Deep Filtering With Adaptive Learning Rates

Hongjiang Qian, George Yin, *Life Fellow, IEEE*, and Qing Zhang, *Senior Member, IEEE*

***Abstract*—This article develops a new deep learning framework for general nonlinear filtering. Our main contribution is to present a computationally feasible procedure. The proposed algorithms have the capability of dealing with challenging (infinitely dimensional) filtering problems involving diffusions with randomly-varying switching. First, we convert it to a problem in a finite-dimensional setting by approximating the optimal weights of a neural network. Then, we construct a stochastic gradient-type procedure to approximate the neural network weight parameters, and develop another recursion for adaptively approximating the optimal learning rate. The convergence of the combined approximation algorithms is obtained using stochastic averaging and martingale methods under suitable conditions. Robustness analysis of the approximation to the network parameters with the adaptive learning rate is also dealt with. We demonstrate the efficiency of the algorithm using highly nonlinear dynamic system examples.**

***Index Terms*—Deep learning, filtering, stochastic approximation (SA).**

I. Introduction

THIS article develops a novel approach to nonlinear filtering using deep learning techniques. It presents a computationally feasible procedure. The proposed algorithms have the capacity of dealing with challenging filtering problems involving switching diffusions. First, we convert the infinitely dimensional problem to a finite-dimensional setting. The problem becomes to approximate the optimal weights of the corresponding neural network (NN). Then, we construct a stochastic gradient-type procedure to approximate the optimal weight parameters, and develop another recursion for adaptively approximating the optimal learning rate (LR).

There is a long history of nonlinear filtering. Given the state of a system that is not completely observable, filtering is concerned with state estimation based on partial observations of the system state. Nonlinear filtering focuses on state estimation. Devoted to conditional mean or distribution, the problem is in fact, infinite dimensional. To illustrate the setup and the nature of the problem and fundamental difficulty, we briefly describe the setting of the problem. We begin by considering the state of the system x_t, a Markov process with an associate operator $\mathcal{A}$ (called the extended generator). A function of x_t is observable with additive white noise so that the observation process y_t is

$$dy_t = \widetilde{h}(x_t)dt + dw_t$$

where $\widetilde{h}$ is a continuous function and w_t is a Brownian motion independent of x_t. Nonlinear filtering focuses on calculating the conditional distribution or conditional mean of x_t given the information of the observation up to time t, namely, $\mathcal{Y}_t := \sigma\{y_s : s \le t\}$ (the σ-algebra generated by y_s up to time t). Early developments in nonlinear filtering can be found in the classical work of Kushner [12], Duncan [7], Mortensen [17], and Zakai [28], among others.

Consider a function space E (a complete and separable metric space) and sample paths of x_t in the space of right-continuous functions with left-hand limits endowed with weak topology [known as $D[0,T]$ (see [14, Ch. 7])]. Denote by π_t the conditional distribution of x_t given $\mathcal{Y}_t$ (information of the observation up to time t). For any Borel function (Borel measurable) f on E, the conditional mean given the information of the observation up to time t is given by $\pi_t(f) = \int_E f(x)\pi_t(dx)$. Then, $\pi_t(f)$ is known to satisfy the so-called Kushner equation

$$d\pi_t(f) = \pi_t(\mathcal{A}f)dt + (\pi_t(\widetilde{h}f) - \pi_t(f)\pi_t(\widetilde{h}))d\nu_t$$

where $d\nu_t = dy_t - \pi_t(\widetilde{h})dt$ and ν_t is known as the innovation process. Denote by $\sigma_t(f)$ the unnormalized conditional distribution (that is, the conditional distribution is not a probability measure)

$$\sigma_t(f) = \pi_t(f)\exp\left(\int_0^t \pi_u(\widetilde{h})d\nu_u + \frac{1}{2}\int_0^t (\pi_u(\widetilde{h}))^2 du\right).$$

Then, $\sigma_t(f)$ satisfies the following equation (see [28])

$$d\sigma_t(f) = \sigma_t(\mathcal{A}f)dt + \sigma_t(\widetilde{h}f)dy_t.$$

It has been referred to as the Duncan–Mortensen–Zakai equation subsequently. To calculate the unnormalized conditional distribution $\sigma_t(f)$, a typical approach is to first find a solution $\sigma_t(f)$ to the Duncan–Mortensen–Zakai equation, then show that it is indeed the conditional distribution under uniqueness of the solution of the differential equation. One of such approaches can be found in [11] in terms of the well posedness of the corresponding martingale problem; see also [5] for related treatment. Note that the conditional distribution given above involves partial differential equations of infinite dimension. Although the celebrated results of the Kushner equation and the Duncan–Mortensen–Zakai equation decisively settled the

Manuscript received 11 December 2021; revised 19 April 2022; accepted 7 June 2022. Date of publication 15 June 2022; date of current version 29 May 2023. This work was supported in part by the National Science Foundation under Grant DMS-2204240. Recommended by Associate Editor D. Antunes. *(Corresponding author: George Yin.)*

Hongjiang Qian and George Yin are with the Department of Mathematics, University of Connecticut, Storrs, CT 06269 USA (e-mail: hongjiang.qian@uconn.edu; gyin@wayne.edu).

Qing Zhang is with the Department of Mathematics, University of Georgia, Athens, GA 30602 USA (e-mail: qz@uga.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TAC.2022.3183147.

Digital Object Identifier 10.1109/TAC.2022.3183147

matters of nonlinear filtering theoretically, the computation of nonlinear filtering remains to be an extremely challenging task. Various efforts on numerical methods for solving these equations have been made. For example, Lototsky *et al.* [16] developed a spectral approach for nonlinear filtering based on the Cameron–Martin version of the Wiener chaos expansion. An advantage of this approach is its separation of the computations involving the observations and those involving the system parameters. The latter is shifted to offline and makes online computation more efficient.

In this article, we focus on general filtering with regime switching, which are diffusions modulated by a continuous-time Markov chain. General filtering with regime switching is a class of challenging problems. There are some efforts along this direction. Early work on discrete-time hybrid models can be found in [2], in which an interactive multiple model algorithm to account for regime switchings was introduced. Subsequent efforts can be found in [6]. They studied a filtering problem with Markovian jumps. Their observation process consists of an image-based sensor for the system mode and a conventional sensor for the state. Under these conditions, they were able to obtain a suboptimal filter; see also [4] for a linear recursive least-square state approach in the absence of the system mode observer.

For recent progress on general filtering, we refer the reader to [9] and [10]. In [9], the authors used Galerkin's approximation to solve a Zakai equation, whereas in [10], the authors considered distributed mean-field filters for traffic networks and developed a scheme decomposing the entire state space into subspaces and performing the distributed filters independently. Their main effort was still on developing approximation methods of infinite-dimensional filtering equations.

We note that the difficulty of using the conditional distribution based filtering is the underlying stochastic differential equations are infinite dimensional. Thus, the aforementioned methods still have to deal with the inherent "curse of dimensionality." This makes the filtering very difficult and challenging for general nonlinear systems.

It is the purpose of this article to develop a new framework with solid foundation, which is a deep neural network (DNN)-based filtering approach. Our emphasis is on obtaining a computationally feasible approach. In this article, we treat general dynamic systems involving randomly-varying switching processes. Such systems are termed hybrid diffusions or switching diffusions. A distinct feature is the coexistence of continuous states and discrete states (also termed discrete events). That is, the systems are running in continuous time, but the states are hybrid involving both continuous and discrete states. Under such a setup, it is computationally difficult or even virtually impossible to directly use the traditional approaches in nonlinear filtering. In this article, we focus on computable methods based on DNNs.

A. Deep Filtering

The recent advent in machine learning and neural computation has promoted the extensive use of NNs. To approximate functions arising from applications using NNs has shown promising outcomes. The essential ideas rely on composition of hidden layers of base functions. Note that a DNN is one with several hidden layers. In this article, we only consider a fully connected NN with no connections between nodes in the same layer. We refer the reader to [20] and [15] for an introduction to deep learning and applications. Note also that stochastic gradient methods play a key role in deep learning optimization when searching for loss-function minimizing network weights. An extensive coverage along this line can be found in [3].

Recently, a new type of DNN-based filtering is developed in [24]. The idea is to generate Monte Carlo samples and then use these samples to train a DNN. The observed data are used as inputs to the DNN and the state trajectories from the Monte Carlo samples are used as the target. The least squares error between the target and calculated output is used as a loss function for the DNN training to generate a weight vector. Then these weight vectors are applied to another set (out-of-sample) of Monte Carlo samples of the actual dynamic system. Such a state estimation procedure is termed a deep filter (DF). For convenience and simplicity, we also refer to the corresponding calculated DNN output as the DF. The DF has the promise as a powerful tool for state estimation thanks to the DNN. Indeed, it was shown in [24] that the DF compares favorably to the traditional Kalman filter, which requires system linearity, Gaussian distributions, and delicate mathematical derivations. The DF, on the other hand, demands none of these. It is remarkable that the DFs can be used to treat switching models with jumps, which cannot be handled by using the usual filtering techniques. In addition, in applications, real data can be used to train the underlying DNN so as to bypass the traditional model calibration. Filtering or state estimation is difficult in general and is often extremely time consuming, whereas the deep filtering techniques alleviate the difficulty and reduce computational complexity, which is promising for handling large-scale systems.

Clearly, the design behind the DF is completely different from the traditional conditional mean estimation philosophy. Rather than searching for conditional mean or distribution of the state observations, the DF approach employs a DNN that is obtained by data-based training through back-propagation and then feed-forwarding filtering mechanisms. A key component of the DF is the recursive algorithm searching for optimal weight parameters that minimizes the corresponding loss function, which in fact, is a finite-dimensional optimization problem.

B. Adaptive LRs

In [24], the training for deep filtering was done using back-propagation with constant LRs. We note that the procedure developed in [24] is only semirecursive or not fully recursive. For recursive computation, it is more desirable to have fully recursive procedures. Note that the constant LR choice is common in some machine learning related works. Nevertheless, if the LR is too small, it may take forever to converge; if too large, it tends to overshoot leading to oscillation divergence. In order to make it work, some initial guess work and preliminary runs are needed to reach suitable LRs. Typically, to make the filtering more efficient, one would apply a larger LR initially and decrease its value over time. In view of this, it is desirable to vary LRs

to adapt the descent of the loss functions over iterations. There are some efforts along this direction. For example, time-based LR schedules and step-based LR schedules were also used in [18], in which the LRs decrease over time and iteration, respectively. In these cases, the LRs are taken to be constants independent of the value of the loss functions. Various adaptive LRs can be found in Ruder [22] including the popular ones (RMSprop and Adam). These LRs are taken to be functions of the gradient of the loss functions in explicit forms. They are structured to fit the need of the specific underlying application. Their design is *ad hoc* in nature. Additional attempts were made to choose LRs adaptively in [21]. In which, the updated LR is proportional to the loss function. It is our observation that a potential drawback of this approach is its reliance on the magnitude of the loss function. Its performance is sensitive to the initial value of the LR when the loss function is large. In deep filtering problems, the noise can be large especially in cases with random switching. In these cases, it is desirable to choose LRs that are adaptive to various noise levels. As noted by LeCun *et al.* [15], it has long been recognized by the learning community that the adaptive learning is an effective approach. Our approach is along this direction focusing on fully recursive stochastic gradient descent algorithms with adaptive LRs that is different from the exiting results in the learning literature. Moreover, although a class of algorithms has been proposed in [24], the asymptotic properties of the deep filtering have not been fully analyzed. There is a need to establish the foundation of DF. The main objective of this article is to set up a solid foundation for a deep filtering methods with adaptive LRs. In particular, we develop a stochastic approximation (SA)-based algorithms by making use of perturbed gradient estimates. We examine the robustness of the approximation to the network parameters and establish convergence of the scheme together with error bounds under suitable conditions. To demonstrate the effectiveness of our filtering schemes, we carry out numerical experiments with nonlinear systems in a multidimensional setting. In contrast to the preliminary version of the DF introduced in [24], the DF algorithm developed in this article is fully recursive with adaptive LRs.

Our main contributions of this article include the following.

1) We develop a brand new framework for general filtering and state estimation. We are able to treat switching diffusions and propose deep learning-based algorithms with adaptive LRs. A key component of the DF is development for fully coupled recursive algorithms searching for optimal weight parameters and optimal LRs that minimize the corresponding loss functions. Treating filtering problems for diffusions modulated by a continuous-time Markov chain is a challenging task. The development of feasible algorithms is scarce and there has been no convergence results provided to date for such problems to the best of authors' knowledge.
2) Our stochastic gradient descent with adaptive LRs is fully recursive and strongly coupled aiming to achieve loss-function minimization in a more systematic way and enabling recursive computation. We carry out the corresponding convergence analysis for the LRs. In particular, we use SA methods to establish the convergence of the LRs and to obtain error bounds on the weight parameters.
3) Rather than dealing with infinite-dimensional conditional means, we are treating a finite-dimensional parameter optimization problem using double recursions. Such a setting enables us to establish robustness results on DNN weight parameters in terms of the adaptive LRs.
4) We focus on filtering with switching models, which is difficult to handle under the conditional mean (or distribution) based approach. In particular, we note that in case of switching diffusions, even if the systems are linear in the state variables, the overall systems are still nonlinear because of the switching. There has been no computational feasible methods for general conditional mean (or distribution) type filtering with switching to date. In these cases, our algorithms provide a viable alternative.

