Journal of Machine Learning Research 23 (2022) 1-55 Submitted 8/21; Revised 1/22; Published 4/22

Policy Evaluation and Temporal–Difference Learning in Continuous Time and Space: A Martingale Approach

Yanwei Jia YJ2650@COLUMBIA.EDU
Department of Industrial Engineering and Operations Research
Columbia University
New York, NY 10027, USA

Xun Yu Zhou XZ2574@COLUMBIA.EDU
Department of Industrial Engineering and Operations Research &
The Data Science Institute
Columbia University
New York, NY 10027, USA

Editor: Csaba Szepesvari

Abstract

We propose a unified framework to study policy evaluation (PE) and the associated temporal difference (TD) methods for reinforcement learning in continuous time and space. We show that PE is equivalent to maintaining the martingale condition of a process. From this perspective, we find that the mean–square TD error approximates the quadratic variation of the martingale and thus is not a suitable objective for PE. We present two methods to use the martingale characterization for designing PE algorithms. The first one minimizes a "martingale loss function", whose solution is proved to be the best approximation of the true value function in the mean–square sense. This method interprets the classical gradient Monte-Carlo algorithm. The second method is based on a system of equations called the "martingale orthogonality conditions" with test functions. Solving these equations in different ways recovers various classical TD algorithms, such as TD(λ), LSTD, and GTD. Different choices of test functions determine in what sense the resulting solutions approximate the true value function. Moreover, we prove that any convergent time-discretized algorithm converges to its continuous-time counterpart as the mesh size goes to zero, and we provide the convergence rate. We demonstrate the theoretical results and corresponding algorithms with numerical experiments and applications.

Keywords: continuous time and space, reinforcement learning, policy evaluation, temporal difference, martingale

1. Introduction

Policy evaluation (PE) is a crucial step in most critic-related reinforcement learning (RL) algorithms such as actor-critic algorithms and policy iteration. Its objective is to estimate/predict the value function of a given policy using samples, generally without knowledge about the environment. Existing PE methods have predominantly been limited to discrete-time problems with finite-state Markov decision processes (MDPs). For instance, Monte Carlo methods use samples to estimate expectations assuming the whole sample trajectories can be repeatedly presented for training; hence they are compatible with offline learning. The most popular PE methods are based on the temporal difference (TD) error. These are incremental learning procedures driven by the error between temporally successive predictions. Sutton (1988) argues that predictions of the TD methods are both more accurate and easier to compute than other methods. More importantly, these methods can learn the value in real-time before a task terminates; hence it can be used both online and offline (Sutton and Barto, 2018).

Despite the fast development and vast applications, there are two major limitations in the current study on RL in general and on PE in particular. First, most algorithms are developed for MDPs, and little attention has been paid to problems with continuous time and space. The few existing studies in the continuous setting have been largely restricted to deterministic systems; see for example Baird (1993); Doya (2000); Frémaux et al. (2013); Vamvoudakis and Lewis (2010) and Lee and Sutton (2021), where the state processes follow ordinary differential equations (ODEs) and there are no environmental noises. In particular, Baird (1993) and Doya (2000) are the first to propose some continuous-time versions of the TD methods. In real life, however, there are abundant examples in which an agent can or indeed needs to interact with a *random* environment at ultra-high frequency, e.g., high-frequency stock trading, autonomous driving, and robots navigation. Second, while there have been numerous PE algorithms proposed using function approximation such as residual gradient, gradient Monte Carlo, and TD(λ), they were usually devised in heuristic and ad hoc manners and their underlying objectives were not always clearly stated.[1] Although many of them are proved to be convergent, the limiting functions are not always interpreted properly especially if the function approximators do not contain the true solutions. In short, there seems a lack of a *unified* framework to study PE and there is need for a continuous time and space perspective, from which many well-known algorithms appear as discretizations.

The goal of this paper is to bridge these gaps by providing a unified theoretical underpinning of PE in continuous time and space with general Markov diffusion processes. Instead of discretizing time, state, and action from the start and then applying the existing discrete techniques and results, we carry out all our theoretical analysis for the continuous setting and discretize time only at the final, algorithmic stage. The advantage of doing so is two-fold. On one hand, as Doya (2000) argues, the control performance with this approach will be smoother and the right granularity for discretization will be guided by the function approximation. On the other hand, and indeed more importantly, for analyses in a continuous setting, we have plenty of well-developed tools at our disposal including those of stochastic calculus, differential equations, and stochastic control, which, in turn, will provide better interpretability/explainability to the underlying learning technologies.

Stochastic optimization in continuous time and space, also known as stochastic control, has a long history starting from the 1960s. However, its theory is model-based, namely, the system dynamics and the objective functions are assumed to be given and known. The problem can then be solved by well-established approaches such as Pontryagin's maximum principle and Bellman's dynamic programming. For full accounts of the stochastic control theory see, e.g., Yong and Zhou (1999) and Fleming and Soner (2006). On the other hand, to our best knowledge, the study on model-free RL for diffusion processes started only recently. Wang et al. (2020) propose an entropy-regularized, stochastic relaxed control

1. See Appendix A, Table 1, for a list of names of existing PE algorithms for MDPs.

formulation for trading off exploration and exploitation in continuous time and space, and derive the continuous version of the Boltzmann distribution (Gibbs measure) as the optimal exploratory policy. When the problem is linear–quadratic (LQ), namely the dynamic is linear and the payoff is quadratic in state and action, the optimal strategy specializes to Gaussian exploration. Wang and Zhou (2020) apply this general theory to a mean–variance financial portfolio selection problem, which is inherently of an LQ structure, and design an algorithm for extensive simulation and empirical experiments. Dai et al. (2020) further consider the equilibrium mean–variance strategies addressing the time-inconsistent issue of the problem. Guo et al. (2022) extend the formulation and results of Wang et al. (2020) to mean-field games. Gao et al. (2020) use the idea of Wang et al. (2020) to a non-learning problem – simulated annealing for nonconvex optimization formulated as controlling the temperature of a Langevin diffusion.

For PE, there are generally two aspects one should address. First and more fundamentally, one specifies a mathematical objective against which a learning task is evaluated. Usually, such an objective is described by either an optimization problem (to minimize a loss/error function) or a system of equations. Second and on the implementation front, one designs an algorithm to achieve the objective. Many papers have contributed to the second aspect, namely, to develop more efficient numerical solvers to accelerate convergence, reduce variance, or save computational cost; see, e.g., Xu et al. (2002); Liu et al. (2016); Du et al. (2017). In contrast to that line of research, the present paper focuses on the first aspect aiming at building a unified theoretical framework for PE. We propose and analyze several common objectives in the continuous setting, and demonstrate that they generate continuous counterparts of some of the best-known PE algorithms for MDPs. This not only leads to PE algorithms for the continuous problems but also provides additional foundations for the discrete ones. As our algorithms designed for continuous setting are discretized in time for implementation, their convergence with a fixed discretization mesh size has been already established by existing results. Moreover, we show that, as the discretization gets finer, the limiting point of a convergent discrete-time algorithm also converges to the corresponding solution to the continuous problem, and we further provide the convergence rate.

The entire theoretical analysis of the current paper is premised upon the fact that the value function along the state process combined with the accumulated running payoff is a martingale. This martingality naturally gives rise to a target for offline learning: the value of the martingale at any given time is the least square estimate of that at the terminal time. On the other hand, the martingality leads to orthogonality conditions that in turn generate algorithms corresponding to many existing well-known TD algorithms for MDPs.

A similar martingale condition can also be derived for discrete-time MDPs, which is equivalent to the so-called Bellman equation or the Bellman consistency. In Appendix C we provide such a derivation. However, to our best knowledge, in the existing RL literature such a condition has not been explicitly presented – even if it is rather straightforward to deduce – nor has it been employed to study PE. Instead, the Bellman equation has been the predominant tool to devise PE algorithms. We demonstrate that the change of perspective from the Bellman equation to the martingality is crucial in our analysis.

Specifically, our main contributions can be summarized as follows:

(i) We show that the continuous analogue of the naïve residual gradient method, which minimizes the mean-square TD error (Barnard, 1993; Baird, 1995; Doya, 2000; Wang and Zhou, 2020), converges to the minimizer of the *quadratic variation* of the aforementioned martingale. It is, therefore, *inconsistent* with the learning objective. This in turn provides a theoretical explanation why the method is not a desired approach for PE when the environment is stochastic.

(ii) We propose a martingale loss function based on the total mean-square error between the said martingale process and its terminal value. We prove that minimizing such a loss function is equivalent to minimizing the mean-square error between the approximate value function and the true one. This loss function is implementable on samples, and justifies the Monte Carlo PE with function approximation (Sutton and Barto, 2018) in the classical MDP and RL literature.

(iii) We provide a unified perspective to interpret TD errors and the related algorithms, including TD(λ), least square TD (LSTD), and gradient TD (GTD and its variants), based on the martingale orthogonality conditions. Specifically, by introducing a finite number of suitable test functions to these conditions, the learning problem is transformed into a system of equations called moment conditions. From this vantage point, we realize that TD(λ) is nothing but to directly apply stochastic approximation to solve such equations, LSTD is to solve them explicitly when they form a linear system, and GTD methods are to solve various quadratic forms of the moment conditions. In addition, different choices of the test functions determine in what sense the true value function is approximated. For example, TD(λ) essentially correspond to different test functions for different values of λ, and hence may converge to *different* limits.

For reader's easy reference, we present Table 1 in Appendix A summarizing popular PE methods and algorithms, and the interpretations we will have discovered in this paper in terms of the objectives and the convergent limits of the algorithms.

As the conditional expectation in the expression of the value function is connected to both a partial differential equation (PDE) through the Feynman–Kac formula (Karatzas and Shreve, 2014) and to a backward stochastic differential equation (BSDE) through a martingale representation theorem (El Karoui et al., 1997), the results of the current paper have natural implications on applying machine learning methods to numerically solve (high-dimensional) PDEs in search of breaking the "curse of dimensionality". The latter has been a hotly pursued topic lately; see for example Raissi (2018) and Huré et al. (2019). Han et al. (2018) propose a deep learning approach to solving PDEs by solving the associated BSDEs via simulation. All these works need to assume that the coefficients of the PDEs are known. The results of our paper shed light on solving PDEs by PE methodologies in a data-driven way, in view of the intimate connection among PDEs, BSDEs and PE for Markov diffusion process.[2]

The rest of the paper proceeds as follows. In Section 2, we formulate the PE problem in continuous time and space and present the martingale characterization of the value function. In Section 3, we extend the classical mean-square TD error to the continuous setting and show why it is not a proper objective when the environment is stochastic through

2. It is interesting to note that there seems to be less research on solving recursive Bellman-like equations using MDPs, even though the same curse of dimensionality exists for discrete-time equations.

simple simulated counter-examples and theoretical analysis. In Section 4, we propose several objectives for PE from the martingale perspective, based on which we recover and interpret some well-studied PE algorithms. We also present numerical experiments for demonstration. Section 5 is devoted to some extensions of our problem formulation along with applications to option-like payoffs and linear-quadratic problems. In Section 6 we discuss the choice of test functions and the way to do function approximation from the algorithmic perspective. We conclude in Section 7. Appendix contains some supplementary materials and all the proofs.[3]

2. Problem Formulation and Preliminaries

Throughout this paper, by convention all vectors are *column* vectors unless otherwise specified, and $\mathbb{R}^k$ is the space of all k-dimensional vectors (hence $k \times 1$ matrices). Let A and B be two matrices of the same size. We denote by $A \circ B$ the inner product between A and B, by $|A|$ the Euclidean/Frobenius norm of A, and write $A^2 := AA^\top$, where $A^\top$ is A's transpose.

A general continuous-time RL problem can be formulated under the stochastic control framework with controlled Itô's stochastic differential equations (SDEs), analogous to MDPs in discrete time. However, since this paper concerns only a part (though a crucial part) of the RL problem, namely policy evaluation (PE) under a *fixed* control policy, we will formulate the problem *without* the control variable, which is the continuous-time counterpart of the Markov reward process (MRP) in discrete time.[4]

Let d, m be given positive integers, $T > 0$, and $b : [0, T] \times \mathbb{R}^d \mapsto \mathbb{R}^d$ and $\sigma : [0, T] \times \mathbb{R}^d \mapsto \mathbb{R}^{d\times m}$ be given functions. The *state* (or *feature*) dynamic follows a Markov diffusion process governed by an SDE:

$$\mathrm{d}X_s = b(s, X_s)\mathrm{d}s + \sigma(s, X_s)\mathrm{d}W_s, \tag{1}$$

such that for any given initial time–state pair $(t, x) \in [0, T] \times \mathbb{R}^d$, the SDE (1) with $X_t = x$ admits a solution $X = \{X_s, t \le s \le T\}$ on a certain filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}; \{\mathcal{F}_s\}_{s\ge t})$ along with a standard $\{\mathcal{F}_s\}_{s\ge t}$-adapted m-dimensional Brownian motion $W = \{W_s,\, s \ge t\}$. Note here we are concerned with the *weak* solution which includes the filtered probability space and the Brownian motion as part of the solution. See Karatzas and Shreve (2014) for various notions of solutions to an SDE.

Assuming the weak solution of (1) is unique (i.e. all the solutions have identical probability distribution, even if possibly with different sample paths), we define the *value function*

$$J(t, x) = \mathbb{E}\left[\int_t^T r(s, X_s)\mathrm{d}s + h(X_T)\Big| X_t = x\right], \tag{2}$$

where r is an (instantaneous) reward (cost) function (i.e. rate of reward/cost conditioned on time and state) and h the lump-sum reward (cost) function applied at the end of the planning period, T.

3. The codes to reproduce our results are publicly available at `https://www.dropbox.com/sh/5vyaw0yognhcabf/AACsArMcNmEuSwpXxcRq-qT1a?dl=0`.
4. PE sometimes is also referred to as the *prediction problem*. A general stochastic control formulation of RL can be founded in Wang et al. (2020), which will also be reviewed in Appendix B.

Unlike most RL problems that are formulated in an infinite planning horizon (known as *continuing tasks*), the current paper mainly focuses on a finite horizon setting (known as *episodic tasks*).[5] Finite horizons reflect limited lifespans of real-life tasks, e.g., a trader sells a financial contract with a maturity date, a robot finishes a task before a deadline, and a video gamer strives to pass a checkpoint given a time limit.

