonlinear filtering problem. *Partial Differ. Equ. Appl.*, 4, 2023.

[5] K. Bågmark and F. Rydin. High-dimensional Bayesian filtering through deep density approximation. *arXiv:2511.07261*, 2025.

[6] C. Beck, S. Becker, P. Cheridito, A. Jentzen, and A. Neufeld. Deep learning based numerical approximation algorithms for stochastic partial differential equations and high-dimensional nonlinear filtering problems. *arXiv:2012.01194*, 2020.

[7] C. Beck, S. Becker, P. Cheridito, A. Jentzen, and A. Neufeld. Deep splitting method for parabolic PDEs. *SIAM J. Sci. Comput.*, 43:A3135–A3154, 2021.

[8] C. Beck, S. Becker, P. Grohs, N. Jaafari, and A. Jentzen. Solving the Kolmogorov PDE by means of deep learning. *J. Sci. Comput.*, 88(3):73, 2021.

[9] S. S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House Publishers, 1999.

[10] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*, volume 15 of *Texts in Applied Mathematics*. Springer, New York, third edition, 2008.

[11] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

[12] G. Burgers, P. J. van Leeuwen, and G. Evensen. Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, 126(6):1719 – 1724, 1998.

[13] F. Cassola and M. Burlando. Wind speed and wind energy forecast through Kalman filtering of numerical weather prediction model output. *Appl. Energy*, 99:154–166, 2012.

[14] P. Cattiaux and L. Mesnager. Hypoelliptic non-homogeneous diffusions. *Probab. Theory Related Fields*, 123:453–483, 2002.

[15] S. Challa and Y. Bar-Shalom. Nonlinear filter design using Fokker-Planck-Kolmogorov probability density evolutions. *IEEE Trans. Aerosp. Electron. Syst.*, 36:309–315, 2000.

[16] N. Chopin. Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *Ann. Statist.*, 32(6):2385–2411, 2004.

[17] A. Corenflos and A. Finke. Particle-MALA and Particle-mGRAD: Gradient-based MCMC methods for high-dimensional state-space models. *arXiv:2401.14868*, 2024.

[18] A. Corenflos, Z. Zhao, S. Särkkä, J. Sjölund, and T. B. Schön. Conditioning diffusion models by explicit forward-backward bridging. *Proc. 28th Int. Conf. Artif. Intell. Stat. (AISTATS)*, 2024.

[19] D. Crisan, A. Lobbe, and S. Ortiz-Latorre. An application of the splitting-up method for the computation of a neural network representation for the solution for the filtering equations. *Stoch. Partial Differ. Equ.: Anal. Comput.*, 10:1050–1081, 2022.

[20] N. Cui, L. Hong, and J. R. Layne. A comparison of nonlinear filtering approaches with an application to ground target tracking. *Signal Processing*, 85:1469–1492, 2005.

[21] G. Da Prato. *Introduction to Stochastic Analysis and Malliavin Calculus*, volume 13 of *Lecture Notes. Scuola Normale Superiore di Pisa (New Series)*. Edizioni della Normale, Pisa, third edition, 2014.

[22] P. Date and K. Ponomareva. Linear and non-linear filtering in mathematical finance: a review. *IMA J. Manag. Math.*, 22:195–211, 2011.

[23] B. Demissie, M. A. Khan, and F. Govaers. Nonlinear filter design using Fokker-Planck propagator in Kronecker tensor format. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2016.

[24] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987.

[25] L. Duc, T. Kuroda, K. Saito, and T. Fujita. Ensemble Kalman filter data assimilation and storm surge experiments of tropical cyclone nargis. *Tellus A*, 67:25941, 2015.

[26] W. E, J. Han, and A. Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Commun. Math. Stat*, 5:349–380, Nov. 2017.

[27] W. E and B. Yu. The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems. *Commun. Math. Stat*, 1:1–12, 2018.

[28] A. Finke and A. H. Thiery. Conditional sequential Monte Carlo in high dimensions. *Ann. Statist.*, 51:437–463, 2023.

[29] R. Frey and V. Köck. Convergence analysis of the deep splitting scheme: the case of partial integro-differential equations and the associated forward backward SDEs with jumps. *SIAM J. Sci. Comput.*, 47(1):A527–A552, 2025.

[30] G. Galanis, P. Louka, P. Katsafados, I. Pytharoulis, and G. Kallos. Applications of Kalman filters based on non-linear functions to numerical weather predictions. *Ann. Geophys*, 24:1–10, 2006.