The rest of this article is organized as follows. Section II begins with the setup of filtering problems for a class of switching diffusion processes. In addition to the continuous-time problem, we present a discrete-time approximation scheme. Our main effort is on developing computable results using deep learning machinery, which is presented in Section IV. Under the deep filtering framework that we propose, a main task is to carry out the parameter estimations for the NN weight. We, thus, consider this problem in Section III. This section proposes a novel algorithm that involves adaptive LR approximation and then use the adaptive LR in the approximation of stochastic gradient type procedures for the parameter estimates. Section VI provides several examples on how our algorithms can be implemented. Section VI presents several remarks and possible extension. Finally, the Appendix containing some technical results is placed at the end of this article.

II. Filtering of Switching Diffusions

We consider hybrid nonlinear filtering problems involving a random switching process that is represented by a continuous-time Markov chain $\alpha(\cdot)$. Suppose that $W(\cdot)$ and $\widetilde{W}(\cdot)$ are two independent multidimensional standard Brownian motions. A distinct feature here is that $\alpha(\cdot)$ is a finite-state Markov chain with state space $\mathcal{M} = \{1, \ldots, m_0\}$ and generator $Q = (q_{ij})$ so that $q_{ij} \geq 0$ and $\sum_{j=1}^{m_0} q_{ij} = 0$ for each $i \in \mathcal{M}$. In this article, the switching process $\alpha(t)$ is not directly observable. Our main concern is the estimation of $X(t)$ as the state. So we do not need the information of $\alpha(t)$ or $\alpha(t)$ can be treated as noise. We assume $\alpha(\cdot)$ is independent of both $W(\cdot)$ and $\widetilde{W}(\cdot)$. The system state is $X(t) \in \mathbb{R}^{d_1}$ and the observation is $Y(t) \in \mathbb{R}^{d_2}$. The state and observation pair satisfies the following stochastic differential equations:

$$\begin{cases} dX(t) = f(X(t), \alpha(t))dt + \sigma(X(t), \alpha(t))dW(t) \\ dY(t) = g(X(t), \alpha(t))dt + \sigma_1(X(t), \alpha(t))d\widetilde{W}(t). \end{cases} \tag{1}$$

The inclusion of the switching process makes the filtering task increasingly more difficult or even impossible using the traditional approach.

In this article, our main interest is on developing computable numerical procedures for state estimation. To build a computable

filter algorithm, we work with a discrete-time system of the form

$$\begin{cases} x_{n+1} = x_n + \eta f(x_n, \alpha_n) + \sqrt{\eta}\,\sigma(x_n, \alpha_n) w_n \\ y_{n+1} = y_n + \eta g(x_n, \alpha_n) + \sqrt{\eta}\,\sigma_1(x_n, \alpha_n) v_n \end{cases} \tag{2}$$

where $\eta > 0$ is a small step size, and α_n is a skeleton process of $\alpha(t)$. That is, $\alpha_n = \alpha(\eta n)$ for each n. We take α_n, $\{w_n\}$, and $\{v_n\}$ to be independent random processes, in which $\{w_n\}$ and $\{v_n\}$ can be thought of as the discrete-time "approximations" of the corresponding Brownian motions in (1). It is readily seen that (2) is a discretization of (1) with step size η. We have the following result concerning the approximation to the continuous-time filtering problem.

Proposition 1: Suppose that (1) has a unique (in the sense in distribution) solution for each initial condition. Assume that f, g, σ, and σ_1 have continuous partial derivatives with respect to x up to the second order; $\{w_n\}$ and $\{v_n\}$ are sequences of independent and identically distributed random variables such that $\mathbb{E}w_n = 0$, $\mathbb{E}v_n = 0$, $\mathbb{E}w_n w_n' = I$, and $\mathbb{E}v_n v_n' = I$ (the identity matrix with appropriate dimension). For the approximation (2), define piecewise constant interpolations by

$$x^\eta(t) = x_n,\ y^\eta(t) = y_n,\ \alpha^\eta(t) = \alpha_n \ \text{ for } \ t \in [n\eta, n\eta + \eta). \tag{3}$$

Then, the process $(x^\eta(\cdot), y^\eta(\cdot), \alpha^\eta(\cdot))$ converges weakly to $(X(\cdot), Y(\cdot), \alpha(\cdot))$, which is a solution to (1), where $\alpha(\cdot)$ is the Markov chain generated by Q.

We stated the proposition by using general conditions. An equivalent way to state it as: The martingale problem (see [27]) associated with (1) has a unique solution in the sense in distribution. Note we only need the uniqueness holds in the sense in distribution (not in the strong or pathwise sense). Sufficient conditions that ensure the existence and uniqueness in the pathwise sense can be stated as follows. Suppose that $f(\cdot, \alpha)$, $g(\cdot, \alpha)$, $\sigma(\cdot, \alpha)$, and $\sigma_1(\cdot, \alpha)$ are Lipschitz continuous for each $\alpha \in \mathcal{M}$. Because proving Proposition 1 is not our main line of work here and because our earlier work already covers the main idea, only ideas of proof is sketched in the Appendix. We mainly show the discrete iterations converge to the switching diffusion; some more details on switching diffusions can be found in [27].

III. Estimation of NN Parameters

To use machine learning techniques for filtering, there are two main ingredients. One of them is the use of NN parameter θ and the other is the selection of LR ρ. Note that in our case, the NN parameters are in finite dimensional spaces. A key in our procedure is to find the optimal parameters. Nowadays, a commonly used procedure for estimating parameter θ is a stochastic gradient type algorithm, rooted to the methods of SA. The procedure uses a scheme based on noisy gradient estimates of a loss function $\nabla\overline{J}(\theta)$. SA is a well studied subject initiated in the early 1950s for root findings and stochastic optimization. A state-of-the-art treatment of SA can be found in [14]. The idea is to choose the parameters of the neural net so that a loss function (objective function) $\overline{J}(\theta)$ is minimized. To do this, let $\nabla J(\theta, \xi_k)$ be the gradient estimates of $\nabla\overline{J}(\theta)$ at time k with $\{\xi_k\}$ denoting a sequence of observation noise. Following such an idea, a recursive algorithm to estimate θ can be constructed as follows:

$$\theta_{k+1} = \theta_k - \rho\nabla J(\theta_k, \xi_k) \tag{4}$$

where $\rho > 0$ is a selected LR. We write (4) as a general scheme with nonadditive noise. A simple example is $\nabla J(\theta, \xi) = \nabla\overline{J}(\theta) + \xi$. That is, $\overline{J}(\theta)$ observed with an additive noise ξ. Nevertheless, (4) is much more general than the additive noise model.

In the existing machine learning literature, a constant LR is commonly used. However, there were few systematic treatments of how to choose the LR beyond the work of [21]. There have been no in-depth serious mathematical analysis on properties of algorithms with adaptive LRs. To analyze (4), we can use the methods of [14]. The analysis is based on martingale averaging methods. We show that as $\rho \to 0$, an interpolated sequence $\theta^\rho(t) = \theta_k$ for $t \in [k\rho, k\rho + \rho)$ has a limit (in the sense of weak convergence) so that the limit is a solution of a martingale problem. From a dynamic system point of view, the limit is a solution of an ordinary differential equation. The abovementioned analysis is based on the scaling $\rho \to 0$, $k \to \infty$, and $k\rho$ "matches" the continuous time t. Then, we can use stability argument to show that when $k \to \infty$, $\rho \to 0$, but $k\rho \to \infty$ to get the convergence to the minimizer of the cost function. We will return to this point at a later time.

Nevertheless, we note that in practice, one normally does not use an LR that goes to 0. Rather, one uses a fixed constant LR as it has been done in the machine learning literature. It is clear that not all LRs are equal. Some of them are better than the other experimentally. A usual practice is to choose the LRs by trial and error.

In this article, we propose a systematic approach, namely, an adaptive strategy. The idea is that while we are updating the parameter θ, we also adaptively update the LRs. Then, updated learning rates are used in the parameter estimation for θ. Thus, in lieu of a single process in the iteration, we have two sequences (a pair of sequences for estimating (θ, ρ)) simultaneously. In the process of updating, θ and ρ are not changing at the same scale or frequency. In fact, we update θ more often than that ρ. To reflect this, we propose an algorithm of the following form. We allow the function to be varying with the NN parameter θ. For our machine learning scheme, rather than assuming that the LR is a fixed constant as in the most of the existing works, we will generate a sequence of LRs $\{\rho_n\}$. The sequence is generated so as to minimize another objective function $\overline{\chi}$ that depends on the LR. Similar to getting the estimates $\{\theta_k\}$, we use a sequence of noisy gradient estimates of $\overline{\chi}(\rho, \theta_e)$ (to be specified later) with a time-dependent cost function with $\{\zeta_n\}$ denoting the observation noise. The quantity θ_e is a constant value returned from estimates of θ parameter. Our estimation algorithms are still of stochastic gradient type. The procedure involves two levels. One of them is the estimate of θ, whereas the other is the estimate for the LR. We denote by $\theta_k^{n\ell}$ the iterate within the epoch and ρ_n the iterate for the LR across the epoch. Denote $\theta_k^{n\ell} = \theta_{n\ell+k}$ for $k \le \ell$. So in particular, $\theta_\ell^{n\ell} = \theta_{n\ell+\ell}$ (similar notation for $\{\xi_k^{n\ell}\}$). Suppose that the noisy gradients are available to us. Then, we

consider the following stochastic gradient procedures:

$$\begin{aligned}\theta_{k+1}^{n\ell} &= \theta_k^{n\ell} - \rho_n \nabla_\theta J(\theta_k^{n\ell}, \xi_k^{n\ell}),\ k = 0, \ldots, \ell - 1 \\ \rho_{n+1} &= \rho_n - \varepsilon \widehat{G}_n\end{aligned} \tag{5}$$

where $\nabla_\theta J(\theta_k^{n\ell}, \xi_k^{n\ell})$ denotes the noisy gradients of J w.r.t. θ, $\varepsilon > 0$ is a small parameter serving as a stepsize, and $\widehat{G}_n$ denotes a sequence of time-varying (n dependent) estimates of the partial derivative w.r.t. ρ of the loss function $(\partial/\partial\rho)\overline{\chi}(\rho, \theta_e)$. The $\widehat{G}_n$, in fact, depends on ρ, θ_e a constant value within each epoch, and a sequence of noise processes, which we will specify shortly. The idea is that we choose a loss function so that the noisy gradient estimate $\nabla_\theta J(\theta_k^{n\ell}, \xi_k^{n\ell})$ is available (e.g., we may choose a quadratic (in θ) function). For the iterates of ρ_n, however, because of large amount of data is used, the form of the gradient or partial derivatives with respect to ρ is not readily available. Thus, we have to use a noisy finite difference to approximate the partial derivative.

Remark 2: The rationale of the construction can be illustrated as follows. As far as the practical algorithm is concerned, we consider that ℓ as the number of iterations within the nth epoch. We are developing an adaptive LR scheme, so ρ_n is changing (decreasing). However, for the first recursion $\theta_k^{n\ell}$, the ρ_n is a fixed constant LR. Starting from a preselected $\rho_0 > 0$ and initial value $\theta_k^0 = \theta_0^{0\ell} = \theta_0$, we can start the iteration to get θ_k^0 for $k = 0, \ldots, \ell$ with ρ_0 used. Then, a computed constant value $\theta_e = \theta_\ell^0$ is used in the second iteration for calculating ρ_1. This computed value of ρ_1 is then used in the approximation θ_k^ℓ. It returns a value θ_ℓ^ℓ (set as θ_e) to be used in the next estimate of ρ_2 and so on. That is, within each epoch, we use a constant LR.

Because of the second iteration in (5), the LR, in fact, is changing in accordance with a gradient descent procedure. We are searching for the optimal LR using the second recursion. Note that $\widehat{G}_n$ depends on θ_e that is a constant but varies for different epochs.