The PE task is, for a fixed given policy (which is suppressed in the formulation above due to the reason we stated earlier), to devise a numerical procedure to find $J(t,x)$ as a *function* of (t,x) using multiple sample trajectories of the process $\{s, X_s, r(s,X_s)\}_{t\leq s\leq T}$, where $\{X_s,\ t\leq s\leq T\}$ is the solution to (1), *without* the knowledge of the model parameters (the functional forms of b,σ,r,h). Hence we cover the settings of on-policy (i.e., the samples are generated under a *target* policy)[6], episodic (i.e., the same learning task is repeated for many episodes/multiple trajectories), offline (i.e., the approximated function is updated after a full episode/trajectory has been run) and online (i.e., the approximated function is updated in real time as we go). We emphasize that for a finite-horizon problem, it is generally too ambitious to expect an effective algorithm that learns from a *single* trajectory with no resets, due to the limited sample size. Learning with a single trajectory is usually done in an infinite horizon setting.

We make the following standard regularity assumptions on the coefficients of (1) and the reward function (2) to ensure the theoretical well-posedness of the problem:

Assumption 1 *The following conditions hold true:*

(i) b,σ,r,h *are all continuous functions in their respective arguments;*

(ii) b,σ *are uniformly Lipschitz in* x*, i.e., for* $\varphi=b,\ \sigma$*, there exists a constant* $C>0$ *such that*

$$|\varphi(t,x)-\varphi(t,x')|\leq C|x-x'|,\ \ \forall t\in[0,T],\ \ x,x'\in\mathbb{R}^d;$$

(iii) b,σ *have linear growth in* x*, i.e., for* $\varphi=b,\ \sigma$*, there exists a constant* $C>0$ *such that*

$$|\varphi(t,x)|\leq C(1+|x|),\ \ \forall(t,x)\in[0,T]\times\mathbb{R}^d;$$

(iv) r *and* h *both have polynomial growth in* x*, i.e., there exist constants* $C>0$ *and* $\mu\geq 1$ *such that*

$$|r(t,x)|\leq C(1+|x|^{\mu}),\ \ |h(x)|\leq C(1+|x|^{\mu}),\ \forall(t,x)\in[0,T]\times\mathbb{R}^d.$$

Under Assumption 1(i)-(iii), the SDE (1) admits a unique strong solution (and hence a unique weak solution) whose moments of any given order are uniformly bounded; see, e.g., Karatzas and Shreve (2014). The unique existence of a weak solution alone requires much weaker assumptions; see e.g. Stroock and Varadhan (1979), but we will not pursue along

5. We will briefly discuss the infinite horizon case with exponentially discounted payoff in Subsection 5.1.
6. Sutton et al. (2008) uses "behavioral policy" to describe the policy to generate observations and "target policy" to describe the policy we want to evaluate. Off-policy means training on data from a behavioral policy in order to learn the value of a target policy, and on-policy means that the behavioral policy coincides with the target policy in learning.

that line. On the other hand, Assumption 1(iv) is to ensure that $J(t,x)$ is finite for any (t,x).

We now recall some existing results on Markov diffusion processes underpinning the theoretical analysis in this paper. First, J can be characterized by a PDE based on the celebrated Feynman–Kac formula (Karatzas and Shreve, 2014):[7]

$$\begin{cases} \mathcal{L}J(t,x) + r\big(t,x\big) = 0, \ (t,x) \in [0,T) \times \mathbb{R}^d, \\ J(T,x) = h(x), \end{cases} \tag{3}$$

where

$$\mathcal{L}J(t,x) := \frac{\partial J}{\partial t}(t,x) + b\big(t,x\big) \circ \frac{\partial J}{\partial x}(t,x) + \frac{1}{2}\sigma^2\big(t,x\big) \circ \frac{\partial^2 J}{\partial x^2}(t,x)$$

is known as the *infinitesimal generator* associated with the diffusion process (1). Here, $\frac{\partial J}{\partial x} \in \mathbb{R}^d$ is the gradient, and $\frac{\partial^2 J}{\partial x^2} \in \mathbb{R}^{d\times d}$ is the Hessian.

The above PDE would be fully specified had the model been completely known.[8] If the state space has a dimension up to 4 (i.e. $d \le 4$), the equations can be efficiently solved numerically by methods such as Monte-Carlo and finite element algorithms. Unfortunately, in many practical applications the model parameters are not known, nor is the dimension small. Here, to avoid unnecessary technicality, we assume

Assumption 2 *The PDE* (3) *admits a classical solution $J \in C^{1,2}([0,T) \times \mathbb{R}^d)$ satisfying the polynomial growth condition, i.e., there exist constants $C > 0$ and $\mu \ge 1$ such that*

$$|J(t,x)| \le C(1 + |x|^{\mu}), \ \forall (t,x) \in [0,T] \times \mathbb{R}^d.$$

Second, the PDE (3) is related to the following forward–backward stochastic differential equation (FBSDE):

$$\begin{cases} \mathrm{d}X_s = b(s,X_s)\mathrm{d}s + \sigma(s,X_s)\mathrm{d}W_s, \ s \in [t,T]; \ \ X_t = x, \\ \mathrm{d}Y_s = -r\big(s,X_s\big)\mathrm{d}s + Z_s\mathrm{d}W_s, s \in [t,T]; \ Y_T = h(X_T). \end{cases} \tag{4}$$

Its solution, $\{(X_s, Y_s, Z_s), \ t \le s \le T\}$, has the following representations in terms of J:

$$Y_s = J(s,X_s), \ \ Z_s = \frac{\partial J}{\partial x}(s,X_s)^{\top}\sigma\big(s,X_s\big), \ \ s \in [t,T]. \tag{5}$$

The above relationship can be easily seen by applying Itô's formula to J; for details see El Karoui et al. (1997).

For any fixed $(t,x) \in [0,T] \times \mathbb{R}^d$ and $\{X_s, \ t \le s \le T\}$ solving the first equation of (4), define

$$M_s := J(s,X_s) + \int_t^s r\big(s', X_{s'}\big)\mathrm{d}s' \equiv Y_s + \int_t^s r\big(s', X_{s'}\big)\mathrm{d}s', \ \ s \in [t,T]. \tag{6}$$

The following result is the theoretical foundation of this paper, which characterizes the value function J by the martingality of M.

7. This PDE is a spacial case of the (nonlinear) Hamilton-Jacobi-Bellman (HJB) equation in continuous-time stochastic control when the control variable is fixed.
8. Some of this PDE's theoretical properties, such as existence, uniqueness, and regularity, have been well studied in terms of viscosity solution; see, e.g., Crandall et al. (1992); Beck et al. (2021).

Proposition 1 *Suppose Assumptions 1 and 2 hold. For any fixed* $(t, x) \in [0, T] \times \mathbb{R}^d$ *and* $\{X_s,\ t \leq s \leq T\}$ *solving the first equation of (4), the process* $M = \{M_s, t \leq s \leq T\}$ *is a square-integrable martingale. Conversely, if there is a continuous function* $\tilde{J}$ *such that for all* $(t, x) \in [0, T] \times \mathbb{R}^d$, $\tilde{M} = \{\tilde{M}_s, t \leq s \leq T\}$ *is a square-integrable martingale, where* $\tilde{M}_s := \tilde{J}(s, X_s) + \int_t^s r\big(s', X_{s'}\big)\mathrm{d}s'$, *and* $\tilde{J}(T, x) = h(x)$, *then* $\tilde{J} \equiv J$ *on* $[0, T] \times \mathbb{R}^d$.

This proposition inspires a martingale approach to PE in continuous-time RL, which will be developed in this paper. Essentially, the approach exploits the equivalence between PE (Feynman–Kac formula) and the martingality.

Finally, for a square-integrable semi-martingale $M = \{M_t, 0 \leq t \leq T\}$, its quadratic variation process, denoted by $\langle M \rangle = \{\langle M \rangle_t, 0 \leq t \leq T\}$, is defined to be (Karatzas and Shreve, 2014)

$$\langle M \rangle_t = \lim_{||\Delta|| \to 0} \sum_{i=0}^{K-1} (M_{\tau_i} - M_{\tau_{i-1}})^2 < \infty,$$

where $\Delta : 0 = \tau_0 < \cdots < \tau_K = t$ is an arbitrary partition of the interval $[0, t]$, and $||\Delta|| = \max_{1 \leq i \leq K}\{\tau_i - \tau_{i-1}\}$ is the largest mesh size. For M defined by (6), we have

$$\langle M \rangle_t = \langle Y \rangle_t = \int_0^t |Z_s|^2 ds, \quad t \in [0, T]. \tag{7}$$

Introduce

$$L^2_{\mathcal{F}}([0, T]) = \left\{ \kappa = \{\kappa_t, 0 \leq t \leq T\} \text{ is real-valued and } \mathcal{F}_t\text{-progressively measurable} : \mathbb{E} \int_0^T \kappa_t^2 \mathrm{d}t < \infty \right\}.$$

It is a Hilbert space with L^2-norm $||\kappa||_{L^2} = (\mathbb{E}\int_0^T \kappa_t^2 \mathrm{d}t)^{\frac{1}{2}}$. More generally, for any semi-martingale $Y = \{Y_s,\ s \geq 0\}$, we denote

$$L^2_{\mathcal{F}}([0, T]; Y) = \left\{ \kappa = \{\kappa_t, 0 \leq t \leq T\} :\ \kappa \text{ is } \mathcal{F}_t\text{-progressively measurable and } \mathbb{E} \int_0^T |\kappa_t|^2 \mathrm{d}\langle Y \rangle_t < \infty \right\}.$$

3. Temporal Difference Error in Continuous Time

In this section, we first review Doya (2000)'s TD error approach for deterministic dynamics and then explain why we can *not* extend that approach to the stochastic setting.

3.1 Doya's TD algorithm for deterministic dynamics

Many RL algorithms for discrete-time MDPs use TD error to evaluate policies. Doya (2000) extends the TD approach to RL with continuous time and space, albeit only for *deterministic* dynamics. For readers' convenience and for highlighting the key differences between deterministic and stochastic settings, we briefly review Doya (2000)'s approach here.

In our setting with $\sigma = 0$ (and hence all the expectations are dropped), Doya's approach starts with the obvious identity

$$J(t, X_t) = \int_t^{t'} r(s, X_s)\mathrm{d}s + J(t', X_{t'}), \quad t' \in (t, T]. \tag{8}$$

Rearranging this equation and dividing both sides by $t' - t$, we obtain

$$\frac{J(t', X_{t'}) - J(t, X_t)}{t' - t} + \frac{1}{t' - t} \int_t^{t'} r(s, X_s)\mathrm{d}s = 0. \tag{9}$$

Letting $t' \to t$ on the left hand side motivates the definition of the *TD error rate*:[9]

$$\delta_t := \dot{J}_t + r_t, \tag{10}$$

where $\dot{J}_t := \frac{d}{dt} J(t, X_t)$ is the total derivative of J along (t, X_t), and $r_t := r(t, X_t)$.

The *function approximation* approach widely employed for PE first approximates J by a parametric family of functions J^θ (upon using linear spans of basis functions or neural networks, or taking advantage of any known or plausible structure of the underlying problem), with $\theta \in \Theta \subseteq \mathbb{R}^L$. Henceforth, we always use θ-superscripted functions to denote those corresponding to the parameterized functions. For instance, $\delta_t^\theta := \dot{J}_t^\theta + r_t$.

Doya (2000) determines θ by minimizing the *mean–square TD error* (MSTDE)

$$\mathrm{MSTDE}(\theta) := \frac{1}{2} \int_0^T |\delta_t^\theta|^2 \mathrm{d}t \equiv \frac{1}{2} \int_0^T \left|\dot{J}_t^\theta + r_t\right|^2 \mathrm{d}t, \tag{11}$$

in view of the fact that this error *ought* to be zero theoretically.

To approximate and compute $\mathrm{MSTDE}(\theta)$, we discretize $[0, T]$ into small intervals $[t_i, t_{i+1}]$, $i = 0, 1, \cdots, K-1$, with an equal length Δt, where $t_0 = 0$ and $t_K = T$. This leads to

$$\mathrm{MSTDE}(\theta) \approx \frac{1}{2} \sum_{i=0}^{K-1} \left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1} - t_i} + r_{t_i} \right)^2 \Delta t =: \mathrm{MSTDE}_{\Delta t}(\theta). \tag{12}$$

A gradient descent algorithm is then applied to obtain the minimizer θ^* of $\mathrm{MSTDE}_{\Delta t}$ which in turn determines $J(t, x) = J^{\theta^*}(t, x)$. Namely,

$$\theta \leftarrow \theta - \alpha \sum_{i=0}^{K-1} \left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1} - t_i} + r_{t_i} \right) \left(\frac{\partial J^\theta}{\partial \theta}(t_{i+1}, X_{t_{i+1}}) - \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i}) \right), \tag{13}$$

where α is the learning rate (step size). This updating rule is also referred to as the *naïve residual gradient* method (Barnard, 1993; Baird, 1995).

The above algorithm is stated in the offline setting; namely, one uses the *whole* sample trajectory to update θ. However, TD-learning is often advocated for *online* learning: instead of observing the full sample path over $[0, T]$, one updates the estimate of the value function at each discrete time point using all available historical information. Take the most popular one-step method for example. With the time discretization described above, this method updates θ by

$$\theta \leftarrow \theta - \alpha \left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1} - t_i} + r_{t_i} \right) \left(\frac{\partial J^\theta}{\partial \theta}(t_{i+1}, X_{t_{i+1}}) - \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i}) \right).$$

9. Doya (2000) still refers this term as "TD error", while we add "rate" in its definition to reflects that it is indeed the *instantaneous* temporal difference at a given time t. However, we will use both terms interchangeably in this paper.

Notice that this increment is just one term in that of (13).

The most important feature of these TD-based algorithms that makes them implementable for learning is that one can *observe* the payoffs r_{t_i} and the states X_{t_i}, and hence can compute $J^\theta(t_i, X_{t_i})$, $i = 0, 1, \cdots, K-1$, through samples, *without* having to know the model parameters.

3.2 Mean-square TD error for stochastic dynamics

If we are to extend the MSTDE approach *naïvely* from Doya (2000)'s deterministic setting to the current stochastic (diffusion) setting, then we first note that the following equation, which is similar to (8), holds

$$J(t, X_t) = \mathbb{E}\left[\int_t^{t'} r(s, X_s)\mathrm{d}s + J(t', X_{t'})\Big|\mathcal{F}_t\right].\ \ t' \in (t, T]. \tag{14}$$

This equation is called *Bellman's consistency.* Then

$$\mathbb{E}\left[\frac{J(t', X_{t'}) - J(t, X_t)}{t' - t} + \frac{1}{t' - t}\int_t^{t'} r(s, X_s)\mathrm{d}s\right] = 0. \tag{15}$$

We may then be tempted to define a stochastic version of the TD error rate as in (10). Unfortunately, the path-wise total derivative $\dot{J}_t = \frac{d}{dt}J(t, X_t)$ no longer exists in the current diffusion case; hence, the TD error rate δ_t is not well defined now. This issue stems from the intrinsic non-differentiability of (non-degenerate) diffusion processes. For instance, it is well-known that with probability one, the sample trajectory of a Brownian motion is nowhere differentiable.