[31] E. Gobet. *Monte-Carlo Methods and Stochastic Processes*. CRC Press, Boca Raton, FL, 2016.

[32] I. R. Goodman, R. P. S. Mahler, and H. T. Nguyen. *Mathematics of Data Fusion*. Kluwer Academic Publishers Group, Dordrecht, 1997.

[33] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F (Radar and Signal Processing)*, 140(2):107–113, 1993.

[34] M. Hairer, M. Hutzenthaler, and A. Jentzen. Loss of regularity for Kolmogorov equations. *Ann. Probab.*, 43:468–527, 2015.

[35] J. Han, W. Hu, J. Long, and Y. Zhao. Deep Picard iteration for high-dimensional nonlinear PDEs. *SIAM J. Sci. Comput.*, 48(1):C1–C24, 2026.

[36] Z. Hu, Z. Zhang, G. E. Karniadakis, and K. Kawaguchi. Score-based physics-informed neural networks for high-dimensional Fokker-Planck equations. *SIAM J. Sci. Comput.*, 47(3):C680–C705, 2025.

[37] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *J. Basic Eng.*, 83:95–108, 1961.

[38] F. C. Klebaner. *Introduction to Stochastic Calculus with Applications*. Imperial College Press, London, third edition, 2012.

[39] A. Klenke. *Probability Theory*. Universitext. Springer, London, second edition, 2014.

[40] P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*, volume 23 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, 1992.

[41] H. J. Kushner. On the differential equations satisfied by conditional probablitity densities of Markov processes, with applications. *J. Soc. Industrial Appl. Math., Series A: Control*, 2:106–119, 1964.
[42] A. Lobbe. Deep learning for the Benes filter. *Stochastic Transport in Upper Ocean Dynamics*, 10:195–210, 2023.
[43] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nat. Mach. Intell.*, 3:218–229, 2021.
[44] C. A. Naesseth, F. Lindsten, and T. B. Schön. High-dimensional filtering using nested sequential Monte Carlo. *IEEE Trans. Signal Process.*, 67:4177–4188, 2019.
[45] J. Quinn. A high-dimensional particle filter algorithm. *arXiv:1901.10543*, 2019.
[46] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 378:686–707, 2019.
[47] P. Rebeschini and R. van Handel. Can local particle filters beat the curse of dimensionality? *Ann. Appl. Probab.*, 25:2809–2866, 2015.
[48] W. Rutzler. Nonlinear and adaptive parameter estimation methods for tubular reactors. *Ind. Eng. Chem. Res.*, 26:325–333, 1987.
[49] S. Särkkä and L. Svensson. *Bayesian Filtering and Smoothing*, volume 17 of *Institute of Mathematical Statistics Textbooks*. Cambridge University Press, Cambridge, second edition, 2023.
[50] M. Schauer, F. van der Meulen, and H. van Zanten. Guided proposals for simulating multi-dimensional diffusion bridges. *Bernoulli*, 23(4A):2917–2950, 2017.
[51] J. W. Siegel. Optimal approximation rates for deep ReLU neural networks on Sobolev and Besov spaces. *J. Mach. Learn. Res.*, 24:1–52, 2023.
[52] C. Snyder. Particle filters, the "optimal" proposal and high-dimensional systems. In *Proceedings of the ECMWF Seminar on Data Assimilation for atmosphere and ocean*, pages 1–10, 2011.
[53] C. Snyder, T. Bengtsson, and M. Morzfeld. Performance bounds for particle filters using the optimal proposal. *Mon. Weather Rev.*, 143:4750–4761, 2015.
[54] M. Zakai. On the optimal filtering of diffusion processes. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 11:230–243, 1969.
[55] Y. Zeng and S. Wu, editors. *State-Space Models*. Statistics and Econometrics for Finance. Springer, New York, 2013.
[56] Z. Zhao, Z. Luo, J. Sjölund, and T. B. Schön. Conditional sampling within generative diffusion models. *Philos. Trans. R. Soc. A*, 383(2299):20240329, 2025.

Kasper Bågmark, Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE–412 96 Gothenburg, Sweden

Email address: bagmark@chalmers.se

Adam Andersson, Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, S–412 96 Gothenburg, Sweden, and Saab AB, S–412 76 Gothenburg, Sweden

Email address: adam.andersson@chalmers.se

Stig Larsson, Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE–412 96 Gothenburg, Sweden

Email address: stig@chalmers.se

Filip Rydin, Department of Electrical Engineering, Chalmers University of Technology and University of Gothenburg, SE–412 96 Gothenburg, Sweden

Email address: filipry@chalmers.se