We use $\chi_n(\rho, \theta_e, \zeta_n)$ to denote a sequence of estimates of $\overline{\chi}(\rho, \theta_e)$, where ζ_n denotes the observation noise. In fact, because we are using a noisy finite difference, we use $\{\zeta_n^\pm\}$ ($\{\zeta_n^+\}$ and $\{\zeta_n^-\}$) two sequences of observation noise processes, which are in the nonadditive form following the usual finite difference approximation. In addition, there is another additive noise sequence $\{\varpi_n\}$ that is independent of $\{\zeta_n^\pm\}$. We write $\widehat{G}_n$ in detail as follows:

$$\widehat{G}_n = \frac{\chi_n^+ - \chi_n^-}{2\delta} + \frac{\varpi_n}{2\delta} \tag{6}$$

where $\{\varpi_n\}$ is a sequence of martingale difference noise

$$\chi_n^\pm(\rho, \zeta) = \chi_n(\rho \pm \delta, \theta_e, \zeta^\pm) \tag{7}$$

$\delta = \delta_\varepsilon > 0$ is the finite difference parameter satisfying $\delta_\varepsilon \to 0$ as $\varepsilon \to 0$ but $(\varepsilon/\delta_\varepsilon^2) \to 0$ [e.g., we may choose $\delta_\varepsilon = \varepsilon^{1/6}$.] So in particular, we have $\chi_n^\pm(\rho_n, \zeta_n) = \chi_n(\rho_n \pm \delta_\varepsilon, \theta_e, \zeta_n^\pm)$. Define

$$\begin{aligned}b_n &= \frac{\overline{\chi}(\rho_n + \delta, \theta_e) - \overline{\chi}(\rho_n - \delta, \theta_e)}{2\delta_\varepsilon} - \frac{\partial \overline{\chi}(\rho_n, \theta_e)}{\partial \rho} \\ \widetilde{\chi}_n(\rho, \zeta^\pm) &= [\chi_n^+(\rho, \theta_e, \zeta^+) - \overline{\chi}(\rho + \delta_\varepsilon, \theta_e)] \\ &\quad - [\chi_n^-(\rho, \theta_e, \zeta^-) - \overline{\chi}(\rho - \delta_\varepsilon, \theta_e)].\end{aligned} \tag{8}$$

Then, the second recursion in (5) can be written as

$$\rho_{n+1} = \rho_n - \varepsilon \frac{\partial \overline{\chi}(\rho_n, \theta_e)}{\partial \rho} - \varepsilon b_n - \frac{\varepsilon}{2\delta_\varepsilon} \widetilde{\chi}_n(\rho_n, \zeta_n^\pm) - \frac{\varepsilon}{2\delta_\varepsilon} \varpi_n. \tag{9}$$

The interpretation of (9) is that b_n can be considered as a bias term, and the last two terms in (9) can be considered as noise term, in which $\widetilde{\chi}_n$ represents nonadditive noise, whereas ϖ_n represents additive noise.

To proceed, we state the conditions needed for our recursive algorithms.

(A1)The $\{\varpi_n\}$ is a sequence of martingale difference noise that is independent of $\{\xi_n\}$ and $\{\zeta_n\}$ satisfying $\mathbb{E}|\varpi_n|^2 < \infty$; $\{\xi_n\}$ and $\{\zeta_n\}$ are bounded sequences of noises such that

a) for each i and ℓ, and each $\theta \in \mathbb{R}^d$, $\{\nabla_\theta J(\theta, \xi_n^{i\ell})\}$ is a bounded stationary ϕ-mixing sequence with appropriate mixing rate such that

$$\mathbb{E}\nabla_\theta J(\theta, \xi_n^{i\ell}) = \nabla_\theta \overline{J}(\theta)$$

b) for each n and each $\rho \in \mathbb{R}$, $\{\chi_n(\rho, \theta_e, \zeta_n^\pm)\}$ are bounded stationary sequences of ϕ-mixing processes with mixing rate $\widetilde{\psi}(n)$ such that

$$\sum_n \widetilde{\psi}(n) < \infty$$

and that there is a continuous function $\overline{\chi}(\cdot)$ satisfying

$$\mathbb{E}\chi_i(\rho, \theta_e, \zeta_i^\pm) = \overline{\chi}(\rho, \theta_e).$$

(A2) For each n, each ξ, and each ζ, $\nabla_\theta J(\cdot, \xi)$, $\overline{\chi}(\cdot, \theta_e)$, and $\chi_n(\cdot, \theta_e, \zeta^\pm)$ have continuous partial derivatives up to the second order w.r.t. θ and ρ, respectively.

(A3) The following conditions hold.

a) The differential equation

$$\dot{\theta}(t) = -\nabla_\theta \overline{J}(\theta(t)) \tag{10}$$

has a unique solution for each initial condition.

b) The differential equation

$$\dot{\rho}(t) = -\frac{\partial}{\partial \rho} \overline{\chi}(\rho(t), \theta_e) \tag{11}$$

has a unique solution for each initial condition.

Remark 3: It is well known that if a sequence is stationary ϕ-mixing, then it is strongly ergodic. As a result, condition (A1) implies that for each positive integer m, as $n \to \infty$

$$\frac{1}{n} \sum_{j=m}^{n+m-1} \mathbb{E}_m \nabla_\theta J(\theta, \xi_j) \to \nabla_\theta \overline{J}(\theta) \text{ in probability.} \tag{12}$$

In fact, the abovementioned convergence also takes place without the conditional expectation $\mathbb{E}_m$ because of the strong ergodicity. In addition, in view of the well-known mixing inequality [8, p. 349], for some $K > 0$

$$\begin{aligned}&|\mathbb{E}_m \chi_{k+m}(\rho, \theta_e, \zeta_{k+m}^\pm)| \\ &\quad = |\mathbb{E}_m \chi_{k+m}(\rho, \theta_e, \zeta_{k+m}^\pm) - \mathbb{E}\chi_{k+m}(\rho, \theta_e, \zeta_{k+m}^\pm)| \\ &\quad \le K\widetilde{\psi}(k)\end{aligned} \tag{13}$$

because $\mathbb{E}\chi_{k+m}(\rho,\theta_e,\zeta^{\pm}_{k+m})=0$.

For generality, we assumed that the $\widehat{G}_n$ is dependent of n and also includes both nonadditive noise and additive noise. In the simplest case, $(\partial/\partial\rho)\overline{\chi}(\rho,\theta_e)+$ noise. However, our setup is far more general.

Note that (A3) essentially requires that the associated (degenerate) martingale problems have a unique solution. It is degenerate because no second-order term is involved in the operator (to be given later). Note that we only need the uniqueness in the sense of in distribution.

IV. Asymptotic Properties of the Adaptive Learning Algorithm

This section is devoted to the convergence rate of the adaptive learning algorithm. For convenience, we suppress the θ-dependence in $\nabla_\theta J^{n\ell}_k$ and write it simply as $\nabla J^{n\ell}_k$ instead in what follows. Because the iterations for $\theta^{n\ell}_k$ use a constant LRs that do not go to 0 within each epoch, we begin our analysis by examining the algorithm for the LR ρ_n first.

A. Convergence of ρ_n

We shall use weak convergence methods to establish the convergence property. To begin, define a piecewise constant interpolation

$$\rho^\varepsilon(t)=\rho_n,\ \text{ for } t\in[n\varepsilon,n\varepsilon+\varepsilon). \tag{14}$$

To proceed, we first work on the convergence of $\{\rho^\varepsilon(\cdot)\}$. Note that under this interpolation, we may examine more closely the martingale difference noise term. If we define

$$M^\varepsilon(t)=\frac{\varepsilon}{2\delta_\varepsilon}\sum_{j=0}^{t/\varepsilon-1}\varpi_j$$

$$\mathbb{E}|M^\varepsilon(t)|^2=\frac{\varepsilon}{\delta^2_\varepsilon}\sum_{j=0}^{t/\varepsilon-1}\varepsilon\mathbb{E}|\varpi_j|^2\to 0\ \text{ as }\ \varepsilon\to 0.$$

Thus, by the familiar martingale inequality

$$P\left(\sup_{t\le T}|M^\varepsilon(t)|\ge\widetilde{\mu}\right)\le\frac{\mathbb{E}|M^\varepsilon(T)|^2}{\widetilde{\mu}^2}\to 0\ \text{ as }\ \varepsilon\to 0. \tag{15}$$

Since we do not assume *a prior* that the sequence generated by (5) is bounded, we need to use a truncation device [14, p. 284]. Let N be a fixed but otherwise arbitrary number satisfying $N>0$. We focus on a truncated version of the iterates ρ^N_n defined by

$$\begin{aligned}\rho^N_{n+1}=\rho^N_n-\varepsilon\frac{\partial\overline{\chi}(\rho^N_n,\theta_e)}{\partial\rho}T^N(\rho^N_n)\\ -\varepsilon b_nT^N(\rho^N_n)-\frac{\varepsilon}{2\delta}\widetilde{\chi}_n(\rho_n,\zeta^{\pm}_n)T^N(\rho^N_n)-\frac{\varepsilon}{2\delta}\varpi_n\end{aligned} \tag{16}$$

where $T^N(\cdot)$ is a sufficiently smooth truncation function defined by

$$T^N(\rho)=\begin{cases}1 & \text{if } \rho\in[-N,N]\\ 0 & \text{if } \rho\in\mathbb{R}-[-(N+1),N+1].\end{cases}$$

Note that we only truncate ρ_n not the noise ζ_n. Define $\rho^{\varepsilon,N}(t)=\rho^N_n$ for $t\in[\varepsilon n,\varepsilon n+\varepsilon)$. Then, $\rho^{\varepsilon,N}(t)=\rho^\varepsilon(t)1_{\{t\le\tau\}}$ with τ being the first exit time from $[-N,N]$ (i.e., $\tau=\min\{t:\rho^\varepsilon(t)\notin[-N,N]\}$). That is, $\rho^{\varepsilon,N}(t)$ is equal to $\rho^\varepsilon(t)$ up until the first exit from $[-N,N]$. Thus, by the definition [14, p. 284], $\rho^{\varepsilon,N}(\cdot)$ is an N-truncation of $\rho^\varepsilon(\cdot)$. In lieu of $\rho^\varepsilon(\cdot)$, we focus on $\{\rho^{\varepsilon,N}(\cdot)\}$. We shall first establish the tightness of $\{\rho^{\varepsilon,N}(\cdot)\}$, and proceed with the weak convergence of the sequence. Then, we will let $N\to\infty$ to complete the proof. In what follows, for notation simplicity, we use t/ε to denote $\lfloor t/\varepsilon\rfloor$, the integer part of t/ε. without using the floor function notation.

Lemma 4: Assuming $\rho^\varepsilon_0=\rho_0$ and conditions (A1)–(A3), $\rho^{\varepsilon,N}(\cdot)$ converges weakly to $\rho^N(\cdot)$ such that $\rho^N(\cdot)$ is a solution of (11) with initial condition $\rho^N(0)=\rho_0$ and with $(\partial\overline{\chi}/\partial\rho)$ replaced by $\partial\overline{\chi}^N/\partial\rho$.

Remark 5: Note that in view of (A3), the associated martingale problem with operator L is given by

$$Lh(\rho)=-\frac{\partial\overline{\chi}(\rho,\theta_e)}{\partial\rho}\frac{dh(\rho)}{d\rho} \tag{17}$$

where $h(\cdot)$ is a suitably smooth function, (e.g., a continuously differentiable function with compact support).

For simplicity, we have chosen the initial condition to be independent of ε. We could replace this condition by allowing ρ_0 to be ε dependent. In this case, we will assume that ρ^ε_0 converges weakly to ρ_0. All the subsequent arguments still carry over.

Proof of Lemma 4 : In view of Remark 5, we shall show that $\rho^N(\cdot)$ is a solution of the martingale problem with operator

$$\begin{aligned}L^Nh(\rho):=-\frac{\partial\overline{\chi}^N(\rho,\theta_e)}{\partial\rho}\frac{dh(\rho)}{d\rho}\ \text{ where}\\ \frac{\partial\overline{\chi}^N(\rho,\theta_e)}{\partial\rho}=\frac{\partial\overline{\chi}(\rho,\theta)}{\partial\rho}T^N(\rho).\end{aligned} \tag{18}$$

To proceed, we first show that $\{\rho^{\varepsilon,N}(\cdot)\}$ is tight in the space of functions that are right continuous having left limits endowed with the Skorohod metric (see [14, Ch. 7] for various definitions and notion of weak convergence). Note that weak convergence is a generalization of convergence in distribution.

Step 1: We first show that $\{\rho^{\varepsilon,N}(\cdot)\}$ is tight in $D([0,\infty]:\mathbb{R})$, where $D([0,\infty]:\mathbb{R})$ denotes the space of functions defined on $[0,\infty)$ taking values in $\mathbb{R}$, which are right continuous, have left limits, endowed with the Skorohod metric (equivalently, the sequence is sequentially compact).