To facilitate our analysis without getting overly technical, we make the following regularity assumption on the value function approximators J^θ we use in this paper:

Assumption 3 *$J^\theta(t, x)$ is a sufficiently smooth function of (t, x, θ) so that all the derivatives required exist in the classical sense. Moreover, for all $\theta \in \Theta$, $J^\theta(\cdot, X_\cdot)$, $\mathcal{L}J^\theta(\cdot, X_\cdot) + r_\cdot$, $\left|\frac{\partial J^\theta}{\partial x}(\cdot, X_\cdot)^\top \sigma(\cdot, X_\cdot)\right| \in L^2_{\mathcal{F}}([0, T])$, and their L^2-norms are continuous functions of θ.*

Given an approximator J^θ, a theoretically well-defined error based on (14) in continuous time is the so-called *Bellman's error rate*:

$$\lim_{t' \to t+} \frac{1}{t' - t}\mathbb{E}\left[\int_t^{t'} r(s, X_s)\mathrm{d}s + J^\theta(t', X_{t'}) - J^\theta(t, X_t)\Big|\mathcal{F}_t\right] = \mathcal{L}J^\theta(t, X_t) + r\big(t, X_t\big). \tag{16}$$

This can be derived by applying Itô's formula to $J^\theta(t, X_t)$.

If there is no randomness in the environment, the conditional expectation in (16) vanishes and hence Bellman's error coincides with TD error (10) in the deterministic case. In a non-degenerate stochastic environment, however, only Bellman's error $\{\mathcal{L}J^\theta(t, X_t) + r\big(t, X_t\big), 0 \le t \le T\}$ is well defined on sample trajectories. Note that this error is zero everywhere for the true value function, according to (3). So it seems natural to set a PE objective to minimize Bellman's error. Unfortunately, this error accounts for the conditional expectation and thus represents the *average* of temporal differences over infinitely many sample trajectories that are distributed according to the SDE (1). Therefore, the knowledge about the state dynamics is *required* in computing the conditional expectation or, equivalently, in applying the operator $\mathcal{L}$. This knowledge is nevertheless unknown to the agent in our RL setting. In other words, in sharp contrast to the TD error, Bellman's error and its discretization version cannot be computed with only samples in a black-box environment.

On the other hand, even though MSTDE does not exist theoretically in the continuous-time stochastic setting, we can still define and compute its *discretization* version, in a way analogous to (12):

$$\mathrm{MSTDE}_{\Delta t}(\theta) := \frac{1}{2}\mathbb{E}\left[\sum_{i=0}^{K-1}\left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1}-t_i} + r_{t_i}\right)^2 \Delta t\right]. \tag{17}$$

Indeed, Wang and Zhou (2020) use this version to develop a PE algorithm for the mean–variance problem. A natural question is whether minimizing $\mathrm{MSTDE}_{\Delta t}(\theta)$ (or equivalently applying the stochastic version of the residual gradient algorithm (13)) will lead to the correct solution in the stochastic environment. The answer is unfortunately negative, as illustrated by the following example.

Example 1 Let us find a function that represents the conditional expectation $J(t,x) = \mathbb{E}[X_1|X_t = x]$, where $X_t = W_t$ is a Brownian motion. This is probably the simplest example possible. Because the Brownian motion is a martingale, we know the *ground truth* solution $J(t,x) = x$. Pretending we do not know this solution, we proceed to minimize $\mathrm{MSTDE}_{\Delta t}(\theta)$ to learn J based on the simulated sample paths of the state process $X \equiv W = \{W_t,\ 0 \le t \le 1\}$, which is a standard Brownian motion starting from $W_0 = 0$.

We first use a parameterized family $J^\theta(t,x) = [\theta(1-t)+1]x$ to approximate J. This family contains the true function when $\theta = 0$. The discretized MSTDE is

$$\mathrm{MSTDE}_{\Delta t}(\theta) = \frac{1}{2}\mathbb{E}\left[\sum_{i=0}^{K-1}\left(\frac{\big(\theta(1-t_{i+1})+1\big)X_{t_{i+1}} - \big(\theta(1-t_i)+1\big)X_{t_i}}{t_{i+1}-t_i}\right)^2 \Delta t\right].$$

We apply the stochastic gradient decent (SGD) with the updating rule

$$\theta \leftarrow \theta - \alpha\sum_{i=0}^{K-1}\left(\frac{\big(\theta(1-t_{i+1})+1\big)X_{t_{i+1}} - \big(\theta(1-t_i)+1\big)X_{t_i}}{t_{i+1}-t_i}\right)\left[(1-t_{i+1})X_{t_{i+1}} - (1-t_i)X_{t_i}\right].$$

In our simulation we use multiple independent episodes for training. We take the time grid size as $\Delta t = 0.01$, initialize the parameter to be $\theta^{(0)} = -1$, and apply the above updating rule with the learning rate $\alpha = 0.01$.

Figure 1 illustrates the convergence of θ to $\theta^*_{\mathrm{MSTDE}} = -\frac{3}{2}$ which is *not* the true solution $\theta_{\mathrm{true}} = 0$. In other words, the value function is not correctly learned by MSTDE. Equivalently, it does not solve the PDE (3) or the FBSDE (4) correctly.

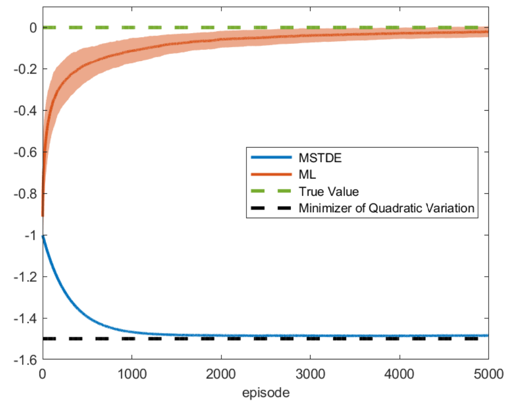

Figure 1: **The paths of parameters over episodes with different objectives for Example 1.** The true solution is $\theta_{\text{true}} = 0$. Applying SGD to minimize $\text{MSTDE}_{\Delta t}$ leads to $\theta^*_{\text{MSTDE}} = -\frac{3}{2}$. Applying SGD to minimize the martingale loss function generates the correct solution. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

3.3 Theoretical characterization of mean-square TD error

To understand *theoretically* why taking the objective of minimizing MSTDE does not work for stochastic problems, recall the processes (X, Y, Z) and the martingale M defined through (4)–(6) in which we take $t = 0$. Then

$$\begin{aligned}
&\sum_{i=0}^{K-1}\left(\frac{J(t_{t+1}, X_{t_{i+1}}) - J(t_i, X_{t_i})}{t_{i+1} - t_i} + r_{t_i}\right)^2 \Delta t \\
=&\frac{1}{\Delta t}\sum_{i=0}^{K-1}\left(J(t_{i+1}, X_{t_{i+1}}) - J(t_i, X_{t_i}) + \int_{t_i}^{t_{i+1}} r_{t_s}\mathrm{d}s + O\big((\Delta t)^2\big)\right)^2 \\
\approx&\frac{1}{\Delta t}\langle M\rangle_T = \frac{1}{(\Delta t)^2}\langle Y\rangle_T = \frac{1}{\Delta t}\int_0^T |Z_t|^2 \mathrm{d}t,
\end{aligned}$$

which is *not* zero, unlike in the deterministic setting. Hence, minimizing the MSTDE is wrong, because it is equivalent to minimizing the expected *quadratic variation* of the martingale M, which should *not* be minimized as the objective for estimating the value function.[10]

Take Example 1 again:

Example 1 (Continued) Let us use the previously taken parameterized family $Y_t^\theta = J^\theta(t, X_t) = [\theta(1-t) + 1]X_t$ to approximate J. Then

$$\mathrm{d}M_t^\theta \equiv \mathrm{d}Y_t^\theta = [\theta(1-t) + 1]\mathrm{d}W_t,$$

10. A related notion in financial econometrics is the *realized variance* of a time series, which is proved to be an unbiased estimate of the integrated variance or the quadratic variation, see, e.g., Barndorff-Nielsen and Shephard (2002).

leading to

$$\langle M^\theta \rangle_1 = \int_0^1 \big[1 + \theta(1-t)\big]^2 \mathrm{d}t = \frac{1}{3}\theta^2 + \theta + 1,$$

which is minimized at $\theta^* = -\frac{3}{2}$, instead of the desired value $\theta_{\text{true}} = 0$. This theoretical value matches the simulation result reported in Figure 1.

We now present a slightly more involved example, one that includes a running reward term.

Example 2 We seek a function representing the conditional expectation $J(t,x) = \mathbb{E}[X_1^2 - \int_t^1 \mathrm{d}s | X_t = x]$ where $X_t = W_t$ is a Brownian motion. Theoretically, the problem is equivalent to solving the following BSDE:

$$\mathrm{d}Y_t = \mathrm{d}t + Z_t \mathrm{d}W_t,\ Y_1 = X_1^2.$$

The true solution is $Y_t = X_t^2$, $Z_t = 2X_t$, namely, $J(t,x) = x^2$. If we use a parameterized family $Y_t^\theta = J^\theta(t, X_t) = [\theta_0(1-t)+1]X_t^2 + \theta_1(1-t)X_t + \theta_2(1-t)$ to approximate J, then the desired parameter values are $\theta_{\text{true}} = (0,0,0)$.

Let us compute the quadratic variation of $M_t^\theta := Y_t^\theta - t$. By Itô's lemma and replacing X_t by W_t, we obtain

$$\mathrm{d}M_t^\theta = (-\theta_2 - \theta_1 W_t - \theta_0 W_t^2 - 1)\mathrm{d}t + \big[2\big(\theta_0(1-t)+1\big)W_t + \theta_1(1-t)\big]\mathrm{d}W_t.$$

Then its expected quadratic variation is

$$\begin{aligned}
\mathbb{E}[\langle M^\theta \rangle_1] =& \mathbb{E}\int_0^1 \big[2\big(\theta_0(1-t)+1\big)W_t + \theta_1(1-t)\big]^2 \mathrm{d}t \\
=& \int_0^1 \Big[4\big(\theta_0(1-t)+1\big)^2 t + \theta_1^2(1-t)^2\Big]\,\mathrm{d}t \quad . \\
=& 4\left(\frac{1}{12}\theta_0^2 + \frac{1}{3}\theta_0 + \frac{1}{2}\right) + \frac{1}{3}\theta_1^2,
\end{aligned}$$

which attains minimum at $\theta_0^* = -2, \theta_1^* = 0$.

Here, the parameter θ_2 is not present in the expected quadratic variation, and hence remains undetermined. However, due to numerical errors in computing the TD error, we can determine θ_2 by minimizing the high-order small term in the quadratic variation, given the minimizer, (θ_0^*, θ_1^*), of the leading term. To do this, recall we have the following expansion:

$$(\mathrm{d}M_t^\theta)^2 = \underbrace{\cdots}_{\text{leading-order term}} (\mathrm{d}W_t)^2 + \underbrace{\cdots}_{\text{high-order small term}} (\mathrm{d}t)^2 + \underbrace{\cdots}_{\text{mean-zero term}} \mathrm{d}W_t \mathrm{d}t.$$

So, parameters will be determined first through the leading term in the quadratic variation. Parameters that do not show up in the leading term have much smaller but non-negligible impact on the TD error, which can be determined through the second term in the above. Finally, the mean-zero term can be ignored because it will be averaged out.

Therefore, in the current example, θ_2 will be determined through minimizing the following:

$$\mathbb{E}\int_0^1 (-\theta_2 - \theta_1^* W_t - \theta_0^* W_t^2 - 1)^2 \mathrm{d}t = \int_0^1 \Big[(\theta_2+1)^2 + 2(\theta_2+1)\theta_0^* t + 3\theta_0^{*2} t^2\Big]\,\mathrm{d}t.$$

The minimizer is $\theta_2^* = 0$. So, optimal parameters to minimize the MSTDE are $\theta^*_{\text{MSTDE}} = (-2, 0, 0)$ and hence the resulting learned function is $J(t, x) = (2t-1)x^2$. However, the true function is $J(t, x) = x^2$.[11]

We now verify this analysis by simulation. The discretized mean-square TD error is

$$\text{MSTDE}_{\Delta t}(\theta) = \frac{1}{2}\mathbb{E}\left[\sum_{i=0}^{K-1}\left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1} - t_i} - 1\right)^2 \Delta t\right].$$

We initialize the parameter to be $\theta^{(0)} = (-1, -1, -1)$, and use the SGD algorithm. The learning rate is taken as 0.01. The result, shown in Figure 2, is consistent with the above theoretical analysis, which incidentally justifies our scheme of determining some of the parameters through the high-order term.

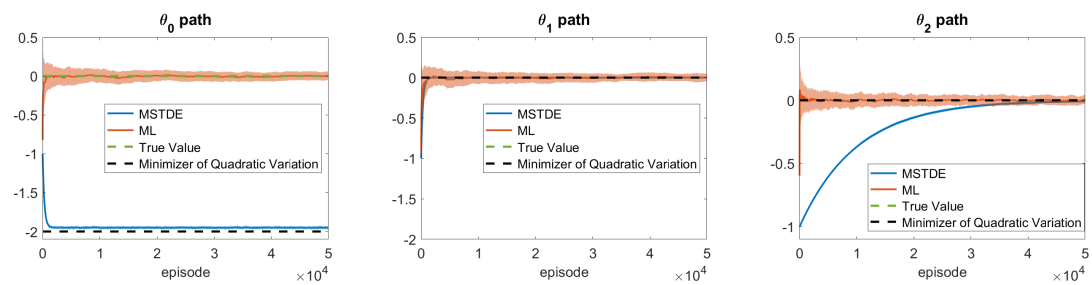

Figure 2: **The paths of parameters over episodes with different objectives for Example 2.** The true solution is $\theta_{\text{true}} = (0, 0, 0)$. Applying SGD to minimize mean-square TD error leads to $\theta^*_{\text{MSTDE}} = (-2, 0, 0)$. Applying SGD to minimize the martingale loss function leads to the desired solution. We repeat the experiment for 100 times to calculate the standard deviations of the predicted parameters, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

Next, we present a general result stipulating that any algorithm minimizing $\text{MSTDE}_{\Delta t}$ indeed converges to the minimizer of the quadratic variation of M^θ.