Use $\mathbb{E}_n$ to denote the conditional expectation with respect to the σ-algebra generated by $\mathcal{F}_n=\{\zeta_j,\varpi_j:j<n\}$. Fix any $\Lambda(\cdot)\in C^2_0$ (C^2 function with compact support), it is readily seen that

$$\begin{aligned}&\mathbb{E}_n\Lambda(\rho^N_{n+1})-\Lambda(\rho^N_n)\\ &=-\varepsilon\Lambda_\rho(\rho^N_n)\frac{\partial\overline{\chi}(\rho^N_n,\theta_e)}{\partial\rho}T^N(\rho^N_n)-\varepsilon\Lambda_\rho(\rho^N_n)b_nT^N(\rho^N_n)\\ &\quad-\frac{\varepsilon}{2\delta}\Lambda_\rho(\rho^N_n)\widetilde{\chi}_n(\rho_n,\zeta^{\pm}_n)T^N(\rho^N_n)+O\left(\frac{\varepsilon^2}{\delta^2_\varepsilon}\right).\end{aligned} \tag{19}$$

In the abovementioned, we used the fact that $\mathbb{E}_n\varpi_n/(2\delta_\varepsilon)=0$. Using (8), calculation of the bias term leads to

$$\begin{aligned} & b_nT^N(\rho_n^N) \\ &= \left[\frac{\overline{\chi}(\rho_n^N+\delta_\varepsilon,\theta_e)-\overline{\chi}(\rho_n^N-\delta_\varepsilon,\theta_e)}{2\delta_\varepsilon}-\partial\overline{\chi}(\rho_n,\theta_e)\right]T^N(\rho_n^N) \\ &= O\left\{\left[\frac{\overline{\chi}_{\rho\rho}(\rho_n^+,\theta_e)\delta_\varepsilon^2}{\delta_\varepsilon}\right]T^N(\rho_n^N)\right\}=O(\delta_\varepsilon) \end{aligned} \tag{20}$$

where ρ_n^+ is on the line segment joining ρ_n and ρ_{n+1}. Because $\{\zeta_n^\pm\}$ are stationary ϕ-mixing sequences, by virtue of (13), for each $0<T<\infty$ and $t<T$, it is easily verified that

$$\frac{\varepsilon}{2\delta_\varepsilon}\sum_{j=t/\varepsilon}^{T/\varepsilon}\mathbb{E}_{t/\varepsilon}\widetilde{\chi}_j(\rho,\zeta_j^\pm)T^N(\rho)\to 0 \text{ in probability as } \varepsilon\to 0. \tag{21}$$

[We are taking $n=t/\varepsilon$ with the convention that t/ε is meant to be the integer part of t/ε. In the rest of the tightness part of the proof, we will use either n or t/ε whichever is more convenient for us.] Define a perturbation by

$$\Lambda_1^\varepsilon(\rho,t)=-\frac{\varepsilon}{2\delta_\varepsilon}\Lambda_\rho(\rho)\sum_{j=t/\varepsilon}^{T/\varepsilon}\mathbb{E}_{t/\varepsilon}\widetilde{\chi}_j(\rho,\zeta_j^\pm)T^N(\rho). \tag{22}$$

Recalling that $n=t/\varepsilon$

$$\sup_{n\le T/\varepsilon}\mathbb{E}|\Lambda_1^\varepsilon(\rho_n^N,n\varepsilon)|\to 0 \text{ as } \varepsilon\to 0.$$

Now, define the perturbed smooth function $\Lambda^\varepsilon(\rho)=\Lambda(\rho)+\Lambda_1^\varepsilon(\rho,n\varepsilon)$. For any $\Xi\in C_0^2$, define an operator $\widehat{A}^\varepsilon$ as

$$\widehat{A}^\varepsilon\Xi(\rho_n^N)=\mathbb{E}_n\Xi(\rho_{n+1}^N)-\Xi(\rho_n^N).$$

It can be seen that

$$\begin{aligned} \widehat{A}^\varepsilon\Lambda_1^\varepsilon(n\varepsilon) &:= \widehat{A}^\varepsilon\Lambda_1^\varepsilon(\rho_n^N,n\varepsilon) \\ &= \frac{\varepsilon}{2\delta}\Lambda_\rho(\rho_n^N)\widetilde{\chi}_n(\rho_n,\zeta_n^\pm)T^N(\rho_n^N)+o(1) \end{aligned} \tag{23}$$

where $o(1)\to 0$ in probability as $\varepsilon\to 0$. Note that the first term on the second line of (23) cancels with the first term on the third line of (19). By virtue of (19), (20), the boundedness due to the truncation, the boundedness of the noise $\{\zeta_n^\pm\}$, the martingale difference noise $\{\varpi_n\}$, and the definition of $\Lambda^\varepsilon(t)$, it can be shown as in [13, Th. 2, p. 68] that $\{\widehat{A}^\varepsilon\Lambda^\varepsilon(\cdot)\}$ is tight. Thus, by Lemmas 5 and 7 on [13, pp. 50–51], $\{\rho^{\varepsilon,N}(\cdot)\}$ is tight as desired.

Step 2: We next proceed to characterize the limit process. Because $\{\rho^{\varepsilon,N}(\cdot)\}$ is tight, it is sequentially compact (see [14, Ch. 7]). Thus, we can select a convergent subsequence. Pick out such a subsequence, and for simplicity still use ε as its index with a limit denoted by $\rho^N(\cdot)$. We proceed to characterize the limit process. By Skorohod representation, in an enlarged probability space, we can find $\widetilde{\rho}^{\varepsilon,N}(\cdot)$ and $\widetilde{\rho}^N(\cdot)$ which are equal to $\rho^{\varepsilon,N}(\cdot)$ and $\rho^N(\cdot)$, respectively, with probability one (w.p.1) such that $\widetilde{\rho}^{\varepsilon,N}(\cdot)$ converges to $\widetilde{\rho}^N(\cdot)$ w.p.1 in the enlarged probability space. With a slight abuse of notation, we may assume that the sequence itself $\rho^{\varepsilon,N}(\cdot)\to\rho^N(\cdot)$ w.p.1, and the convergence is uniform on each bounded set.

To show that $\rho^N(\cdot)$ is the solution of a martingale problem with operator L^N, we need only show that for any $h\in C_0^1$, continuously differentiable functions with compact support, any bounded and continuous function $H(\cdot)$, any positive integer κ, any $t,s>0$, $\iota\le\kappa$, and $t_\iota\le t$

$$\mathbb{E}H(\rho^N(t_\iota):\iota\le\kappa)\Big[h(\rho^N(t+s))-h(\rho^N(t)) - \int_t^{t+s}L^Nh(\rho^N(u))du\Big]=0. \tag{24}$$

To verify (24), we work on the sequence $\rho^{\varepsilon,N}(\cdot)$. Using the weak convergence and the Skorohod representation, we have

$$\begin{aligned} &\lim_{\varepsilon\to 0}\mathbb{E}H(\rho^{\varepsilon,N}(t_\iota):\iota\le\kappa)[h(\rho^{\varepsilon,N}(t+s))-h(\rho^{\varepsilon,N}(t)] \\ &=\mathbb{E}H(\rho(t_\iota):\iota\le\kappa)[h(\rho^N(t+s))-h(\rho^N(t))]. \end{aligned} \tag{25}$$

Choose a sequence $\{k_\varepsilon\}$ such that $k_\varepsilon\to\infty$ and $\delta_\varepsilon k_\varepsilon\to\infty$ as $\varepsilon\to 0$ but $\Delta_\varepsilon=\varepsilon k_\varepsilon\to 0$ as $\varepsilon\to 0$. Subdividing the interval $[t/\varepsilon,(t+s)/\varepsilon)$ by using Δ_ε, we have

$$\begin{aligned} & h(\rho^{\varepsilon,N}(t+s))-h(\rho^{\varepsilon,N}(t)) \\ &= \sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon}[h(\rho_{jk_\varepsilon+k_\varepsilon})-h(\rho_{jk_\varepsilon})] \\ &= \sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon}\sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1}\frac{dh(\rho_{jk_\varepsilon})}{d\rho}\sum_{\iota=1}^4L_{k,\iota} \end{aligned}$$

where

$$\begin{aligned} L_{k,1}&=L_{k,1}(\rho)=-\varepsilon\frac{\partial\overline{\chi}(\rho,\theta_e)}{\partial\rho}T^N(\rho) \\ L_{k,2}&=-\varepsilon b_kT^N(\rho_k^N) \\ L_{k,3}&=L_{k,3}(\rho)=-\frac{\varepsilon}{2\delta}\widetilde{\chi}_k(\rho,\zeta_k^\pm)T^N(\rho) \\ L_{k,4}&=-\frac{\varepsilon}{2\delta}\varpi_k. \end{aligned} \tag{26}$$

By virtue of the estimates on the bias (20) and the martingale difference noise ϖ_n, it is readily seen that

$$\begin{aligned} &\lim_{\varepsilon\to 0}\mathbb{E}H(\rho^{\varepsilon,N}(t_\iota):\iota\le\kappa) \\ &\quad\times\sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon}\sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1}\frac{dh(\rho_{jk_\varepsilon})}{d\rho}L_{k,2}=0 \\ &\lim_{\varepsilon\to 0}\mathbb{E}H(\rho^{\varepsilon,N}(t_\iota):\iota\le\kappa) \\ &\quad\times\sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon}\sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1}\frac{dh(\rho_{jk_\varepsilon})}{d\rho}L_{k,4}=0. \end{aligned} \tag{27}$$

Letting $\varepsilon jk_\varepsilon\to u$, then for each k satisfying $jk_\varepsilon\le k<jk_\varepsilon+k_\varepsilon$, $\varepsilon k\to u$. Thus, the continuity of $(\partial\chi^N/\partial\rho)(\cdot,\theta_e,\zeta)$ implies that $(\partial\chi^N/\partial\rho)(\rho_k,\theta_e,\zeta_k)$ can be replaced by $(\partial\chi^N/\partial\rho)(\rho_{jk_\varepsilon},\theta_e,\zeta_k)$ with an error term going to 0 in probability. As a result, we have

$$\begin{aligned} &\lim_{\varepsilon\to 0}\mathbb{E}H(\rho^{\varepsilon,N}(t_\iota):\iota\le\kappa)[h(\rho^{\varepsilon,N}(t+s))-h(\rho^{\varepsilon,N}(t))] \\ &=\lim_{\varepsilon\to 0}\mathbb{E}H(\rho^{\varepsilon,N}(t_\iota):\iota\le\kappa) \end{aligned}$$

$$\times \left[\sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} [L_{k,1}(\rho^N_k) + L_{k,3}(\rho^N_k)] \right]$$

$$= \lim_{\varepsilon\to 0} \mathbb{E} H(\rho^{\varepsilon,N}(t_\iota) : \iota \le \kappa) \times \left[\sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} \times [L_{k,1}(\rho^N_{jk_\varepsilon}) + L_{k,3}(\rho^N_{jk_\varepsilon})] \right]. \tag{28}$$

It can be shown that

$$\lim_{\varepsilon\to 0} \mathbb{E} H(\rho^{\varepsilon,N}(t_\iota) : \iota \le \kappa) \times \left[\sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} L_{k,1}(\rho^N_{jk_\varepsilon}) \right] = \mathbb{E} H(\rho^N(t_\iota) : \iota \le \kappa) \left[- \int_t^{t+s} \frac{dh(\rho^N)}{d\rho} \frac{\partial \overline{\chi}^N(\rho^N(u))}{\partial \rho} \right]. \tag{29}$$

As for $L_{k,3}(\rho^N_{jk_\varepsilon})$, we have

$$- \frac{\varepsilon}{2\delta_\varepsilon} \sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} \widetilde{\chi}_k(\rho^N_{jk_\varepsilon}, \zeta^\pm_k) = -\frac{1}{2} \sum_{j=t/\Delta_\varepsilon}^{(t+s)/\Delta_\varepsilon} \Delta_\varepsilon \Gamma^\varepsilon_{jk_\varepsilon} + o(1)$$

where $o(1) \to 0$ in probability and

$$\Gamma^\varepsilon_{jk_\varepsilon} = \frac{1}{\delta_\varepsilon k_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} \widetilde{\chi}_k(\rho^N_{jk_\varepsilon}, \zeta^\pm_k).$$

We claim that

$$\Gamma^\varepsilon_{jk_\varepsilon} \to 0 \text{ in probability as } \varepsilon \to 0. \tag{30}$$

To see this, we compute the moment of $\Gamma^\varepsilon_{jk_\varepsilon}$. We have

$$\begin{aligned} &\mathbb{E}|\Gamma^\varepsilon_{jk_\varepsilon}| \\ &= \frac{1}{\delta_\varepsilon k_\varepsilon} \mathbb{E} \Big| \frac{dh(\rho^N_{jk_\varepsilon})}{d\rho} \Big| \Big| \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \widetilde{\chi}_n(\rho^N_{jk_\varepsilon}, \zeta^\pm_k) \Big| \\ &\le \frac{K}{\delta_\varepsilon k_\varepsilon} \mathbb{E} \Big| \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \mathbb{E}_{jk_\varepsilon} \widetilde{\chi}_n(\rho^N_{jk_\varepsilon}, \zeta^\pm_k) \Big| \\ &\le \frac{K}{\delta_\varepsilon k_\varepsilon} \sum_{k=jk_\varepsilon}^{jk_\varepsilon+k_\varepsilon-1} \widetilde{\psi}(k - jk_\varepsilon) \\ &\le \frac{K}{\delta_\varepsilon k_\varepsilon} \to 0. \end{aligned} \tag{31}$$

In the abovementioned, we used K as a generic positive constant, whose value may change for different usage. We also used $\mathbb{E}\chi_n(\rho^N_{jk_\varepsilon}, \zeta^\pm_k) = 0$, the well-known mixing inequality with the mixing measure $\widetilde{\psi}(k)$, the summability of $\widetilde{\psi}(k)$, and the choice of k_ε. In view of (31), (30) follows.

Now, we combine the estimates (25)–(30) to obtain

$$\lim_{\varepsilon\to 0} \mathbb{E} H(\rho^{\varepsilon,N}(t_\iota) : \iota \le \kappa)[h(\rho^{\varepsilon,N}(t+s)) - h(\rho^{\varepsilon,N}(t)] = \mathbb{E} H(\rho^N(t_\iota) : \iota \le \kappa)[\int_t^{t+s} L^N h(\rho^N(u)) du]. \tag{32}$$

By virtue of (25)–(32), and noting (18), we arrive at (24). Thus, Lemma 4 is proved. ■

Theorem 6: Under the conditions of Lemma 4, $\rho^\varepsilon(\cdot)$ converges weakly to $\rho(\cdot)$ such that $\rho(\cdot)$ is the solution of (10).

Outline of Proof. We have shown that the truncated process $\rho^{\varepsilon,N}(\cdot)$ converges to $\rho^N(\cdot)$. Here, we need to show that the untruncated process is also convergent. The argument is very similar to [13, p. 44, Corollary], so we will only indicate the main steps but leave the details out. Denote by $P^{\rho_0}(\cdot)$ and $P^N(\cdot)$ the measures induced by $\rho(\cdot)$ and $\rho^N(\cdot)$, respectively. The measure $P^{\rho_0}(\cdot)$ is unique by (A3). Thus, for each $T < \infty$, $P^{\rho_0}(\cdot)$ agrees with $P^N(\cdot)$ on all Borel subsets of the set of paths in $D([0,\infty) : \mathbb{R})$ with values in $[-N, N]$ for $t \le T$. However, $P^{\rho_0}(\sup_{t\le T} |\rho(t)| \le N) \to 1$ as $N \to \infty$. The abovementioned together with $\rho^{\varepsilon,N}(\cdot)$ converges to $\rho^N(\cdot)$ weakly implies that $\rho^\varepsilon(\cdot)$ converges weakly to $\rho(\cdot)$. The uniqueness of the ODE then implies the desired results. This completes the outline of the proof.

So far, the analysis is based on the scaling $\varepsilon \to 0$, $n \to \infty$ and εn remains to be bounded (εn is essentially the continuous time t). To proceed, we consider the case that $\varepsilon \to 0$ and $n \to \infty$ such that $\varepsilon n \to \infty$. We state a result in the following proposition.

Proposition 7: Suppose that the conditions of Theorem 6 hold; (11) has a unique stationary point ρ^* that is stable in the sense of Lyapunov; $\{\rho_n\}$ is bounded in probability in $\mathbb{R}$. Then, $\rho^\varepsilon(t_\varepsilon + \cdot)$ converges weakly to ρ^* as $\varepsilon \to 0$ and $t_\varepsilon \to \infty$.

Remark 8: Note that for simplicity, we have assumed that $\{\rho_n\}$ is bounded in probability. Sufficient conditions can be derived, which are similar to Theorem 10. We omit the details, however.

We provide a discussion on the main ideas of proof below. Let us choose $T > 0$ and consider a convergent subsequence of the pair of processes $\{\rho^\varepsilon(t_\varepsilon + \cdot), \rho^\varepsilon(t_\varepsilon - T + \cdot)\}$, with limit denoted by $(\rho(\cdot), \rho_T(\cdot))$. We have $\rho(0) = \rho_T(T)$. The value of $\rho_T(0)$ is not known, but all the possible such $\rho_T(0)$, over all T and convergent subsequences, belong to a set that is tight. This together with the stability condition, for any $\delta > 0$ there is a $T_\delta < \infty$ satisfying for all $T \ge T_\delta$, $\rho_T(T)$ will be in a δ-neighborhood of ρ^* with probability $\ge 1 - \delta$ yielding the desired assertion.

Before we analyze the iteration for $\theta^{n\ell}_k$ in (5), we first present a result on the analysis of (4). Because the proof is very similar to Theorem 6 and Proposition 7, the verbatim proofs are omitted.

Theorem 9: Consider the auxiliary algorithm (4). The following results hold.

a) Under the conditions of Theorem 6, with $\theta^\rho(t) = \theta_k$ for $t \in [k\rho, k\rho + \rho)$, $\theta^\rho(\cdot)$ converges weakly to $\theta(\cdot)$ such that $\theta(\cdot)$ is the solution of the ordinary differential (10).

b) Suppose that (10) has a unique stationary point θ^* that is stable in the sense of Lyapunov, $\{\theta_k\}$ is tight in $\mathbb{R}^d$. Then, $\theta^\rho(t_\rho + \cdot)$ converges weakly to θ^* as $\rho \to 0$ and $t_\rho \to \infty$.

Because our adaptive algorithm uses small step size ρ_n that are not tending to 0, we will use Theorem 9 together with Theorem 6 to obtain some practically useful results. The question we wish to address is: Suppose $\rho_n \not\to 0$. How far are we away from the optimal value θ^* if we use algorithm (5)? Our approach is based on a perturbed Lyapunov function method. The main idea can be illustrated as follows. We aim to use Lyapunov stability to carry out the analysis. However, the noise sequence is correlated. We thus introduce a small perturbation to the Lyapunov function, which results in the needed cancelation of un-wanted terms. Then we obtain an estimate of $\mathbb{E}V(\theta_k^{n\ell})$ in terms of the LR ρ_n. This can be considered as a robustness result. In particular, when the Lyapunov function is locally quadratic in that $V(\theta) = (\theta - \theta^*)'S(\theta - \theta^*) + o(|\theta - \theta^*|^2)$, for a positive definite matrix S, then we can obtain a mean squares estimate $\mathbb{E}|\theta_k^{n\ell} - \theta^*|^2 = O(\rho_n)$ if $\{\rho_n\}$ is a constant. Then, we treat the random ρ_n, naturally, we need to use a conditional expectation argument. The details are to follow.

B. Error Bounds on $\theta_k^{n\ell}$

For simplicity, denote the conditional expectation with respect to the σ-algebra $\mathcal{F}_k^{n\ell}$ by $\mathbb{E}_k$.

A4) There is a Lyapunov function $V(\cdot) : \mathbb{R}^d \to \mathbb{R}$ such that

a) $V(\theta) \to \infty$ as $|\theta| \to \infty$ and $V(\theta) > 0$ for $\theta \neq \theta^*$;
b) $V(\cdot)$ is twice continuously differentiable and the Hessian $\nabla^2 V(\theta)$ is uniformly bounded;
c) $[\nabla V(\theta)]'\nabla\overline{J}(\theta) \geq \lambda V(\theta)$ for some $\lambda > 0$;
d) the following estimates hold

$$\begin{aligned}
&\sum_{j=k}^{\infty} |\mathbb{E}_\ell[\nabla V(\theta)]'[\nabla J(\theta, \xi_j^{n\ell}) - \nabla\overline{J}(\theta)]| \\
&\quad \leq K(V(\theta) + 1) \\
&\sum_{j=k}^{\infty} |\mathbb{E}_k[\nabla J(\theta, \xi_j^{n\ell}) - \nabla\overline{J}(\theta)]|^2 \\
&\quad \leq K(1 + V(\theta)) \\
&\sum_{j=k}^{\infty} |\mathbb{E}_k[\nabla J(\theta, \xi_j^{n\ell}) - \nabla\overline{J}(\theta)]_\theta|^2 \\
&\quad \leq K(1 + V(\theta))
\end{aligned} \tag{33}$$

where $[\cdot]_\theta$ denotes the partial derivatives with respect to the variable θ;

(A5) $|\nabla J(\theta, \xi)|^2 \leq K(V(\theta) + 1)$ and $\nabla^2 J(\theta, \xi)$ (the Hessian of J w.r.t. θ) is bounded.

Theorem 10: For a fixed n and a constant LR ρ_n, the following results hold.

1) $\{\theta_k^{n\ell}\}$ is bounded in probability.
2) There is a $\widetilde{\kappa} = \widetilde{\kappa}(\rho_n)$ such that for $k \geq \widetilde{\kappa}$,

$$\mathbb{E}V(\theta_k^{n\ell}) = O(\rho_n).$$

The proof of this result is moved to the Appendix. To proceed, we obtain a couple of corollaries. The first one relates the error bounds to the second moments, which is a direct consequence of Theorem 10 with a particular class of Lyapunov function chosen.

Corollary 11: Assume the conditions of Theorem 10. In addition, suppose that the Lyapunov function is locally quadratic in that

$$V(\theta) = (\theta - \theta^*)'S(\theta - \theta^*) + o(|\theta - \theta^*|^2)$$

for a positive definite matrix S. Fix a ρ_n in the nth epoch. Then, we obtain a mean squares estimate

$$\mathbb{E}|\theta_k^{n\ell} - \theta^*|^2 = O(\rho_n).$$

Remark 12: The abovementioned result is under the premise that ρ_n is a fixed constant. In the actual computation, ρ_n is an LR coming from the adaptive learning algorithm. Let us denote the σ-algebra generated by the random ρ_js up to time n by $\mathcal{G}_n = \sigma\{\rho_j : j \leq n\}$. Then, Theorem 10 can be restated as

$$\mathbb{E}_{\mathcal{G}_n} V(\theta_k^{n\ell}) = O(\rho_n).$$

Similarly, Corollary 11 can be restated as $\mathbb{E}_{\mathcal{G}_n}|\theta_k^{n\ell} - \theta^*|^2 = O(\rho_n)$.

So far, the result is for $\theta_k^{n\ell}$ with $k \leq \ell$. The bounds, in fact, are for each epoch. Next we try to obtain a bound that is across the epoch. Note that ρ_n comes from adaptive LRs, so it depends on ε. By virtue of Proposition 7, we have $\rho^\varepsilon(t_\varepsilon + \cdot) \to \rho^*$ weakly or in probability. Effectively, for n large, ε small, and $n\varepsilon \to \infty$, we have

$$\rho_n = \rho^* + o(1), \quad \text{where } o(1) \to 0 \text{ in probability.}$$

In view of Corollary 11, for some $K > 0$

$$\begin{aligned}
&\mathbb{E}_{\mathcal{G}_n}|\theta_k^{n\ell} - \theta^*|^2 \\
&\quad \leq K\rho_n \\
&\quad \leq K(\rho^* + o(1)).
\end{aligned}$$

Taking expectation mentioned above, we obtain $\mathbb{E}|\theta_k^{n\ell} - \theta^*|^2 = O(\rho^*)$ for n and k sufficiently large and ε sufficiently small. We summarize this as follows.

Corollary 13: There exist n_0, k_0, ε_0 such that for all $n \geq n_0$, $k \geq k_0$, and $\varepsilon \leq \varepsilon_0$, we have $\mathbb{E}|\theta_k^{n\ell} - \theta^*|^2 = O(\rho^*)$.

V. Numerical Examples

In this section, we present several numerical examples to illustrate how our adaptive LR algorithm works. In our examples, we use N_{sample} to denote the number of training samples (the number of in-sample Monte Carlo simulations), use N to stand for the total number of steps in the time horizon of the state and observation processes, and use n_0 to be the training window size. For $k = n_0, \ldots, N$ with fixed sample ω, we take $\{y_k(\omega), y_{k+1}(\omega), \ldots, y_{k+n_0-1}(\omega)\}$ as the input of the NN and the corresponding state $x_{k+n_0-1}(\omega)$ as the target. The NN used in our experiments being fully connected has four hidden layers. The neurons of each hidden layer are 32, 16, 8, 8 for first three examples and 128, 64, 32, 8 for the last example. The dimension of the input layer of NN equals n_0 times the dimension of observation process y. The output layer has the same dimension as that of the state process. We use the rectified linear activation function for hidden layers and the identity activation function

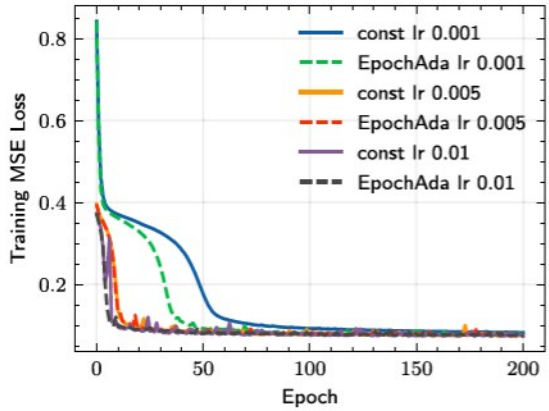

Fig. 1. Example 1: MSE loss functions.