First, it follows from Itô's lemma that

$$dM_t^\theta = \left[\mathcal{L}J^\theta(t, X_t) + r_t\right]\mathrm{d}t + \left(\frac{\partial J^\theta(t, X_t)}{\partial x}\right)^\top \sigma(t, X_t)\mathrm{d}W_t.$$

Hence

$$d\langle M^\theta\rangle_t \equiv \left(\mathrm{d}M_t^\theta\right)^2 = \underbrace{\left[\left(\frac{\partial J^\theta(t, X_t)}{\partial x}\right)^\top \sigma(t, X_t)\right]^2 (\mathrm{d}W_t)^2}_{\text{leading-order term: quadratic variation}} + \underbrace{\left[\mathcal{L}J^\theta(t, X_t) + r_t\right]^2 (\mathrm{d}t)^2 + (\cdots)\mathrm{d}W_t\mathrm{d}t}_{\text{high-order small term}}. \tag{18}$$

11. Even though the two parameters θ_1^*, θ_2^* agree with the correct ones, this seems to be a coincidence and the final learned value function still deviates from the true one.

The first term on the right hand side is the leading order term since $(\mathrm{d}W_t)^2 = \mathrm{d}t$. Therefore, minimizing the left hand side is essentially to minimize this leading term.

Before presenting the theorem that formalizes this result, let us note that in the time-discretization throughout this paper, X_{t_i}, $i = 0, 1, \cdots, K$, are discrete *observations* of the continuous-time process X which is the *exact* solution to (1), instead of its approximation resulting from any numerical approximation scheme (such as the ones in Kloeden and Platen, 1992). So, in our paper the only approximation happens in evaluating the cumulative reward between two consecutive observations – we use the instantaneous reward r_{t_i} observed at time t_i multiplied by the length of the time window to approximate the total reward: $r_{t_i}\Delta t \approx \int_{t_i}^{t_{i+1}} r_{t_s}\mathrm{d}s$ – and in calculating the integral on $[0, T]$ by a discrete sum with the forward Euler scheme.

Clearly, the error of approximating cumulative reward is 0 if the running reward is a constant. When it is not a constant, the convergence rate of the approximation requires some conditions, which we put forward as an assumption.

Assumption 4 *There exist constants $C > 0$ and $\mu_1, \mu_2, \mu_3 \geq 0$, such that*

$$|r(t', x') - r(t, x)| \leq C|t' - t|^{\mu_1}|x' - x|^{\mu_2}(|x'|^{\mu_3} + |x|^{\mu_3}), \forall t', t \in [0, T],\ x', x \in \mathbb{R}^d.$$

Theorem 2 *Suppose Assumptions 1, 2, and 3 hold. Let*

$$\theta^*_{\mathrm{MSTDE}}(\Delta t) \in \arg\min_{\theta\in\Theta} \mathrm{MSTDE}_{\Delta t}(\theta)$$

and assume that $\theta^*_{\mathrm{MSTDE}} := \lim_{\Delta t\to 0} \theta^*_{\mathrm{MSTDE}}(\Delta t)$ exists. Then

$$\theta^*_{\mathrm{MSTDE}} \in \arg\min_{\theta\in\Theta} \mathbb{E}\int_0^T \left| \left(\frac{\partial J^\theta(t, X_t)}{\partial x}\right)^\top \sigma(t, X_t) \right|^2 \mathrm{d}t.$$

Moreover, if Assumption 4 also holds true, then

$$\mathbb{E}\int_0^T \left| \left(\frac{\partial J^{\theta^*_{\mathrm{MSTDE}}(\Delta t)}(t, X_t)}{\partial x}\right)^\top \sigma(t, X_t) \right|^2 \mathrm{d}t - \min_{\theta\in\Theta} \mathbb{E}\int_0^T \left| \left(\frac{\partial J^\theta(t, X_t)}{\partial x}\right)^\top \sigma(t, X_t) \right|^2 \mathrm{d}t \leq C\Delta t,$$

for some constant C.

In contrast, the true value function J solves the PDE (3) which corresponds to the coefficient of the $(\mathrm{d}t)^2$ term in (18). So once again the parameters should minimize the mean-square Bellman's error (which as discussed earlier depends on the model parameters and hence any algorithm trying to accomplish it is not implementable). This shows a fundamental discrepancy between the objective of MSTDE and that of PE in the stochastic diffusion environment.

The undesirability of the naïve residual gradient or MSTDE has actually been noticed in the discrete-time MDP literature. For example, Sutton et al. (2009) point out the similar problem of MSTDE and present a simple counterexample in an adsorbing three-state Markov chain. Sutton and Barto (2018, p.272) comment that "by penalizing all TD errors it (MSTDE) achieves something more like temporal smoothing than accurate prediction," although the authors stop short of explaining *why* it achieves temporal smoothing. Our theory confirms this intuition by a rigorous analysis showing that, in the diffusion setting, the MSTDE minimizer is primarily determined through minimizing quadratic variation. As quadratic variation measures the smoothness of a diffusion process, the value function process $\{J^\theta(t, X_t), 0 \le t \le T\}$ has the smoothest trajectory at $\theta = \theta^*_{\text{MSTDE}}$.

3.4 Online mean-square TD error algorithms

So far our discussions have been focused on the offline setting. However, TD is often used for online learning. The question now is whether an online algorithm may correct the errors arising from MSTDE.

Let us take the one-step online method for illustration. Precisely, suppose the time discretization is $0 = t_0 < t_1 < \cdots < t_K = T$. At each time t_{i+1}, $i = 0, \cdots, K-1$, this method updates θ by the following SGD algorithm:

$$\theta \leftarrow \theta - \alpha\left(\frac{J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i})}{t_{i+1} - t_i} + r_{t_i}\right)\left(\frac{\partial J^\theta}{\partial \theta}(t_{i+1}, X_{t_{i+1}}) - \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i})\right),$$

and then loop over all episodes.

Since multiple episodes are used, this procedure, by and large, can be viewed as drawing samples in the time direction uniformly (since the learning rate is constant, one does not differentiate among different times). Therefore, such a one-step updating rule is equivalent to solving

$$\begin{aligned}
&\min_\theta \mathbb{E}_{t\sim\mathcal{U}(0,T)}\left[\left(\frac{J^\theta(t+\Delta t, X_{t+\Delta t}) - J^\theta(t, X_t)}{\Delta t} + r_t\right)^2\right] \\
\approx &\min_\theta \frac{1}{\Delta t}\int_0^T \mathbb{E}[(\mathrm{d}M_t^\theta)^2] \approx \min_\theta \mathbb{E}[\int_0^T \mathrm{d}\langle M^\theta\rangle_t] = \min_\theta \mathbb{E}[\langle M^\theta\rangle_T],
\end{aligned}$$

where $\mathcal{U}(0,T)$ is the uniform distribution on $[0,T]$. This is the same optimization problem as the offline learning when we use the whole trajectory; hence, theoretically, it will lead to the same undesired solution that is determined by Theorem 2.

We revisit Examples 1 and 2 and implement the above online algorithm to minimize the mean-square TD error. Figures 3 and 4 show the results of the learned parameters respectively. As predicted by our analysis, they again converge to the same wrong solutions that are determined by minimizing quadratic variation.

4. Martingale Perspective and Approach

It follows from the previous section that MSTDE is not the right objective/loss function for PE in continuous-time *stochastic* RL. In this section we propose and analyze several other objective functions or criteria all based on the martingality of the process M, and connect some of them to well-studied alternative TD algorithms for MDPs.

4.1 Offline learning: Martingale loss function

In this subsection, we propose a loss function that uses full sample trajectories and is therefore applicable for offline learning, and test the corresponding algorithm's performance.

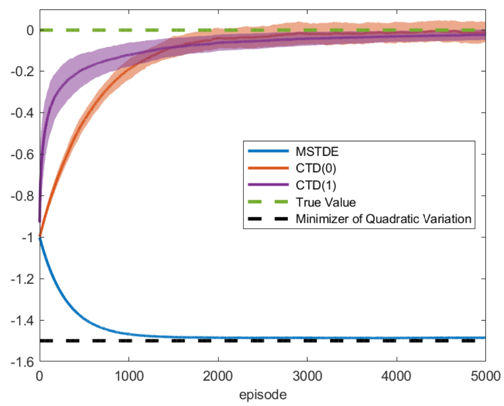

Figure 3: **The paths of parameters over episodes with different objectives under the online setting for Example 1.** The true solution is $\theta_{\text{true}} = 0$. Applying SGD to minimize one-step $\text{MSTDE}_{\Delta t}$ online leads to $\theta^*_{\text{MSTDE}} = -\frac{3}{2}$. CTD(0) and CTD(1) lead to the desired solution. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

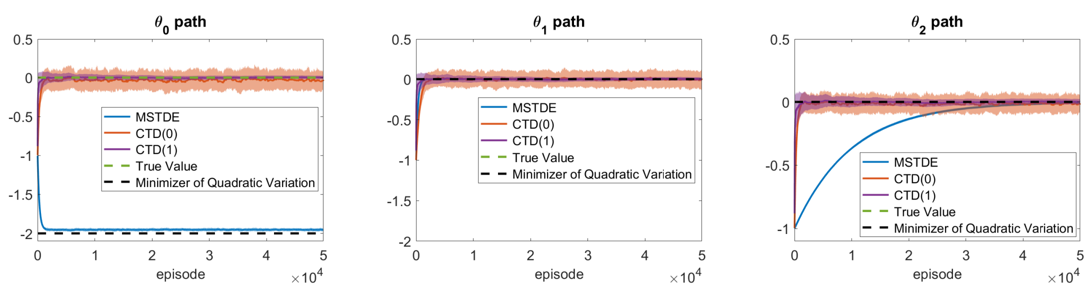

Figure 4: **The path of parameters over episodes for different objectives under the online setting for Example 2.** The true solution is $\theta_{\text{true}} = (0, 0, 0)$. Based on our analysis of quadratic variation, the minimizer is $\theta^*_{\text{MSTDE}} = (-2, 0, 0)$. CTD(0) and CTD(1) lead to the desired solution. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

Let the state process be $\{X_t, 0 \le t \le T\}$. Recall that the square-integrable martingality of $M_t = J(t, X_t) + \int_0^t r(s, X_s)\mathrm{d}s$ characterizes the correct value function. The martingale condition is further equivalent to

$$M_t = \mathbb{E}[M_T|\mathcal{F}_t], \quad \text{for all } t \in [0, T],$$

which in turn stipulates that the process at any given time prior to the terminal time T is the expectation of the terminal value conditional on all the information available at that time. However, a fundamental property of the conditional expectation yields that M_t minimizes the L^2-error between M_T and any $\mathcal{F}_t$-measurable random variables, namely,

$$M_t \equiv \mathbb{E}[M_T|\mathcal{F}_t] = \underset{\xi \text{ is } \mathcal{F}_t\text{-measurable}}{\arg\min} \mathbb{E}|M_T - \xi|^2, \quad \text{for all } t \in [0, T].$$

This property inspires the following loss function, termed the *martingale loss function*:

$$\begin{aligned}
\mathrm{ML}(\theta) :=& \frac{1}{2}||M_T - M_\cdot^\theta||_{L^2}^2 = \frac{1}{2}\mathbb{E}\int_0^T |M_T - M_t^\theta|^2 \mathrm{d}t \\
\approx& \frac{1}{2}\mathbb{E}\Big[\sum_{i=0}^{K-1}\Big(h(X_T) - J^\theta(t_i, X_{t_i}) + \int_{t_i}^T r(s, X_s)\mathrm{d}s\Big)^2 \Delta t\Big] \\
\approx& \frac{1}{2}\mathbb{E}\Big[\sum_{i=0}^{K-1}\Big(h(X_{t_K}) - J^\theta(t_i, X_{t_i}) + \sum_{j=i}^{K-1} r(t_j, X_{t_j})\Delta t\Big)^2 \Delta t\Big] \\
=& \frac{1}{2}\mathbb{E}\Big[\sum_{i=0}^{K-1}\Big(h(X_{t_K}) + \sum_{j=0}^{K-1} r(t_j, X_{t_j})\Delta t - J^\theta(t_i, X_{t_i}) - \sum_{j=0}^{i-1} r(t_j, X_{t_j})\Delta t\Big)^2 \Delta t\Big] \\
=& : \mathrm{ML}_{\Delta t}(\theta),
\end{aligned} \tag{19}$$

where $0 = t_0 < t_1 < \cdots < t_K = T$ is a mesh grid in time. Note that this loss function does not rely on the knowledge of the functional forms of b, σ, r or h.[12] As long as we can observe the accumulated reward $\sum_{j=0}^{i-1} r(t_j, X_{t_j})$ along with the final reward $h(X_T)$, the loss function can be implemented with the expectation replaced by the average over sample trajectories.

This loss function uses the whole trajectory to calculate the difference between the predicted value function and the realized reward-to-go, minimizing which naturally leads to an unbiased estimation. This approach is the continuous-time analogue of the so-called *Monte Carlo policy evaluation* with function approximation (Sutton and Barto, 2018) in the classical MDP and RL literature. It is primarily for offline learning where one observes multiple trajectories offline and updates estimate after observing one full trajectory.

The martingale loss objective is *not* of a TD type; it does not compare the approximate function values at two consecutive times. To explain the difference between the martingale loss function and the mean-square TD error, let us assume that the running reward $r \equiv 0$ for simplicity. In this case, $M_t = J(t, X_t)$ is a martingale. The martingale loss considers the difference values of J between any time instance and the final time, $J(X_T) - J(t_i, X_{t_i}) =$

12. In particular, $M_T = h(X_T) + \int_0^T r_s \mathrm{d}s$ does not depend on the parameter θ and can be directly observed from samples as the total reward obtained over $[0, T]$.