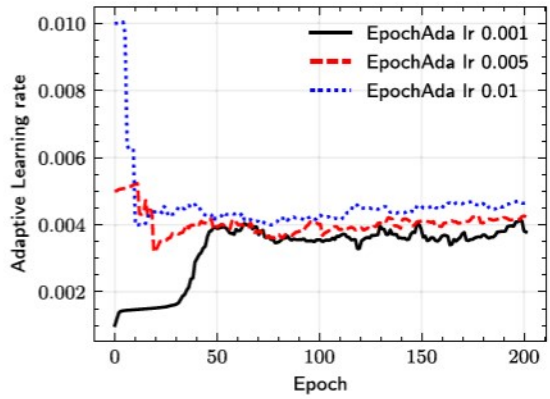

Fig. 2. Example 1: Adaptive LRs with initials 0.001, 0.005, and 0.01.

for the output layer. The loss function J in (5) used for state is taken as the mean squared error (MSE) and the loss function χ for LR is chosen as quadratic error.

In this article, for each of the following examples, we take $N_{\text{sample}} = 256$, $N = 100$, $n_0 = 10$. We use $\varepsilon = 0.001$ in (5) where the second line is substituted by the finite difference form as (39) and take $\delta = 0.01$. We also take the discretization step size $\eta = 0.04$. The noise w_n and v_n are independent Gaussian random variables of mean zero and covariance identity I with suitable dimensions (in one-dimensional case, it is simply 1). Following the training of the NN, we generate $N_{\text{sample}} = 256$ out-of-sample paths to validate our filtering results. We define the relative error of vectors $x(\omega) = (x_{n_0}(w), \ldots, x_N(w))$ and $x^0(\omega) = (x^0_{n_0}(w), \ldots, x^0_N(\omega))$ as

$$\|x - x^0\| = \frac{\sum_{k=1}^{N_{\text{sample}}} \sum_{n=n_0}^{N} \sum_{l=1}^{m} |x_{n,l}(w_k) - x^0_{n,l}(w_k)|}{\sum_{k=1}^{N_{\text{sample}}} \sum_{n=n_0}^{N} \sum_{l=1}^{m} (|x_{n,l}(w_k)| + |x^0_{n,l}(w_k)|)}$$

where $x_n(w) = (x_{n,1}(w), \ldots, x_{n,m}(w))$, $x^0_n(w) = (x^0_{n,1}(w), \ldots, x^0_{n,m}(w))$ for $n = n_0, \ldots, N$.

In what follows, we consider models with regime switching. The switching process $\alpha_n \in \{1, 2\}$ has generator $Q = \begin{bmatrix} -0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix}$. Both linear and nonlinear models of one and two dimensions are considered in first three examples. For higher dimensional systems, we consider a six dimensional nonlinear model in the last example.

A. Example 1: One-Dimensional Nonlinear Model

We consider a one-dimensional state x_n and observation y_n satisfying

$$\begin{cases} x_{n+1} = x_n + \eta \sin(5x_n\alpha_n) + \sqrt{\eta}\sigma w_n \\ y_{n+1} = y_n + 2\eta x_n + \sqrt{\eta}\sigma_1 v_n \end{cases} \tag{34}$$

where $\sigma = 0.7$, $\sigma_1 = 0.2$, and $x_0 = y_0 = 0$. The random variables w_n and v_n are independent Gaussian with mean 0 and variance 1. In Fig. 1, we plot the training loss function with both constant LR and our adaptive LR (denoted by EpochAda) with the initial data being the constant rate. For example, if we initially chose the constant LR $\rho = 0.001$, corresponding loss function is given by the solid line in Fig. 1. Using this $\rho_0 = 0.001$ as the initial value for the adaptive LR, the corresponding loss function decreases much faster after around 20 epochs. Similarly, we can start at $\rho = 0.005$ and 0.01. The corresponding loss functions also improve but the difference between constant LR and adaptive LR are less pronounced.

TABLE I
EXAMPLE 1: RELATIVE ERRORS OF THE OUT-OF-SAMPLE TESTS

ρ_0	0.001	0.005	0.01
EpochAda LR	0.155	0.143	0.143

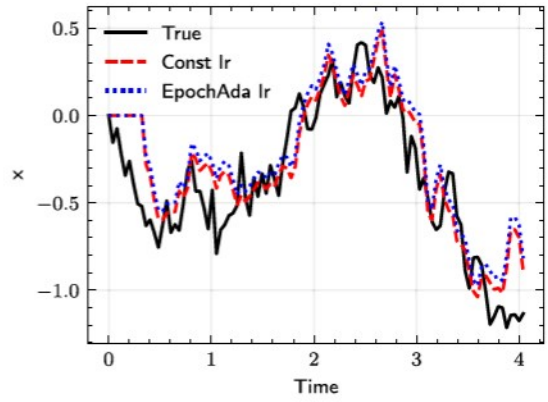

Fig. 3. Example 1: Sample paths of the state and deep filtering with constant $\rho = 0.004$ and the adaptive LR with initial $\rho_0 = 0.004$.

The paths of our adaptive ρ_n can be found in Fig. 2 with initial values being ρ_0=0.001, 0.005, and 0.01, respectively. It can be seen that the LR appears to converge around 0.004. If initial setup was higher then it tends to decrease and if initial setup was lower, it moves up gradually. The corresponding relative errors of the out-of-sample filtering are given in Table I.It is readily seen that the initial data has little influence on the relative error, which is a sign of robustness. It, in fact, is favorable from a computation point of view.

Finally, a sample path of the state x_n and the corresponding paths of DFs with constant LR and adaptive LR are provided in Fig. 3.

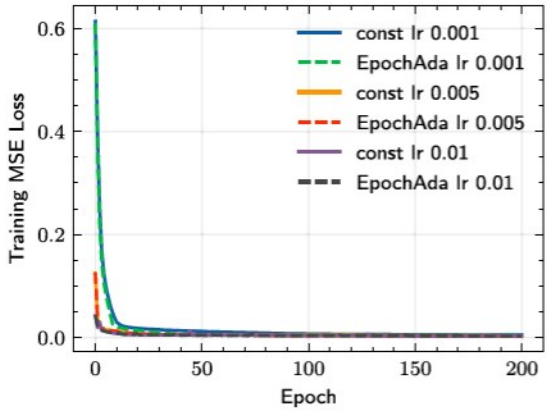

Fig. 4. Example 2: The loss functions.

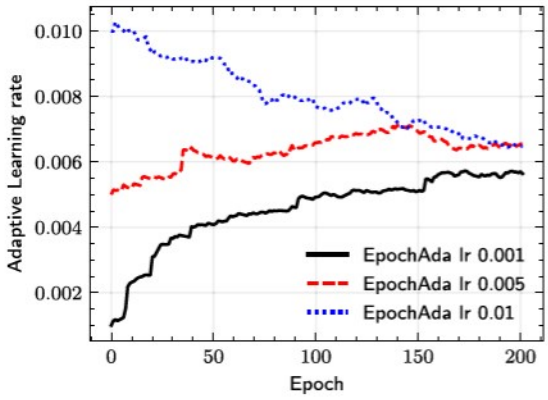

Fig. 5. Example 2: The convergence of the adaptive LRs.

B. Example 2: Two-Dimensional Linear (In (x, y)) Model

We consider a two-dimensional linear (in (x, y)) model where the state x_n and the observation y_n satisfy the equations

$$\begin{cases} x_{n+1} = x_n + \eta F(\alpha_n) x_n + \sqrt{\eta}\sigma(\alpha_n) w_n \\ y_{n+1} = y_n + \eta G(\alpha_n) x_n + \sqrt{\eta}\sigma_1 v_n \end{cases} \tag{35}$$

where the Markov chain $\alpha_n \in \{1, 2\}$, the initial condition $x_0 = (1, -1)'$, and

$$F(1) = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}, \sigma(1) = \begin{bmatrix} 1 & 0.3 \\ 0 & 1 \end{bmatrix}, G(1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$F(2) = \begin{bmatrix} -1 & 0 \\ 1 & -1 \end{bmatrix}, \sigma(2) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, G(2) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

and the observation noise matrix

$$\sigma_1 = \begin{bmatrix} 0.2 & 0.05 \\ 0 & 0.2 \end{bmatrix}.$$

Note that although the system is linear in (x, y), it is still nonlinear due to the presence of α_n. The corresponding loss functions are shown in Fig. 4. As can be seen in Fig. 5, the resulting adaptive LR appears to converge to around 0.006. If we start with this equilibrium with constant LR and adaptive LR, two corresponding out-of-sample paths for the state and DF can be seen in Fig. 6. Both learning rates lead to similar performance.

TABLE II
EXAMPLE 2: RELATIVE ERRORS OF THE STATE x_n AND THE DEEP FILTERING RESULTS

ρ_0	0.001	0.005	0.01
EpochAda LR	0.061	0.058	0.057

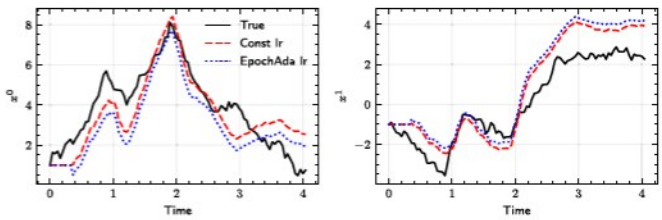

Fig. 6. Example 2: Sample paths of out-of-sample state x_n and the deep filtering outcomes with constant LR $\rho = 0.006$ and adaptive LR with initial $\rho_0 = 0.006$.

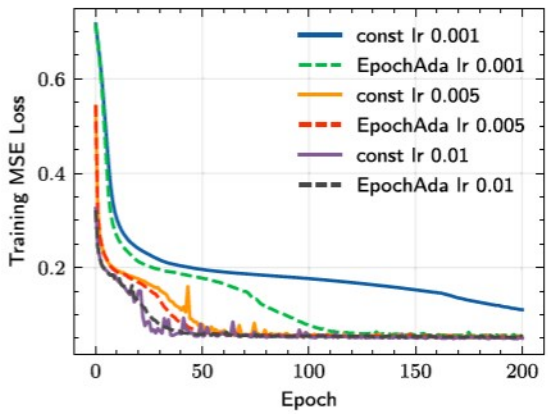

Fig. 7. Example 3: The loss functions.

The relative errors of the state x_n and the deep filtering with adaptive LRs with ρ_0 =0.001, 0.005, and 0.01, respectively, are given in Table II. The behavior is similar to that reported in Table I.

C. Example 3: Two-Dimensional Nonlinear Model

Let us consider a two-dimensional nonlinear model. The state vector x_n and the observation vector y_n satisfy the following equations:

$$\begin{cases} x_{n+1} = x_n + \eta \begin{bmatrix} \sin((0.3x_n^0 + 0.5x_n^1)\alpha_n) \\ \sin(0.3x_n^1\alpha_n) \end{bmatrix} + \sqrt{\eta}\sigma w_n \\ y_{n+1} = y_n + \eta G x_n + \sqrt{\eta}\sigma_1 v_n \end{cases} \tag{36}$$

where $x_n = (x_n^0, x_n^1)'$ and the Markov chain $\alpha_n \in \{1, 2\}$ and

$$\sigma = \begin{bmatrix} 1 & -0.3 \\ 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \sigma_1 = \begin{bmatrix} 0.2 & 0.05 \\ 0 & 0.2 \end{bmatrix}.$$

The initial state is chosen as $x_0 = (1, -1)'$. Like in the first two examples, the training loss functions with constant LRs (0.001, 0.005, 0.01) as well as adaptive LRs with the initial

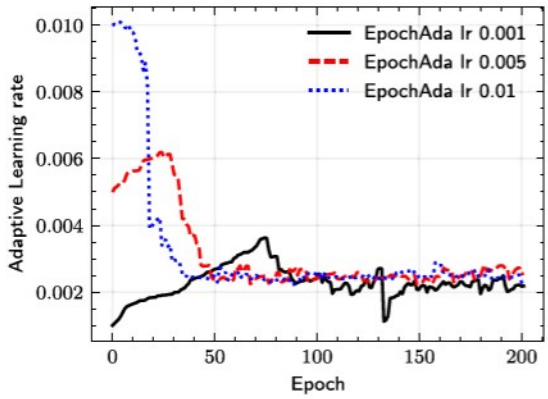

Fig. 8. Example 3: Paths of the adaptive LRs with initials 0.001, 0.005, and 0.01.

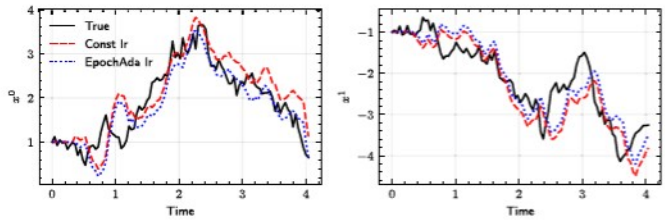

Fig. 9. Example 3: Sample paths of out-of-sample state and the sample paths of the DFs with constant LR $\rho = 0.0025$ and adaptive LR with initial $\rho_0 = 0.0025$.

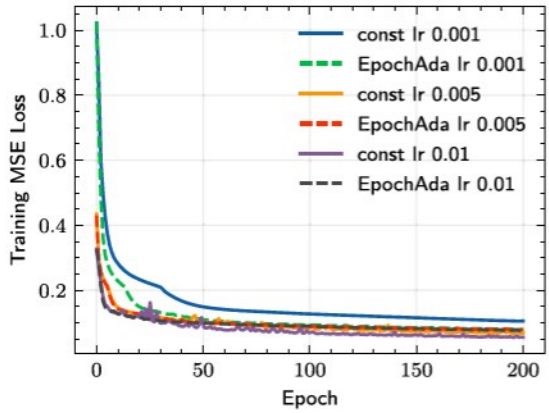

Fig. 10. Example 4: The loss functions.

TABLE III
EXAMPLE 3: RELATIVE ERRORS OF x_n AND THE DEEP FILTERING RESULTS

ρ_0	0.001	0.005	0.01
EpochAda LR	0.112	0.110	0.113

rates are given in Fig. 7 and demonstrate similar behaviors as in the previous models.

The adaptive LRs also shown similar convergence in Fig. 8. It appears to converges to 0.0025 in this example.

The relative errors of the state x_n and the deep filtering outcomes are given in Table III. It again confirm that the results are not much influenced by the initial conditions.

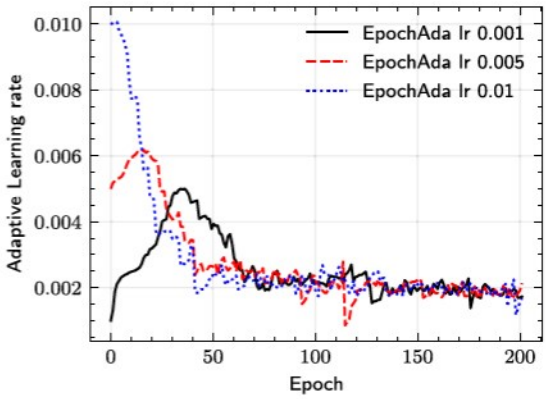

Fig. 11. Example 4: Paths of the adaptive LRs with initials 0.001, 0.005, and 0.01.

TABLE IV
EXAMPLE 4: RELATIVE ERRORS OF x_n AND THE DEEP FILTERING RESULTS

ρ_0	0.001	0.005	0.01
EpochAda LR	0.225	0.228	0.224

Finally, we provide sample paths of the state x_n and the corresponding out-of-sample deep filtering sample paths with constant LR 0.0025 and adaptive LR with $\rho_0 = 0.0025$ in Fig. 9. Both LRs lead to similar performance.

D. Example 4: Six-Dimensional Nonlinear Tracking Model

Finally, we consider the following six-dimensional nonlinear model where the state x_n and the observation y_n satisfying

$$\begin{cases} x_{n+1} = x_n + \eta F x_n + \sqrt{\eta} w_n \\ y_{n+1} = y_n + \eta h(x_n) + \sqrt{\eta}\sigma_1 v_n \end{cases} \tag{37}$$

where $w_n \backsim N(0, I_6)$, $v_n \backsim N(0, I_6)$ with I_6 being a 6×6 dimensional identity matrix, the matrix F is

$$F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -\omega_1^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\omega_2^2 & 0 \end{bmatrix}$$

$\sigma_1 = 0.2$, and the function $h(x)$ is defined as

$$h(x) = [\sqrt{x_0^2 + x_3^2}, \tan^{-1}(x_3/x_0), x_1, x_2, x_4, x_5]'$$

where the state $x = (x_0, x_1, x_2, x_3, x_4, x_5)'$. For experiments, we take $\omega_1 = 1, \omega_2 = 0.5$ and $x_0 = (1, 1, 1, 1, 1, 1)'$. The training losses are exhibited in Fig. 10 and the evolution of adaptive LR is presented in Fig. 11.

We also provide the relative errors of the state x_n and the deep filtering with adaptive learning outcomes in Table IV. These relative errors show the robustness of our DF with respect to the initial selections of the adaptive LRs.

In addition, as depicted in Fig. 11, the adaptive LRs converge after 50 epochs. Finally, we include the sample paths of the state

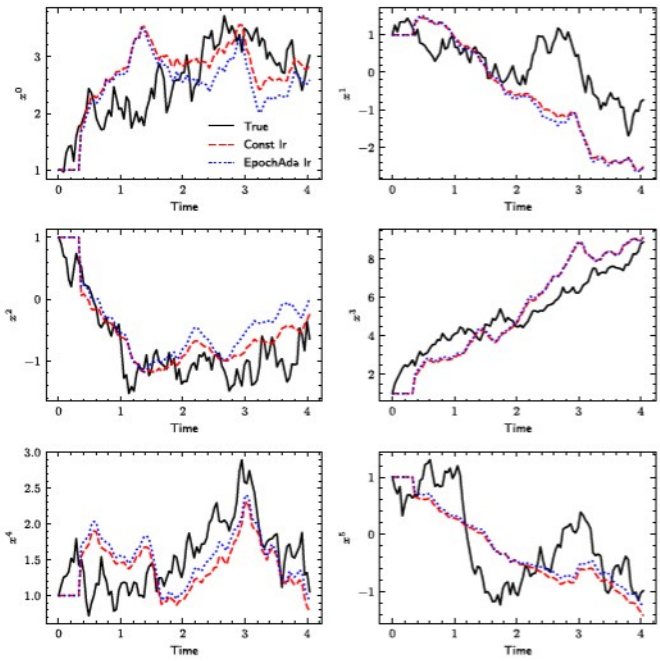

Fig. 12. Example 4: Sample paths of out-of-sample state and the sample paths of the DFs with constant LR $\rho = 0.002$ and adaptive LR with initial $\rho_0 = 0.002$.

x_n and the corresponding out-of-sample deep filtering sample paths with constant LR 0.002 and adaptive LR with initial $\rho_0 = 0.002$ in Fig. 12 . These results demonstrate the effectiveness and capability of the DF dealing with high dimensional and highly nonlinear models.

VI. Further Remarks

In this section, we make several remarks regarding the deep filtering and our approximation algorithms. Also included are several possible extensions.

1) In our algorithm, for example, in (5), we used bounded nonadditive noise. The gradient estimates are of the form $\nabla_\theta J(\theta^{n\ell}, \xi_k^{n\ell})$. This model can be easily generalized to
$$\nabla_\theta J(\theta_k^{n\ell}, \xi_k^{n\ell}) + \psi_k^{n\ell}$$
where $\{\psi_k^{n\ell}\}$ is a sequence of unbounded noise independent of $\xi_k^{n\ell}$ such that $\{\psi_k^{n\ell}\}$ is a stationary mixing process with mean zero and bounded second moments. All previous results still hold. Only notation becomes more complex.
2) If the noisy gradients as in (5) are not available, but we can only observe the function values with noise instead. Then, we can construct a more general version of the algorithm. We use
$$\theta_{k+1}^{n\ell} = \theta_k^{n\ell} - \rho_n \widetilde{J}^{n\ell}(\theta_k^{n\ell}, \xi_k^{n\ell}),\ k = 0, \ldots, \ell - 1 \tag{38}$$
where $\widetilde{J}^{n\ell}$ is the sample finite difference (approximation to the gradient of $\overline{J}^{n\ell}$). That is, for $j = 1, \ldots, d$
$$\widetilde{J}^{n\ell,j}(\theta_k^{n\ell}, \xi_k^{n\ell}) = \frac{J^{n\ell}(\theta_k^{n\ell} + c_n e_j, \xi_k^{n\ell,+}) - J^{n\ell}(\theta_k^{n\ell} - c_n e_j, \xi_k^{i\ell,-})}{2c_n} \tag{39}$$
where e_j is the standard unit vector, i.e., $e_j = (0, \ldots, 1, \ldots)$, $c_n = c_n(\rho(n)) \to 0$ and $\rho_n/c_n \to 0$ as $\rho_n \to 0$, $\delta = \delta(\varepsilon) \to 0$, and $\varepsilon/\delta \to 0$ as $\varepsilon \to 0$. So the $c_n = c_n(\rho_n)$ and $\delta = \delta(\varepsilon)$ are small finite difference parameters. Then, $\widetilde{J}^{n\ell}(\theta_k^{n\ell}, \xi_k^{n\ell}) = (\widetilde{J}^{n\ell,j}(\theta_k^{n\ell}, \xi_k^{n\ell}) :\ j \le d)$. It is readily seen that (39) are sample central finite difference approximation of the gradients.
3) The deep filtering scheme can be more general than it has been presented in this article. It can cover even those models that cannot be covered in terms of typical dynamic systems, for example, highly nonlinear oscillating function with nonadditive noise; see [24] for related numerical results.
4) A hybrid model is treated in this article in which $\alpha(t)$ is not directly observable. It would be interesting to generalize the results of this article to estimate the pair $(X(t), \alpha(t))$ jointly. In this case, additional observations of $\alpha(t)$ are needed. Then, the Wonham filter [25] can be applied.
5) In this article, we proposed an adaptive LR algorithm devoted to the filtering applications. The idea presented in this article may suggest a viable alternative for a systematic way of choosing the LRs via the recursion, which could be beneficial for many machine learning problems. Finally, we would like to emphasize that the Monte Carlo samples for DNN training can be fully replaced by real-world data. This will allow us to bypass the traditional model calibration. In this connection, it would be interesting to examine how our filtering schemes work with real-world applications.
This article work aims at setting up a framework with solid theoretical ground. The real-world applications can be and definitely should be a subject of future research.

VI. Appendix

In this Appendix, we provide a sketch of the proof of Proposition 1 showing discrete-time system (x_n, y_n, α_n) approximates the filtering equations (switching diffusion) $(X(t), Y(t), \alpha(t))$. Let $F(\cdot,\cdot,\cdot) : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \times \mathcal{M} \to \mathbb{R}$ satisfying that for each $\alpha \in \mathcal{M}$, $F(\cdot,\cdot,\alpha)$ has continuous partial derivatives up to the second order w.r.t. x and y. Define
$$\begin{aligned}\mathcal{L}F(x,y,i) &= [\nabla_x F(x,y,i)]' f(x,i) \\ &\quad + \frac{1}{2}\mathrm{tr}[\nabla_x^2 F(x,y,i)\sigma(x,i)\sigma'(x,i)] \\ &\quad [\nabla_y F(x,y,i)]' f(x,i) \\ &\quad + \frac{1}{2}\mathrm{tr}[\nabla_y^2 F(x,y,i)\sigma(x,i)\sigma'(x,i)] \\ &\quad + QF(x,y,\cdot)(i),\ i \in \mathcal{M}\end{aligned} \tag{40}$$
where $\nabla_x F$, $\nabla_y F$, $\nabla_x^2 F$, and $\nabla_y^2 F$ are the gradients and Hessians of F w.r.t. x and y, respectively, $\mathrm{tr}A$ denotes the trace of

the matrix A, and

$$QF(x,y,\cdot)(i)=\sum_{j\in\mathcal{M}}q_{ij}F(x,y,j)\text{ for each }i.$$

We define

$$W^{\eta}(t)=\sqrt{\eta}\sum_{k=0}^{\lfloor t/\eta\rfloor-1}w_k,\ \widetilde{W}^{\eta}(t)=\sqrt{\eta}\sum_{k=0}^{\lfloor t/\eta\rfloor-1}v_k$$

where $\lfloor t/\eta\rfloor$ denotes the integer part of t/η. It can be shown that $W^{\eta}(\cdot)$ converges weakly to a standard Brownian motion $W(\cdot)$, and $\widetilde{W}^{\eta}(\cdot)$ converges weakly to a standard Brownian motion $\widetilde{W}(\cdot)$, respectively. Then, we can proceed to show that $(x^{\eta}(\cdot),y^{\eta}(\cdot),\alpha^{\eta}(\cdot))$ converges weakly to $(X(\cdot),Y(\cdot),\alpha(\cdot))$, which is a martingale problem with operator (40). In this process, we need to use a truncation device similar to the proof in Theorem 6. We refer the reader to [19] for approximation of randomly modulated sequences to switching diffusions, and details on switching diffusions and convergence to switching diffusions to Yin and Zhu [27, Sec. 5.3]. *Proof of Theorem* 10. The proof of the first assertion is simpler than that of the second. In fact, the proof can be done in the second part. We, thus, will only prove the second assertion in what follows.