$h(X_T) - J(t_i, X_{t_i})$, while the TD concerns the difference between two consecutive time instances, $J(t_{i+1}, X_{t_{i+1}}) - J(t_i, X_{t_i})$. The intuition why the former leads to the right solution is that it always compares the current value of J with that of the final time, $h(X_T)$, which is observable and thus can serve as an ultimate and correct target. In fact, instead of thinking of $J(t, x)$ as a bivariate function of time t and space x, in any numerical procedure one is essentially looking for K functions of x, denoted by $J_i(\cdot) = J(t_i, \cdot)$, where $i = 0, \cdots, K-1$, with $J_K(x) = h(x)$ known and given. Therefore, the martingale loss is the aggregated error between J_i and $J_K = h$, minimizing which also minimizes the error between J_i and J_K for *each* i. As a result, each J_i converges to the correct value. In contrast, the *mean-square* TD error represents the aggregated intertemporal L^2 error between J_i and J_{i+1}. When computing this error, each J_i except J_0 shows up twice in $|J_i - J_{i+1}|^2$ and $|J_i - J_{i-1}|^2$; so the resulting function J_i will be twisted away from the true value, leading to a wrong solution.

Finally, we can apply SGD to minimize the proposed martingale loss function and the updating rule is given by

$$\theta \leftarrow \theta + \alpha \sum_{i=0}^{K-1} \left(h(X_{t_K}) + \sum_{j=i}^{K-1} r(t_j, X_{t_j})\Delta t - J^\theta(t_i, X_{t_i}) \right) \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i})\Delta t. \tag{20}$$

Let us call this the ML *(martingale loss) algorithm*, which is the counterpart of the *gradient Monte Carlo* in classical RL with MDP, when $G(t_i) := h(X_{t_K}) + \sum_{j=i}^{K-1} r(t_j, X_{t_j})\Delta t$ is taken as the Monte Carlo target at each t_i (Sutton and Barto, 2018). We apply this algorithm to numerically solve Examples 1 and 2, and find that it leads to the true solution; see Figures 1 and 2. In our implementation, the initial value is the same as before and the learning rate is tuned for smoother convergence. In particular, the initial learning rate is set to be 0.01 and decays according to $(\sharp\text{episode})^{-0.67}$ where $\sharp$episode denotes the number of episode.[13]

The next theorem states that minimizing the martingale loss function is equivalent to minimizing the mean-square error between the estimated value function J^θ and the true value function J. This error is known as the *mean-square value error* (MSVE):

$$\mathrm{MSVE}(\theta) = \mathbb{E} \int_0^T |J(t, X_t) - J^\theta(t, X_t)|^2 \mathrm{d}t. \tag{21}$$

Theorem 3 *It holds that*

$$\arg\min_{\theta \in \Theta} \mathrm{ML}(\theta) = \arg\min_{\theta \in \Theta} \mathrm{MSVE}(\theta).$$

Moreover, under Assumptions 1, 2, and 3, as $\Delta t \to 0$, *any convergent subsequence of the minimizer of the discretized martingale loss function* $\theta^*_{\mathrm{ML}}(\Delta t) \in \arg\min_{\theta\in\Theta} \mathrm{ML}_{\Delta t}(\theta)$ *converges to the minimizer of martingale loss function; that is*

$$\lim_{\Delta t \to 0} \theta^*_{\mathrm{ML}}(\Delta t) = \theta^*_{\mathrm{ML}} \in \arg\min_{\theta \in \Theta} \mathrm{ML}(\theta) = \arg\min_{\theta \in \Theta} \mathrm{MSVE}(\theta).$$

13. This decay schedule satisfies the usual requirement on the decay rate of the learning rate for the convergence of SGD algorithms. Note here our purpose is not to optimize convergence rates, but to confirm the limiting point for a convergent algorithm. Tuning the learning rate is not crucial to our results, as long as the algorithm does converge.

Furthermore, if in addition Assumption 4 holds, then

$$\mathrm{ML}\left(\theta^*_{\mathrm{ML}}(\Delta t)\right) - \min_{\theta\in\Theta} \mathrm{ML}(\theta) \le C(\Delta t)^{\min\{1,\mu_1+\frac{\mu_2}{2}\}}$$

for some constant $C > 0$.

Clearly, MSVE is a theoretically sound loss function for learning. However, by itself it does not lead to an *implementable* algorithm because the true value J is not observable from data. Theorem 3 strengthens the theoretical foundation of the martingale loss function that is implementable. Moreover, this theorem establishes the convergence together with the convergence rate of applying any convergent algorithm developed for minimizing discrete-time martingale loss as the time step tends to zero. Therefore, it also provides a theoretical foundation for implementing the discretization procedure.

We illustrate this result with the following examples.

Example 3 Consider the same learning problem in Example 1, but with a different parameterized value function given by $J^\theta(t,x) = \theta x^3$. Recall $X_t = W_t$ is a Brownian motion. The main difference between this example and the previous ones is that now the parametric family does *not* contain the true solution. Indeed, it does not even satisfy the correct terminal condition that $J^\theta(1,w) = x$, which could happen in more complex problems when the terminal payoff functions are unknown. Recall the true value function is $J(t,x) = x$. Let us compute the MSVE:

$$\mathbb{E}\int_0^1 |J(t,X_t) - J^\theta(t,X_t)|^2 \mathrm{d}t = \mathbb{E}\int_0^1 \left(W_t - \theta W_t^3\right)^2 \mathrm{d}t = \int_0^1 (t - 6\theta t^2 + 15\theta^2 t^3)\mathrm{d}t.$$

The minimizer is $\theta^* = \frac{4}{15}$. According to Theorem 3, minimizing the martingale loss function should converge to this solution.

Example 4 Consider the same learning problem in Example 1, with the parameterized value function $J^\theta(t,x) = x + (1-t)e^{\theta x - \frac{1}{2}\theta^2 t + \theta}$. Recall $X_t = W_t$ is a Brownian motion. This time it satisfies the terminal condition, but still does not contain the true solution. Let us compute the MSVE:

$$\begin{aligned}&\mathbb{E}\int_0^1 |J(t,X_t) - J^\theta(t,X_t)|^2 \mathrm{d}t = \mathbb{E}\int_0^1 (1-t)^2 e^{2\theta W_t - \theta^2 t + 2\theta}\mathrm{d}t \\ =&\int_0^1 (1-t)^2 e^{\theta^2 t + 2\theta}\mathrm{d}t = -\frac{e^{2\theta}(2 - 2e^{\theta^2} + 2\theta^2 + \theta^4)}{\theta^6}.\end{aligned}$$

The minimizer is $\theta^* \approx -2.12568$. According to Theorem 3, minimizing the proposed martingale loss function should converge to this solution.

We test the numerical solutions to Examples 3 and 4 by applying our ML algorithms with SGD. The initial learning rate is taken as 0.001 and decays as $(\sharp\text{episode})^{-0.67}$. Figures 5 and 6 confirm the result of Theorem 3.

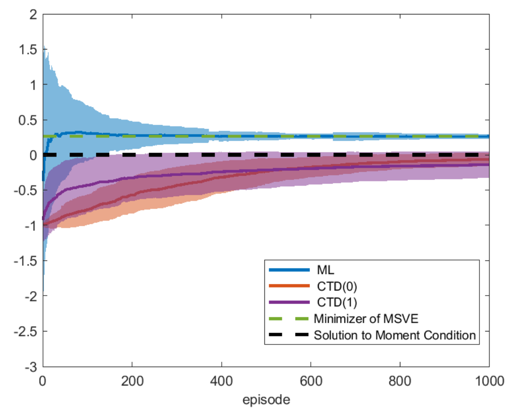

Figure 5: **ML and CTD(λ) methods converge to different points for Example 3.** Applying ML algorithm leads to $\theta^*_{\text{ML}} = \frac{4}{15}$, which is the minimizer of MSVE. CTD methods converge to $\theta^*_{\text{moment}} = 0$, which is the solution to the moment condition. In this case, the moment conditions associated with CTD(0) and CTD(1) have the same solution so the two algorithms converge to the same point. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

4.2 Online and offline learning: TD based on martingale orthogonality conditions

We have proposed a martingale loss function to interpret the Monte Carlo PE. This approach requires the whole sample trajectory over $[0, T]$; so it is inherently offline and is difficult to extend to the online setting where only historical samples are available when one updates the approximated function in real time. Classically, TD learning was introduced as a remedy to enable online learning. However, based on our previous discussion, the mean-square TD error is not the correct objective function to learn the value function even though it can indeed be implemented online. In this section, we propose a different approach, again based on the martingality of the process M, that generates the continuous-time counterparts of several well-studied TD algorithms and that works both online and offline.

This approach starts with noting that M being a square-integrable martingale implies

$$\mathbb{E} \int_0^T \xi_t \mathrm{d}M_t = 0, \tag{22}$$

for any $\xi \in L^2_{\mathcal{F}}([0,T], M)$ (called a *test function*).[14] In fact, the following result shows that this is a necessary and sufficient condition for the parameterized process M^θ_t to be a martingale.

Proposition 4 *In general, a diffusion process $M^\theta \in L^2_{\mathcal{F}}([0,T])$ is a martingale if and only if $\mathbb{E}\int_0^T \xi_t \mathrm{d}M^\theta_t = 0$ for any $\xi \in L^2_{\mathcal{F}}([0,T], M^\theta)$. In the current setting, $\mathbb{E}\int_0^T \xi_t \mathrm{d}M^\theta_t = \mathbb{E}\int_0^T \xi_t \big[\mathcal{L}J^\theta(t, X_t) + r_t\big]\mathrm{d}t$.*

14. It would be more appropriate to call it a *test process* because ξ needs to be generally an adapted stochastic process. However, we use the more common term "test function".

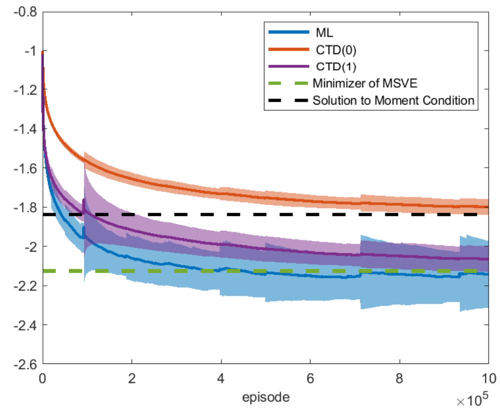

Figure 6: **ML and CTD(λ) methods converge to different points with different values of λ for Example 4.** Applying ML algorithm leads to $\theta^*_{ML} \approx -2.12568$, which is the minimizer of MSVE. CTD(0) converges to $\theta^*_{\text{moment,CTD(0)}} \approx -1.83923$, which is the solution to the moment condition associated with the choice of test function for CTD(0). CTD(1) converges to $\theta^*_{\text{moment,CTD(1)}} \approx -2.12568$, which is the solution to the moment condition associated with the choice of test function for CTD(1). Because of the different choices of test functions, the two CTD algorithms converge to different points. It is a coincidence that ML and CTD(1) converge to the same point. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

We call (22) a family of *martingale orthogonality conditions*. In theory, one should vary all possible test functions and thus this family has *infinitely* many equations. For numerical approximation methods, however, we can choose finitely many test functions in special forms. Notice that, for a parametric family $\{J^\theta : \theta \in \Theta \subset \mathbb{R}^L\}$, in principle, we need at least L equations as our martingale orthogonality conditions in order to fully determine θ. For example, we can take $\xi_t = \frac{\partial J^\theta}{\partial \theta}(t, X_t) \in \mathbb{R}^L$. (Here, and henceforth, ξ_t may be vector-valued and (22) is accordingly a vector-valued equation or a system of equations.) In statistics and econometrics, a problem of the type (22) with a finite number of equations is also referred to as *moment conditions*, and a systematic way to analyze and solve it is known as the *generalized methods of moments* (GMM); see, e.g., Hansen (1982).

To devise algorithms based on this theory, we need to answer the following questions: How to choose these finite number of test functions? And how to solve the resulting moment conditions in an effective and efficient way? It turns out that answering these two questions suitably in our continuous setting gives rise to algorithms that correspond to several well-known conventional TD learning algorithms in discrete setting.

- Choose $\xi_t = \frac{\partial J^\theta}{\partial \theta}(t, X_t)$, and use stochastic approximation (Robbins and Monro, 1951) to update parameters after a whole episode (offline):

$$\theta \leftarrow \theta + \alpha \int_0^T \frac{\partial J^\theta}{\partial \theta}(t, X_t) \mathrm{d}M_t^\theta \approx \theta + \alpha \sum_{i=0}^{K-1} \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i}) \big(J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i}) + r_{t_i} \Delta t \big),$$

or to update parameters at every time step (online):

$$\theta \leftarrow \theta + \alpha \frac{\partial J^\theta}{\partial \theta}(t, X_t) \mathrm{d}M_t^\theta \approx \theta + \alpha \frac{\partial J^\theta}{\partial \theta}(t_i, X_{t_i}) \big(J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i}) + r_{t_i} \Delta t \big).$$

These algorithms correspond to the TD(0) algorithm (Sutton, 1988).

- Choose $\xi_t = \int_0^t \lambda^{t-s} \frac{\partial J^\theta}{\partial \theta}(s, X_s) \mathrm{d}s$, where $0 < \lambda \leq 1$, and use stochastic approximation to update parameters after one episode:

$$\begin{aligned} \theta \leftarrow & \theta + \alpha \int_0^T \int_0^t \lambda^{t-s} \frac{\partial J^\theta}{\partial \theta}(s, X_s) \mathrm{d}s \mathrm{d}M_t^\theta \\ & \approx \theta + \alpha \sum_{i=0}^{K-1} \sum_{j=0}^{i} \Delta t \lambda^{(i-j)\Delta t} \frac{\partial J^\theta}{\partial \theta}(t_j, X_{t_j}) \big(J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i}) + r_{t_i} \Delta t \big), \end{aligned}$$

or to update parameters at every time step:

$$\begin{aligned} \theta \leftarrow & \theta + \alpha \int_0^t \lambda^{t-s} \frac{\partial J^\theta}{\partial \theta}(s, X_s) \mathrm{d}s \mathrm{d}M_t^\theta \\ & \approx \theta + \alpha \sum_{j=0}^{i} \Delta t \lambda^{(i-j)\Delta t} \frac{\partial J^\theta}{\partial \theta}(t_j, X_{t_j}) \big(J^\theta(t_{i+1}, X_{t_{i+1}}) - J^\theta(t_i, X_{t_i}) + r_{t_i} \Delta t \big). \end{aligned}$$

These algorithms correspond to the TD(λ) algorithm (Sutton, 1988). Here the parameter λ dictates how much weight to be put on historical predictions in the procedure. When $\lambda = 1$, it puts equal weight on all the past predictions. The smaller λ becomes, the more weight on more recent predictions. When $\lambda = 0$, past predictions do not matter, resulting in the TD(0) algorithm.

It should be noted that TD(0) and TD(λ) algorithms are *not* exactly gradient based; rather, they use *stochastic approximation* as a first-order method to solve (22). In the literature they are also referred to as *semi-gradient* TD algorithms (Sutton and Barto, 2018) because they do include a part of the gradient.

- Choose $\xi_t = \frac{\partial J^\theta}{\partial \theta}(t, X_t)$ and take the approximated value function to be a linear combination of a series of basis functions: $J^\theta(t,x) = \sum_{j=1}^L \theta_j \phi_j(t,x)$. Then $\frac{\partial J^\theta}{\partial \theta}(t,x) = \phi(t,x) := (\phi_1(t,x), \cdots, \phi_L(t,x))^\top \in \mathbb{R}^L$. In this case, (22) becomes a system of linear equatins in θ and can be solved explicitly as

$$\theta = -\left[\mathbb{E} \int_0^T \phi(t, X_t) \big(\mathrm{d}\phi(t, X_t)^\top \big) \right]^{-1} \mathbb{E} \int_0^T r_t \phi(t, X_t) \mathrm{d}t,$$

assuming the matrix inverse exists. The expectation can be estimated using sample average across multiple trajectories. Hence, if there are M episodes, we have

$$\begin{aligned} \theta = & -\left(\frac{1}{M} \sum_{k=1}^{M} \int_0^T \phi(t, X_t^{(k)}) \big(\mathrm{d}\phi(t, X_t^{(k)})^\top \big) \right)^{-1} \left(\frac{1}{M} \sum_{k=1}^{M} \int_0^T r_t^{(k)} \phi(t, X_t^{(k)}) \mathrm{d}t \right) \\ \approx & -\left(\frac{1}{M} \sum_{k=1}^{M} \sum_{i=0}^{K-1} \phi(t_i, X_{t_i}^{(k)}) \big(\phi(t_{i+1}, X_{t_{i+1}}^{(k)})^\top - \phi(t_i, X_{t_i}^{(k)})^\top \big) \right)^{-1} \left(\frac{1}{M} \sum_{k=1}^{M} \sum_{i=0}^{K-1} r_{t_i}^{(k)} \phi(t_i, X_{t_i}^{(k)}) \Delta t \right), \end{aligned}$$

where the superscript (k) signifies that the corresponding observations are taken from the k-th episode. If there is only one trajectory up to time $t = t_j$, then we estimate the parameter using long-time average (under certain ergodicity condition) to obtain

$$
\begin{aligned}
\theta &= -\left(\frac{1}{t}\int_0^t \phi(s, X_s)\big(\mathrm{d}\phi(s, X_s)^\top\big)\right)^{-1}\left(\frac{1}{t}\int_0^t r_s\phi(s, X_s)\mathrm{d}s\right)\\
&\approx -\left(\frac{1}{j}\sum_{i=0}^{j-1}\phi(t_i, X_{t_i})\big(\phi(t_{i+1}, X_{t_{i+1}})^\top - \phi(t_i, X_{t_i})^\top\big)\right)^{-1}\left(\frac{1}{j}\sum_{i=0}^{j-1} r_{t_i}\phi(t_i, X_{t_i})\Delta t\right).
\end{aligned}
$$

These algorithms correspond to the (linear) least square TD(0), or LSTD(0), algorithms (Bradtke and Barto, 1996). LSTD and its variants (Boyan, 2002) are often discussed in the context of linear function approximation. Despite the name of "least square", it does not solve any minimization problem per se; instead it uses the linear structure to obtain the exact solution to (22) and then use sample average to estimate the expectation. Xu et al. (2002) and Geramifard et al. (2006) develop a more efficient way to implement this solution in a recursive way. The reason why it is called "least square" comes from the instrumental variable approach to regression problems (Ljung and Söderström, 1983).[15] Bradtke and Barto (1996) show that the basis functions in LSTD are indeed instrumental variables.

- We choose $\xi_t = \frac{\partial J^\theta}{\partial \theta}(t, X_t)$, and minimize the GMM objective function

$$
\mathrm{GMM}(\theta) = \frac{1}{2}\mathbb{E}\left[\int_0^T \xi_t \mathrm{d}M_t^\theta\right]^\top A\,\mathbb{E}\left[\int_0^T \xi_t \mathrm{d}M_t^\theta\right],
$$

where A is a given matrix. Different choices of A lead to a variety of algorithms corresponding to what are broadly called *gradient TD* (GTD) algorithms for MDPs. For example, taking $A = I$, the identity matrix, corresponds to GTD(0) by Sutton et al. (2009). Another choice is $A = [\mathbb{E}\int_0^T \xi_t\xi_t^\top \mathrm{d}t]^{-1}$. In this case, the gradient of the objective in θ is (noting ξ_t also depends on θ)

$$
\mathbb{E}\left[\int_0^T \mathrm{d}\left(\frac{\partial M^\theta}{\partial\theta}(t, X_t)\right)\xi_t^\top\right]u + \mathbb{E}\left[\int_0^T \frac{\partial \xi_t}{\partial\theta}^\top \mathrm{d}M_t^\theta\right]u - \mathbb{E}\left[\int_0^T u^\top\xi_t\frac{\partial\xi_t}{\partial\theta}(t, X_t)^\top u\mathrm{d}t\right],
$$

where $u := [\mathbb{E}\int_0^T \xi_t\xi_t^\top\mathrm{d}t]^{-1}\mathbb{E}[\int_0^T \xi_t\mathrm{d}M_t^\theta]$ and $\frac{\partial\xi_t}{\partial\theta}$ is the Jacobian matrix. In particular, $\frac{\partial\xi_t}{\partial\theta} = \frac{\partial^2 J^\theta}{\partial\theta^2}(t, X_t)$ is the Hessian matrix and hence is symmetric. When $J^\theta(t,x) = \sum_{j=1}^L \theta_j\phi_j(t,x)$ is the linear span of basis functions, the last two terms of the gradient will vanish because $\frac{\partial\xi_t}{\partial\theta} = \frac{\partial^2 J^\theta}{\partial\theta^2}(t, X_t) = 0$.

Two GTD algorithms, called GTD2 and TDC (Sutton et al., 2008, 2009), apply stochastic approximation at two different time scales to update u and θ respectively.

15. Instrumental variables are widely used in statistics and econometrics to estimate causal relationship when exploratory variables are endogenous. A necessary condition for being a instrumental variable is that it must be uncorrelated with the residual term. In the context of TD learning, the residual term is the TD error.

Specifically, in both algorithms, u is estimated with long-term average:

$$u \leftarrow u + \alpha_u \left[\xi_t \mathrm{d}M_t^\theta - \xi_t \xi_t^\top u \Delta t\right] \approx u + \alpha_u \big[\xi_{t_i}(M^\theta_{t_{i+1}} - M^\theta_{t_i}) - \xi_{t_i}\xi_{t_i}^\top u \Delta t\big],$$

and then θ is updated with two different one-step sampling methods. The GTD2 algorithm proceeds as follows:

$$\begin{aligned}\theta \leftarrow &\theta - \alpha_\theta \left[\mathrm{d}\left(\frac{\partial M^\theta}{\partial \theta}(t, X_t)\right)\xi_t^\top u + \frac{\partial \xi_t}{\partial \theta}^\top \mathrm{d}M_t^\theta u - u^\top \xi_t \frac{\partial \xi_t}{\partial \theta}^\top u \Delta t\right]\\ \approx &\theta - \alpha_\theta \Bigg[\left(\frac{\partial M^\theta}{\partial \theta}(t_{i+1}, X_{t_{i+1}}) - \frac{\partial M^\theta}{\partial \theta}(t_i, X_{t_i})\right)\xi_{t_i}^\top u\\ &+ \frac{\partial \xi}{\partial \theta}(t_i, X_{t_i})^\top (M^\theta_{t_{i+1}} - M^\theta_{t_i})u - u^\top \xi_{t_i}\frac{\partial \xi}{\partial \theta}(t_i, X_{t_i})^\top u \Delta t\Bigg].\end{aligned}$$

The TDC algorithm observes that $\frac{\partial J^\theta}{\partial \theta}(t, X_t) = \xi_t$ and hence updates θ by

$$\begin{aligned}\theta \leftarrow &\theta - \alpha_\theta \left[\xi_t \mathrm{d}M_t^\theta + \xi_{t'}\xi_t^\top u \Delta t + \frac{\partial \xi_t}{\partial \theta}^\top \mathrm{d}M_t^\theta u - u^\top \xi_t \frac{\partial \xi_t}{\partial \theta}^\top u \Delta t\right]\\ \approx &\theta - \alpha_\theta \Bigg[\xi_{t_i}(M^\theta_{t_{i+1}} - M^\theta_{t_i}) + \xi_{t_{i+1}}\xi_{t_i}^\top u \Delta t\\ &+ \frac{\partial \xi}{\partial \theta}(t_i, X_{t_i})^\top (M^\theta_{t_{i+1}} - M^\theta_{t_i})u - u^\top \xi_{t_i}\frac{\partial \xi}{\partial \theta}(t_i, X_{t_i})^\top u \Delta t\Bigg].\end{aligned}$$

GTD(0), GTD2 and TDC are gradient based methods as well as typical GMM methods to minimize a quadratic form of the conditions (22), where expectations are estimated using long term averages as in Hansen et al. (1996). Sutton et al. (2008, 2009) and Maei et al. (2009) study stochastic approximation and incremental implementation of the gradient of quadratic functions for linear and non-linear function approximation respectively.

All the above methods can be classified into two types. The first type applies stochastic approximation to solve the moment conditions directly, like TD(λ). This is the classical TD learning. The second type follows GMM to minimize a quadratic function of the moment conditions by computing its gradient and approximating the expectation by either long-term average or one long sample trajectory. We call it the GTD method, following Sutton et al. (2008). LSTD is limited to linear approximation only and hence can be considered as a special case of the first type when the moment conditions can be explicitly solved so the only computation needed is to estimate the expectation using samples.

It should be noted that although the goal of this paper is to devise PE algorithms for the continuous setting, the *actual* implementations of the various algorithms described above are all discrete-time with a *fixed* mesh size Δt. These algorithms correspond to some discrete-time versions of the moment conditions. So natural and important theoretical questions are whether such an algorithm converges to the solution of the continuous-time version of the respective moment conditions as $\Delta t \to 0$ and, if yes, what the convergence rate is. The next two theorems answer these questions.

Henceforth we impose the following assumption on the test functions used for moment conditions.

Assumption 5 *A test function $\xi = \{\xi_t, 0 \le t \le T\}$ is an $\mathbb{R}^{L'}$-valued adapted process satisfying $|\xi| \in L^2_{\mathcal{F}}([0,T]; M^\theta)$ and $\mathbb{E}[|\xi_{t'} - \xi_t|^2] \le C(\theta)|t'-t|^\alpha$ for any $t, t' \in [0,T]$, where $C(\theta)$ is a continuous function of θ and $0 < \alpha \le 2$ is a given constant.*

The following is about the convergence of the TD type algorithms when $\Delta t \to 0$.

Theorem 5 *Denote by $\theta^*_{\text{moment}}(\Delta t)$ the solution to the discrete-time moment conditions with mesh size Δt:*

$$\mathbb{E}\sum_{i=0}^{K-1} \xi_{t_i}(M^\theta_{t_{i+1}} - M^\theta_{t_i}) = 0.$$

Then, under Assumptions 1, 2, 3, and 5, as $\Delta t \to 0$, any convergent subsequence of $\theta^*_{\text{moment}}(\Delta t)$ converges to the solution to the continuous-time moment conditions (22)*; that is,*

$$\theta^*_{\text{moment}} := \lim_{\Delta t \to 0} \theta^*_{\text{moment}}(\Delta t)$$

solves (22)*. Moreover, if in addition Assumption 4 holds, then*

$$|\mathbb{E}\int_0^T \xi_t \mathrm{d}M_t^{\theta^*_{\text{moment}}(\Delta t)}| \le C(\Delta t)^{\min\{\frac{\alpha}{2}, \mu_1 + \frac{\mu_2}{2}\}}$$

for some constant C.

The next theorem is on the convergence of the GTD type algorithms when $\Delta t \to 0$.

Theorem 6 *Let the discretized GMM objective function be*

$$\text{GMM}_{\Delta t}(\theta) := \frac{1}{2}\mathbb{E}\left[\sum_{i=0}^{K-1} \xi^\theta_{t_i}(M^\theta_{t_{i+1}} - M^\theta_{t_i})\right]^\top A_{\Delta t}\mathbb{E}\left[\sum_{i=0}^{K-1} \xi^\theta_{t_i}(M^\theta_{t_{i+1}} - M^\theta_{t_i})\right],$$

where $A_{\Delta t}$ is a discretized approximation of A satisfying $|A_{\Delta t} - A| \le \tilde{C}(\theta)|\Delta t|^\beta$, with $\tilde{C}(\theta)$ being a continuous function of θ and $\beta > 0$ a constant.[16] *Then, under Assumptions 1, 2, 3, and 5, as $\Delta t \to 0$, any convergent subsequence of the minimizer of the discretized GMM objective function $\theta^*_{\text{GMM}}(\Delta t) \in \arg\min_{\theta\in\Theta} \text{GMM}_{\Delta t}(\theta)$ converges to the minimizer of the continuous GMM objective function; that is,*

$$\lim_{\Delta t \to 0} \theta^*_{\text{GMM}}(\Delta t) = \theta^*_{GMM} \in \arg\min_{\theta\in\Theta} \text{GMM}(\theta).$$

Moreover, if in addition Assumption 4 holds, then

$$\text{GMM}\left(\theta^*_{\text{GMM}}(\Delta t)\right) - \min_{\theta\in\Theta} \text{GMM}(\theta) \le C(\Delta t)^{\min\{\frac{\alpha}{2}, \mu_1 + \frac{\mu_2}{2}, \beta\}}$$

for some constant C.

16. When A is a constant as in GTD(0), $A_{\Delta t} = A$. When $A = [\mathbb{E}\int_0^T \xi_t\xi_t^\top \mathrm{d}t]^{-1}$ as in GTD2 and TDC, $A_{\Delta t} := [\mathbb{E}\sum_{i=0}^{K-1} \xi_{t_i}\xi_{t_i}^\top \Delta t]^{-1}$ is the discretization approximation of this integral.

From now on, to distinguish our algorithms from their existing discrete-time counterparts, we will add a prefix "C", signifying "continuous", to the names of the algorithms. For instance, we will call them CTD(λ), CLSTD, and so on.