We will use K to denote a generic positive constant, whose value may be different for different appearances. We have

$$\begin{aligned}&\mathbb{E}_kV(\theta_{k+1}^{n\ell})-V(\theta_k^{n\ell})\\&=-\rho_n[\nabla V(\theta_k^{n\ell})]'[\nabla J(\theta_k^{n\ell},\xi_j^{n\ell})]\\&=-\rho_n[\nabla V(\theta_k^{n\ell})]'\nabla\overline{J}(\theta_k^{n\ell})\\&\quad-\rho_n[\nabla V(\theta_k^{n\ell})]'[\nabla J(\theta_k^{n\ell},\xi_j^{n\ell})-\nabla\overline{J}(\theta_k^{n\ell})]\\&\quad+O(\rho_n^2)(V(\theta_k^{n\ell})+1).\end{aligned}\tag{41}$$

To proceed, we use the methods of perturbed Lyapunov function. Define

$$V_1(x,k)=-\rho_n\sum_{j=k}^{\infty}[\nabla V(x)]'[\mathbb{E}_k\nabla J(x,\xi_j^{n\ell})-\nabla\overline{J}(x)]\tag{42}$$

and

$$\widetilde{V}(x,k)=V(x)+V_1(x,k).\tag{43}$$

It is readily checked that by using (A4)

$$\begin{aligned}|V_1(x,k)|&=|\rho_n\sum_{j=k}^{\infty}[\nabla V(x)]'[\mathbb{E}_k\nabla J(x,\xi_j^{n\ell})-\nabla\overline{J}(x)]|\\&=K\rho_n(V(x)+1).\end{aligned}\tag{44}$$

Thus, the perturbation is small in magnitude.

Next, we demonstrate it results in the desired cancelation. In fact, we have

$$\begin{aligned}&\mathbb{E}_k\widetilde{V}(\theta_{k+1}^{n\ell},k+1)-\widetilde{V}(\theta_k^{n\ell},k)\\&=\mathbb{E}_kV(\theta_{k+1}^{n\ell})-V(\theta_k^{n\ell})\\&\quad+\mathbb{E}_kV_1(\theta_k^{n\ell},k+1)-V_1(\theta_k^{n\ell},k)\\&\quad+\mathbb{E}_k[V_1(\theta_{k+1}^{n\ell},k+1)-V_1(\theta_k^{n\ell},k+1)].\end{aligned}\tag{45}$$

In view of the definition of (42), we have

$$\begin{aligned}&\mathbb{E}_kV_1(\theta_k^{n\ell},k+1)-V_1(\theta_k^{n\ell},k)\\&\quad=\rho_n[\nabla V(\theta_k^{n\ell})]'[\nabla J(\theta_k^{n\ell},\xi_j^{n\ell})-\nabla\overline{J}(\theta_k^{n\ell})]\end{aligned}\tag{46}$$

$$\begin{aligned}&\mathbb{E}_k[V_1(\theta_{k+1}^{n\ell},k+1)-V_1(\theta_k^{n\ell},k+1)]\\&\quad=-\rho_n\mathbb{E}_k\sum_{j=k+1}^{\infty}[\nabla V(\theta_{k+1}^{n\ell})]'\\&\qquad\times[\mathbb{E}_{k+1}\nabla J(\theta_{k+1}^{n\ell},\xi_j^{n\ell})-\nabla\overline{J}(\theta_{k+1}^{n\ell})]\\&\qquad+\rho_n\sum_{j=k+1}^{\infty}[\nabla V(\theta_k^{n\ell})]'[\mathbb{E}_k\nabla J(\theta_k^{n\ell},\xi_j^{n\ell})-\nabla\overline{J}(\theta_k^{n\ell})].\end{aligned}\tag{47}$$

Using (A4) and (A5), we can show that

$$\begin{aligned}&|\mathbb{E}_k[V_1(\theta_{k+1}^{n\ell},k+1)-V_1(\theta_k^{n\ell},k+1)]|\\&\quad\le K\rho_n^2(V(\theta_k^{n\ell})+1).\end{aligned}\tag{48}$$

Using (A4) and combining (41), (46)–(48), we obtain

$$\begin{aligned}&\mathbb{E}_k\widetilde{V}(\theta_{k+1}^{n\ell},k+1)-\widetilde{V}(\theta_k^{n\ell},k)\\&\quad\le-\lambda\rho_nV(\theta_k^{n\ell})+K\rho_n^2(V(\theta_k^{n\ell})+1)\\&\quad\le-\lambda\rho_n\widetilde{V}(\theta_k^{n\ell},k)+K\rho_n^2(\widetilde{V}(\theta_k^{n\ell},k)+1).\end{aligned}\tag{49}$$

The last line above follows from the estimate (44). Because ρ_n is small, we may consider ρ_n to be small enough that $K\rho_n^2\le\lambda\rho_n/2$, and

$$\begin{aligned}&\mathbb{E}_k\widetilde{V}(\theta_{k+1}^{n\ell},k+1)\\&\le(1-(\rho_n^2/2))\widetilde{V}(\theta_k^{n\ell},k)+O(\rho_n^2)\\&\le(1-(\rho_n^2/2))^{k+1}\widetilde{V}(\theta_0^{n\ell},0)+\sum_{j=0}^{k}(1-(\rho_n^2/2))^jO(\rho_n^2).\end{aligned}\tag{50}$$

We can choose $\widetilde{\kappa}=\widetilde{\kappa}(\rho_n)$ so that for $k\ge\widetilde{\kappa}$, $(1-(\rho_n^2/2))\le O(\rho_n)$. Taking expectation in (50), we arrive at

$$\mathbb{E}\widetilde{V}(\theta_k^{n\ell},k)=O(\rho_n).\tag{51}$$

Finally, using (44), we also obtain $\mathbb{E}V(\theta_k^{n\ell})=O(\rho_n)$ as desired. ■

REFERENCES

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1979.

[2] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Trans. Autom. Control*, vol. 33, no. 8, pp. 780–783, Aug. 1988.

[3] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *SIAM Rev.*, vol. 60, pp. 223–311, 2018.

[4] O. L. V. Costa, "Linear minimum mean square error estimation for discrete-time Markovian jump linear systems," *IEEE Trans. Autom. Control*, vol. 39, no. 8, pp. 1685–1689, Aug. 1994.

[5] M. H. A. Davis, "Nonlinear filtering and stochastic flows," in *Proc. Int. Congr. Mathematicians*, Berkeley, CA, USA, 1986, pp. 1000–1010.

[6] F. Dufour and P. Bertrand, "An image-based filter for discrete-time Markovian jump linear systems," *Automatica*, vol. 32, pp. 241–247, 1996.

[7] T. E. Duncan, "Probability densities for diffusion processes with applications to nonlinear filtering theory and detection theory," Ph.D. Dissertation, Dept. ECE, Stanford Univ., Stanford, CA, USA, 1967.
[8] S. N. Ethier and T. G. Kurtz, *Markov Processes: Characterization and Convergence*. New York, NY, USA: Wiley, 1986.
[9] R. Frey, T. Schmidt, and L. Xu, "On Galerkin approximations for the Zakai equation with diffusive and point process observations," 2018. [Online]. Available: https://arxiv.org/pdf/1303.0975.pdf
[10] J. Gao and H. Tembine, "Distributed mean-field type filters for big data assimilation," in *Proc. 18th IEEE Int. Conf. High Perform. Comput. Commun.*, 2016, pp. 1446–1453.
[11] T. G. Kurtz and D. L. Ocone, "Unique characterization of conditional distributions in nonlinear filtering," *Ann. Probability*, vol. 16, pp. 80–107, 1988.
[12] H. J. Kushner, "On the differential equations satisfied by conditional probability densities of Markov processes, with applications," *J. SIAM Control A*, vol. 2, pp. 106–119, 1964.
[13] H. J. Kushner, *Approximation and Weak Convergence Methods for Random Processes, With Applications to Stochastic Systems Theory*. Cambridge, MA, USA: MIT Press, 1984.
[14] H. J. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. New York, NY, USA: Springer, 2003.
[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
[16] S. Lototsky, R. Mikulevicius, and B. L. Rozovskii, "Nonlinear filtering revisited: A spectral approach," *SIAM J. Control Optim.*, vol. 35, pp. 435–461, 1997.
[17] R. E. Mortensen, "Maximum-likelihood recursive nonlinear filtering," *J. Optim. Theory Appl.*, vol. 2, pp. 386–394, 1968.
[18] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, Norwell, MA, USA: Kluwer, 2004.
[19] S. L. Nguyen and G. Yin, "Weak convergence of Markov modulated random sequences," *Stochastics*, vol. 82, pp. 521–552, 2010.
[20] M. Nielsen. *Neural Networks and Deep Learning*, vol. 25. San Francisco, CA, USA: Determination Press, 2015.
[21] J. Park, D. Yi, and S. Ji, "A novel learning rate schedule in optimization for neural networks and it's convergence," *Symmetry*, vol. 12, 2020, Art. no. 660.
[22] S. Ruder, "An overview of gradient decent optimization algorithms," 2017, *arXiv:1609.04747*.
[23] H. Wang, T. Zariphopoulou, and X. Y. Zhou, "Exploration versus exploitation in reinforcement learning: A stochastic control approach," *J. Mach. Learn. Res.*, vol. 21, 2020, pp. 1–34.
[24] L. Y. Wang, G. Yin, and Q. Zhang, "Deep filtering," *Commun. Inf. Syst.*, vol. 21, pp. 651–667, 2021.
[25] W. M. Wonham, "Some applications of stochastic differential equations to optimal nonlinear filtering," *SIAM J. Control*, vol. 2, pp. 347–369, 1965.
[26] G. Yin and Q. Zhang, *Continuous-Time Markov Chains and Applications: A Two-Time Scale Approach*, 2nd ed. New York, NY, USA: Springer, 2013.
[27] G. Yin and C. Zhu, *Hybrid Switching Diffusions: Properties and Applications*, New York, NY, USA: Springer, 2010.
[28] M. Zakai, "On the optimal filtering of diffusion processes," *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, vol. 11, pp. 230–243, 1969.

Hongjiang Qian received the B.S degree in mathematics from the Huazhong University of Science and Technology, Wuhan, China, in 2018. He is currently working toward the Ph.D. degree in the Department of Mathematics with the University of Connecticut, Storrs, CT, USA.

He spent two years studying with the Department of Mathematics, Wayne State University, Detroit, MI, USA. His current research interests include stochastic approximation and stochastic systems theory and applications.

George Yin (Life Fellow, IEEE) received the B.S. degree in mathematics from the University of Delaware, Newark, DE, USA, in 1983 and the M.S. degree in electrical engineering and the Ph.D. degree in applied mathematics from Brown University, Providence, RI, USA, in 1987.

He joined the Department of Mathematics, Wayne State University, Detroit, MI, USA, in 1987, and became Professor in 1996 and the University Distinguished Professor in 2017. He moved to the University of Connecticut, Storrs, CT, USA, in 2020. His research interests include stochastic processes, stochastic systems theory and applications.

Dr. Yin was the Chair of the SIAM Activity Group on Control and Systems Theory, and served on the Board of Directors of the American Automatic Control Council. He is the Editor-in-Chief of *SIAM Journal on Control and Optimization*, was a Senior Editor of IEEE CONTROL SYSTEMS LETTERS, and is an Associate Editor of *ESAIM: Control, Optimization and Calculus of Variations*, *Applied Mathematics and Optimization*, and many other journals. He was an Associate Editor for *Automatica* 2005–2011 and IEEE TRANSACTIONS ON AUTOMATIC CONTROL 1994–1998. He is a Fellow of IFAC and a Fellow of SIAM.

Qing Zhang (Senior Member, IEEE) received the Ph.D. degree in applied mathematics from Brown University, Providence, RI, USA.

He is currently a Professor of Mathematics with the University of Georgia, Athens, GA, USA. He has authored or coauthored five monographs on production planning and two-time scale Markovian systems and applications and more than 200 research papers. He coedited six books. He specializes in stochastic systems and control, filtering, and applications in finance.

Dr. Zhang was an Associate Editor for *Automatica*, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, and *SIAM Journal on Control and Optimization*. He is currently a Corresponding Editor for *SIAM Journal on Control and Optimization*. He also served on a number of international conference organizing committees including Co-Chair of the organizing committee for the SIAM Conference on Control and Applications in 2017.