The next important question is in what sense the aforementioned methods approximate the correct value function. First, a convergent CTD(0) or CTD(λ) algorithm should converge to the solution to the moment conditions (22) based on the respective choices of test functions. Intuitively, such an algorithm searches for one particular Bellman's error process $\mathcal{L}J^\theta(t, X_t) + r_t$ *within the parametric family* such that it is orthogonal to the underlying test functions. These TD learning methods are usually easy to implement and work effectively in many applications. To demonstrate, we re-compute the problems in Examples 1 and 2 using online CTD(0) and CTD(1) algorithms with stochastic approximation. The learning rate is chosen as 0.01. Both algorithms converge to the correct values; see Figures 3 and 4.

However, a caveat is that these algorithms may not always work. On one hand, due to possible misspecification of the parametric family, solutions to the moment conditions may not exist, in which case the algorithms will not converge; see Example 5 below where the test function is not properly chosen. On the other hand, as the following continuations of Examples 3 and 4 illustrate, even if the solution to the moment conditions exists uniquely and an algorithm converges, the resulting solution may vary depending on the choice of test functions.

Example 5 Consider the same learning problem in Example 1, now with the parameterized value function $J^\theta(t, x) = x + (1-t)e^{\theta x - \frac{1}{2}\theta^2 t}[(\theta+1)^2 + 1]$. Recall $X_t = W_t$ is a Brownian motion. This family does not contain the true solution. If we choose the test function to be the constant 1 and use CTD(0), a convergent algorithm should solve

$$0 = \mathbb{E}\int_0^1 e^{\theta W_t - \frac{1}{2}\theta^2 t}[(\theta+1)^2 + 1]\mathrm{d}t = (\theta+1)^2 + 1.$$

However, there is no solution to the above equation. Our implementation of CTD(0) indeed generates a divergent sequence of iterates; see Figure 7.

On the other hand, we can use the martingale loss function to get a solution. Indeed, it follows from Theorem 3 that the ML algorithm is equivalent to minimizing

$$\begin{aligned}
&\mathbb{E}\int_0^1 |J(t, X_t) - J^\theta(t, X_t)|^2 \mathrm{d}t = \mathbb{E}\int_0^1 (1-t)^2 e^{2\theta W_t - \theta^2 t}[(\theta+1)^2 + 1]^2 \mathrm{d}t \\
&= \int_0^1 (1-t)^2 e^{\theta^2 t}\mathrm{d}t[(\theta+1)^2 + 1]^2 = -\frac{2 - 2e^{\theta^2} + 2\theta^2 + \theta^4}{\theta^6}[(\theta+1)^2 + 1]^2,
\end{aligned}$$

whose minimizer is around -0.875301. The implementation of ML confirms this theoretical prediction; see Figure 7.

Example 3 (Continued) We revisit this example where $J^\theta(t, x) = \theta x^3$. Recall $X_t = W_t$ is a Brownian motion. There is no running reward so $M_t^\theta = \theta X_t^3$ and $\mathrm{d}M_t^\theta = 3\theta W_t^2 \mathrm{d}W_t + 3\theta W_t \mathrm{d}t$. Hence, any test function that is not identically 0 leads to the *only* solution $\theta^* = 0$. As a result, any convergent CTD algorithm should converge to 0, yielding a value function $J^{\theta^*}(t, x) = 0$ that is significantly deviated from the true one $J(t, x) = x$. See Figure 5 for the CTD(0) and CTD(1) experiment results.

Example 4 (Continued) Consider the parameterized value function $J^\theta(t,x) = x + (1-t)e^{\theta x - \frac{1}{2}\theta^2 t + \theta}$. Recall $X_t = W_t$ is a Brownian motion. In this case, $\mathrm{d}J^\theta(t,X_t) = \mathrm{d}W_t + (1-t)\theta e^{\theta W_t - \frac{1}{2}\theta^2 t + \theta}\mathrm{d}W_t - e^{\theta W_t - \frac{1}{2}\theta^2 t+\theta}\mathrm{d}t$.

If we use the one-step or one-episode CTD(0) algorithm with $\xi_t = \frac{\partial J^\theta}{\partial \theta}(t,X_t) = (1-t)e^{\theta X_t - \frac{1}{2}\theta^2 t + \theta}(X_t - \theta t + 1)$, then the moment condition (22) becomes

$$\begin{aligned}0 =& \mathbb{E}[\int_0^1 (1-t)e^{\theta W_t - \frac{1}{2}\theta^2 t+\theta}(W_t - \theta t + 1)e^{\theta W_t - \frac{1}{2}\theta^2 t + \theta}\mathrm{d}t] \\ =& \int_0^1 (1-t)(1+t\theta)e^{(2+t\theta)\theta}\mathrm{d}t = \frac{e^{2\theta}[2-\theta+\theta^2-\theta^3+e^{\theta^2}(-2+\theta+\theta^2)]}{\theta^5}.\end{aligned}$$

This equation has a unique solution $\theta \approx -1.83923$. A convergent CTD(0) algorithm should converge to this point, which is however *different* from the solution produced by the martingale loss function approach.

If we use the one-step or one-episode CTD(1) algorithm with $\xi_t = \int_0^t \frac{\partial J^\theta}{\partial \theta}(s,X_s)\mathrm{d}s = \int_0^t (1-s)e^{\theta W_s - \frac{1}{2}\theta^2 s + \theta}(W_s - \theta s + 1)\mathrm{d}s$, then the moment condition (22) is

$$\begin{aligned}0 =& \mathbb{E}[\int_0^1 \int_0^t (1-\tau)e^{\theta W_\tau - \frac{1}{2}\theta^2\tau + \theta}(W_\tau - \theta\tau + 1)\mathrm{d}\tau e^{\theta W_t - \frac{1}{2}\theta^2 t + \theta}\mathrm{d}t] \\ =& \int_0^1 \int_0^t \mathbb{E}\big[e^{\theta W_\tau - \frac{1}{2}\theta^2\tau}(W_\tau - \theta\tau + 1)\mathbb{E}[e^{\theta W_t - \frac{1}{2}\theta^2 t}|\mathcal{F}_\tau]\big](1-\tau)e^{2\theta}\mathrm{d}\tau\mathrm{d}t \\ =& \int_0^1 \int_0^t \mathbb{E}[e^{2\theta W_\tau}(W_\tau - \theta\tau + 1)](1-\tau)e^{-\theta^2\tau + 2\theta}\mathrm{d}\tau\mathrm{d}t \\ =& \int_0^1 \int_0^t (1+\tau\theta)(1-\tau)e^{\theta^2\tau + 2\theta}\mathrm{d}\tau\mathrm{d}t \\ =& \frac{e^{2\theta}[6 + 2e^{\theta^2}(-3+\theta+\theta^2) - (-1+\theta)\theta(-2+2\theta+\theta^3)]}{\theta^7}.\end{aligned}$$

There is a unique solution $\theta \approx -2.12568$, to which a convergent CTD(1) algorithm converges. This solution coincides with the one by the martingale loss function approach.

The implementations of the above algorithms are reported in Figure 6, which are consistent with the theoretical analysis.

When the parametric family is a linear span of some basis functions, the unique solution that solves the moment conditions is theoretically guaranteed under very mild conditions, which is numerically generated by the CLSTD algorithm. More generally, all the CGTD methods aim to minimize some quadratic forms of the moment conditions regardless of whether the existence and/or uniqueness of the solution to the conditions holds, and hence usually produce more robust results. Moreover, these methods have a clear geometric interpretation. Recall that the true value function minimizes Bellman's error to zero. The space of approximate linear functions may not contain the true function, but the CGTD algorithms minimize the *projection* of Bellman's error onto the linear space. This intuition is formalized in Sutton et al. (2009) and Maei et al. (2009), who show that the discrete-time GTD minimizers, instead of directly approximating the value function, minimize the so-called *mean-square projected Bellman's error* (MSPBE). Here, we present a continuous-time version of the result with a more general choice of the test functions.

Theorem 7 *For L' linearly independent test functions $\xi^{\theta,(1)}, \cdots, \xi^{\theta,(L')} \in L^2_{\mathcal{F}}([0,T])$, denote by Π_θ the projection operator onto the linear space spanned by $\{\xi^{\theta,(1)}, \cdots, \xi^{\theta,(L')}\}$. Then*

$$\begin{aligned}
&\frac{1}{2}\mathbb{E}\left[\int_0^T \xi_t^\theta \mathrm{d}M_t^\theta\right]^\top \left[\mathbb{E}\int_0^T \xi_t^\theta(\xi_t^\theta)^\top \mathrm{d}t\right]^{-1} \mathbb{E}\left[\int_0^T \xi_t^\theta \mathrm{d}M_t^\theta\right] \\
=&\frac{1}{2}\mathbb{E}\left[\int_0^T \big(\mathcal{L}J^\theta(t,X_t)+r_t\big)\xi_t^\theta \mathrm{d}t\right]^\top \left[\mathbb{E}\int_0^T \xi_t^\theta(\xi_t^\theta)^\top \mathrm{d}t\right]^{-1} \mathbb{E}\left[\int_0^T \big(\mathcal{L}J^\theta(t,X_t)+r_t\big)\xi_t^\theta \mathrm{d}t\right] \\
=&\frac{1}{2}||\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big)||^2_{L^2} =: \mathrm{MSPBE}(\theta).
\end{aligned}$$

Recall Example 5 in which the moment condition admits no solution due to the choice of the test function and hence CTD methods such as CTD(0) will not converge. We now illustrate that CGTD does lead to a solution that is the minimizer of MSPBE.

Example 5 (Continued) Consider the same learning problem in Example 1 with the parameterized value function $J^\theta(t,x) = x + (1-t)e^{\theta x - \frac{1}{2}\theta^2 t}[(\theta+1)^2+1]$. Recall $X_t = W_t$ is a Brownian motion. If we choose the test function to be the constant 1, the projection of $e^{\theta W_t - \frac{1}{2}\theta^2 t}[(\theta+1)^2+1]$ onto the subspace spanned by the test function 1 is $(\theta+1)^2+1$. Theorem 7 yields that CGTD2 or CTDC algorithms should minimize $\mathbb{E}\int_0^1 \left|(\theta+1)^2+1\right|^2 \mathrm{d}t$, whose minimizer is -1. We implement CTD(0) and CGTD2 (along with ML) and show the results in Figure 7. In our implementation, the initial learning rate for all the three algorithms is 0.01 and decays as $(\sharp\text{episode})^{-0.67}$. The results confirm our theoretical analysis.

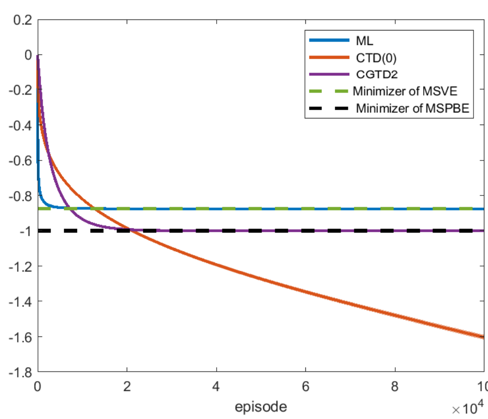

Figure 7: **Minimizing martingale loss function and CGTD methods converge to the respective minimizers, while CTD(0) diverges, for Example 5.** Applying SGD to minimize the martingale loss function leads to $\theta^*_{\mathrm{ML}} \approx -0.875301$, CTD(0) does not converge since there is no solution to the moment condition, and CGTD method leads to $\theta^*_{\mathrm{MSPBE}} = -1$, which is equivalent to minimizing the mean-square projected Bellman's error. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

5. Extensions and Applications

In this section, we extend the martingale characterization to the case with an exponential discount factor, and discuss the non-Markovian setting through an example of a fractional Brownian motion. Then we present two applications – a problem with option-like payoff and a linear-quadratic problem in infinite time horizon.

5.1 Extension to discounted case

In many applications the payoff functions involve discounting. We now extend our analysis for PE to such a case. Note that, in this case, if we let $T \to \infty$, then we will have an infinite time horizon problem.

We modify the value function of (2) to

$$J(t,x) = \mathbb{E}\left[\int_t^T e^{-\rho(s-t)} r(s,X_s)\mathrm{d}s + e^{-\rho(T-t)}h(X_T)\Big|X_t = x\right], \tag{23}$$

where the discount rate $\rho > 0$ is known and given.

The PDE (3) is revised to

$$\begin{cases} \mathcal{L}J(t,x) + r\big(t,x\big) = \rho J(t,x),\ (t,x) \in [0,T) \times \mathbb{R}^d, \\ J(T,x) = h(x), \end{cases} \tag{24}$$

and the FBSDE (4) becomes

$$\begin{cases} \mathrm{d}X_s = b(s,X_s)\mathrm{d}s + \sigma(s,X_s)\mathrm{d}W_s,\ \ s \in [t,T];\ \ X_t = x, \\ \mathrm{d}Y_s = -e^{-\rho(s-t)} r\big(s,X_s\big)\mathrm{d}s + Z_s\mathrm{d}W_s, s \in [t,T];\ Y_T = e^{-\rho(T-t)}h(X_T), \end{cases} \tag{25}$$

whereas the relationship (5) is now

$$Y_s = e^{-\rho(s-t)} J(s,X_s),\ \ Z_s = e^{-\rho(s-t)} \frac{\partial J}{\partial x}(s,X_s)^\top \sigma\big(s,X_s\big),\ \ s \in [t,T]. \tag{26}$$

Finally

$$M_s := e^{-\rho(s-t)} J(s,X_s) + \int_t^s e^{-\rho(s'-t)} r\big(s',X_{s'}\big)\mathrm{d}s' \equiv Y_s + \int_t^s e^{-\rho(s'-t)} r\big(s',X_{s'}\big)\mathrm{d}s',\ \ s \in [t,T] \tag{27}$$

is a martingale.

Fixing the initial time $t = 0$, the above analysis suggests that the martingale loss function should be

$$\begin{aligned} \mathbb{E}\int_0^T |M_T - M_t^\theta|^2 \mathrm{d}t =& \mathbb{E}\int_0^T \left| e^{-\rho T}h(X_T) - e^{-\rho t} J^\theta(t,X_t) + \int_t^T e^{-\rho s} r\big(s,X_s\big)\mathrm{d}s \right|^2 \mathrm{d}t \\ \approx& \mathbb{E}\Bigg[\sum_{i=0}^{K-1}\left(e^{-\rho T}h(X_T) - e^{-\rho t_i}J^\theta(t_i,X_{t_i}) + \sum_{j=i}^{K-1} e^{-\rho t_j} r(t_j,X_{t_j})\Delta t\right)^2 \Delta t\Bigg]. \end{aligned}$$

On the other hand, $\mathrm{d}M_t^\theta = e^{-\rho t}\mathrm{d}J_t^\theta - \rho e^{-\rho t} J_t^\theta \mathrm{d}t + e^{-\rho t} r_t \mathrm{d}t = e^{-\rho t}(\mathrm{d}J_t^\theta + r_t\mathrm{d}t - \rho J_t^\theta \mathrm{d}t)$; hence the martingale orthogonality condition (22) is modified to

$$\mathbb{E}\int_0^T \xi_t(\mathrm{d}J_t^\theta + r_t\mathrm{d}t - \rho J_t^\theta\mathrm{d}t) = 0, \tag{28}$$

for any test function $\xi \in L^2_{\mathcal{F}}([0,T])$. Note here the discount factor has been absorbed by the test function and thus omitted.

If we set $T = \infty$ in (23) and assume the payoff does not depend on t, then the problem becomes

$$J(x) = \mathbb{E}\left[\int_0^\infty e^{-\rho t} r(X_t)\mathrm{d}t | X_0 = x\right].$$

This type of problems occur when the time horizon is sufficiently long or indefinite. Note that in this case the value function does not depend on time explicitly. As a result, there is no longer a terminal condition; instead, it is usually replaced by a growth condition such as $\mathbb{E}[e^{-\rho t} J(X_t)] \to 0$ as $t \to \infty$.

As the martingale loss function requires full trajectories, it may not be directly applicable for the infinite-horizon problem where we are obviously not able to observe a whole sample until "infinity". However, the martingale loss function can still be defined by truncating at a sufficiently long time T with an artificial terminal condition $e^{-\rho T} h(X_T) = 0$. Therefore, in the episodic setup with repeatedly presented finite training sets, we can still learn the value function by minimizing the martingale loss function. However, with a very long horizon, people are more interested in online learning with no reset by observing a single trajectory. As a result, TD learning is a better choice. All the previously discussed TD learning methods with suitable test functions can be applied based on the conditions (28).

5.2 Extension to non-Markovian setting

A key assumption of the current paper is that the state process X is Markovian. Indeed, the Markov property determines that the value function J, defined through (2), is a function of the current state x, instead of the whole past history of X. However, the martingale perspective may extend beyond the Markovian setting. While a general non-Markovian PE theory goes beyond the scope of this paper and warrants a full separate research, here we use an example to illustrate.

Recall in Example 1, the state process $X = W$ is a Brownian motion, which is Markovnian. Now we consider instead a *fractional* Brownian motion W^H with the Hurst index H. When $H = \frac{1}{2}$, W^H reduces to a Brownian motion; but when $H \neq \frac{1}{2}$, it is well known to be a non-Markov process. For basic theory and applications of fractional Brownian motions, see e.g. Mandelbrot and Van Ness (1968).

For a non-Markov process, the value function is a functional of the current time t and the entire state trajectory up to t. For example,

$$\mathbb{E}[W_T^H | \mathcal{F}_t] = W_t^H - \int_0^t \Psi_H(T, s|t)\mathrm{d}W_s^H,$$

where

$$\Psi_H(T, s|t) = -\frac{\sin\left(\pi(H - \frac{1}{2})\right)}{\pi} s^{\frac{1}{2}-H}(t-s)^{\frac{1}{2}-H} \int_t^T \frac{z^{H-\frac{1}{2}}(z-t)^{H-\frac{1}{2}}}{z-s}\mathrm{d}z;$$

see Sottinen and Viitasaari (2017).

For Example 1 with $X_t = W_t$ replaced by $X_t = W_t^H$, the only modification we need is to introduce the value function $J(t, X_{t\wedge\cdot})$ that is a functional of the past trajectory with the terminal condition $J(1, X_{1\wedge\cdot}) = \mathbb{E}[X_1|X_{1\wedge\cdot}] = X_1$. In our experiment, we use a two-layer fully connected neural network plus an LSTM type of recurrent neural network to approximate such a path-dependent functional satisfying the terminal condition:

$$J^\theta(t, X_{t\wedge\cdot}) = X_t + (1-t)\,\mathrm{NN}^\theta\left(\mathrm{LSTM}^\theta(t, X_{t\wedge\cdot})\right),$$

and then minimize the martingale loss function or apply the CTD(0) algorithm to update the parameters. Here LSTM^θ maps a sequence of time-series data $(X_{t_0}, \cdots, X_{t_k}, \cdots, X_{t_K})$ to a sequence of output $(Y_{t_0}, \cdots, Y_{t_k}, \cdots, X_{t_K})$ recursively where the time-t_k output Y_{t_k} depends on the past trajectory $(X_{t_0}, \cdots, X_{t_k})$. For details of this network structure, see Hochreiter and Schmidhuber (1997).

We then compare the mean-square value errors between the learned functional and the ground truth solution with both algorithms. Figure 8 shows the trend of convergence of both the ML and CTD(0) methods, although the convergence is not as sharp as in the other Markov examples (most likely due to the non-Markovian setting and a fairly general neural network used). We also see that CTD(0) performs better than ML in this case.

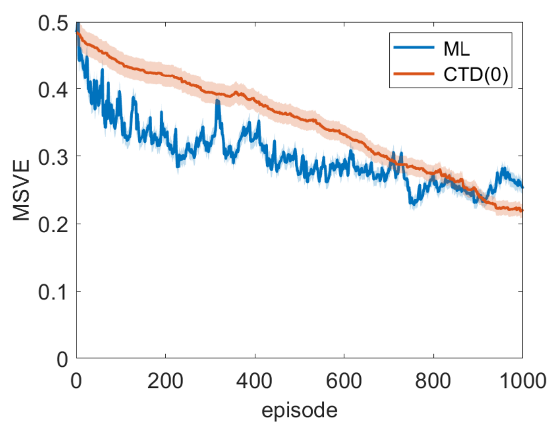

Figure 8: **The mean-square value errors of the learned value functional with ML and CTD(0) algorithm for a fractional Brownian motion with Hurst index** $H = 0.75$. The MSVEs are evaluated with 1000 independent trajectories and standard deviations are computed, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

5.3 Option-like payoff

We apply the theory developed so far to evaluate

$$J(t, x) = \mathbb{E}[e^{-r(T-t)}h(X_T)|X_t = x], \tag{29}$$

where $X = \{X_t,\ 0 \le t \le T\}$ is the state process. This type of evaluation occurs in option pricing in which X is the underlying stock price process and h is the payoff function (usually known and given) at the maturity T.

In our simulation, we generate X from a geometric Brownian motion

$$\frac{\mathrm{d}X_t}{X_t} = (r-q)\mathrm{d}t + \sigma \mathrm{d}W_t,$$

and take h as a call option payoff

$$h(x) = (x-K)^+.$$

Moreover, we set $T = 1, K = 1,\ r = 0.01,\ q = 0,\ \sigma = 0.3$. The price process is generated from $X_0 = 1$.

The value function has a theoretical (ground truth) form given by the Black–Scholes formula

$$J(t,x) = e^{-r(T-t)}[e^{(r-q)(T-t)}x\Phi(d_+) - K\Phi(d_-)],$$

where

$$d_\pm = \frac{\log(x/K) + (r - q \pm \frac{1}{2}\sigma^2)(T-t)}{\sigma\sqrt{T-t}},$$

and Φ is the distribution function of the standard normal.

For learning the value function in our numerical experiment, we parameterize it by $J^\theta(t,x) = (x-K)^+ + (T-t)\,\mathrm{NN}^\theta(t,x)$, where NN^θ is a general bivariate neural network taking both time and space as inputs. This particular form is inspired by that of the payoff function along with the fact that time to maturity, $T-t$, is critical in pricing an option. We use both ML and online CTD(0) in our experiment, with the martingale loss function being

$$\mathbb{E}\int_0^T \left|e^{-rT}(X_T - K)^+ - e^{-rt}J^\theta(t,X_t)\right|^2 \mathrm{d}t.$$

In our implementation, we use a simple three-layer fully connected neural network with softplus activation function and 128 and 64 neurons, that is,

$$\mathrm{NN}^\theta(u) = \theta_5 a\Big(\theta_3 a\big(\theta_1 u + \theta_2\big) + \theta_4\Big) + \theta_6,\ \ a(x) = \log(1+e^x),$$

where $\theta_1 \in \mathbb{R}^{128\times 2}, \theta_2 \in \mathbb{R}^{128\times 1},\ \theta_3 \in \mathbb{R}^{64\times 128}, \theta_4 \in \mathbb{R}^{64\times 1},\ \theta_5 \in \mathbb{R}^{64\times 1}, \theta_6 \in \mathbb{R}$.

Since now we have hundreds of parameters and the functional forms are complex, instead of comparing the learned parameters, we assess the performance of learning by the following three errors:

$$J(0,x_0) - J^\theta(0,x_0),\ \mathbb{E}\int_0^T \left|J(t,X_t) - J^\theta(t,X_t)\right|^2 \mathrm{d}t,\ \mathbb{E}\int_0^T \Big|\frac{\partial J}{\partial x}(t,X_t) - \frac{\partial J^\theta}{\partial x}(t,X_t)\Big|^2 \mathrm{d}t,$$

where θ is the vector of optimized parameters obtained, and J is the ground truth value function. In these errors, the first one is in terms of price difference at the initial time $t=0$, and the second one in terms of the averaged total price differences over time. The last one concerns the accuracy in determining $\frac{\partial J}{\partial x}$, the so-called "Delta" of the option which is the quantity of the underlying stock needed to hedge the option risk.

The PDE (24) satisfied by J is nothing else than the well-known Black-Scholes PDE:

$$\frac{\partial J}{\partial t} + (r-q)x\frac{\partial J}{\partial x} + \frac{1}{2}\sigma^2 x^2 \frac{\partial^2 J}{\partial x^2} - rJ = 0,\ \ J(T,x) = (x-K)^+.$$

Hence, as discussed earlier, PE can also be regarded as an alternative method to solve such a PDE. This in turn presents a benchmark in our experiment for comparison purpose, which is the deep learning method in Han et al. (2018) called the *deep BSDE method* for solving PDEs. Note that their method requires the perfect knowledge about the model parameters or, equivalently, not only samples of $\{X_t, 0 \leq t \leq T\}$ but also samples of $\{W_t, 0 \leq t \leq T\}$ that drives the state process. When implementing the deep BSDE method, we apply a neural network with the same structure to keep the computational cost at the similar level. We use the Adam algorithm for optimization with one trajectory for each episode so that the number of training sample trajectories is also kept the same.[17]

Figure 9 shows the comparison. For the first two criteria, the errors by the two PE methods developed in this paper, (offline) ML and online CTD(0), both converge to zero very quickly, while it takes some time for those with the BSDE method to be close to zero. For the last criterion, the errors by the PE methods remain close to zero and keeps almost flat from the start, while the BSDE method oscillates dramatically at the start before converging to 0. Indeed, we have shown that minimizing the martingale loss function is equivalent to approximating the value function itself in the mean–square sense, without concerning at all the derivatives of the function. In contrast, the deep BSDE method strives also to learn the derivative term (the Z_t term in FBSDE (25)) directly and hence requires more knowledge about the system. This example shows that PE methods can be used to learn the function itself effectively but may not provide an accurate approximation to the derivative value. In particular, in terms of estimating the value function, ML achieves the smallest error and CTD(0) is only slightly behind due to its online setting; but deep BSDE can ultimately learns the derivative. As such, PE methods provide more flexibility for users with tasks such as solving a PDE. It also suggests that for continuous-time RL one should avoid methods relying on the derivatives of the estimated value function.

Finally, we point out that the purpose of this example is to compare our methods with the deep PDE/BSDE method. Because (29) holds in the risk-neutral world where data cannot be actually observed, our method cannot be used directly to evaluate an option price. It remains an interesting problem to price options based on physical probability and the real-world data.

5.4 Infinite time horizon linear-quadratic problem

Consider the following value function

$$J(x) = \mathbb{E}\left[\int_0^\infty e^{-\rho t} r(X_t)\mathrm{d}t | X_0 = x\right],$$

where $X = \{X_t,\ 0 \leq t < \infty\}$ is the state process. In our simulation, we generate X from an OrnsteinUhlenbeck (OU) process

$$\mathrm{d}X_t = a(b - X_t)\mathrm{d}t + \sigma \mathrm{d}W_t,$$

17. The Adam algorithm is proposed in Kingma and Ba (2014) and is considered to be an improvement over the vanilla SGD algorithm. We follow Han et al. (2018) to apply the Adam algorithm. Implementation of neutral networks is through Tensorflow 2. All the computations are conducted on an Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz 1.50 GHz Windows laptop.

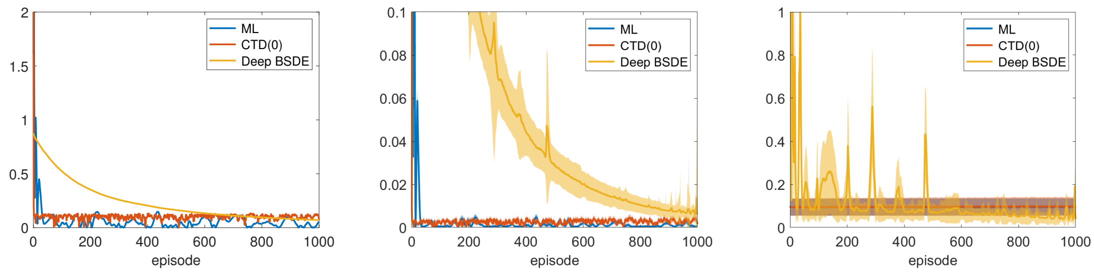

Figure 9: **Comparison of learned value functions by ML, online CTD(0) and deep BSDE methods.** From left to right, we show the errors against the true solutions in terms of $|J(0,x_0)-J^\theta(0,x_0)|$, $\mathbb{E}\int_0^T \left|J(t,X_t)-J^\theta(t,X_t)\right|^2 \mathrm{d}t$, $\mathbb{E}\int_0^T \left|\frac{\partial J}{\partial x}(t,X_t)-\frac{\partial J^\theta}{\partial x}(t,X_t)\right|^2 \mathrm{d}t$ respectively. These expectations are evaluated using 5000 independent trajectories. Standard deviations are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

which is well-known to converge to its unique stationary distribution when $a>0$ and $\sigma>0$. And we take r to be a quadratic function:

$$r(x)=\frac{1}{2}x^2+qx.$$

This is a discounted linear-quadratic (LQ) control problem in infinite time horizon.

By the standard stochastic control theory via dynamic programming (Yong and Zhou, 1999, Chapter 6) we can compute the value function explicitly as $J(x)=\frac{1}{2}Ax^2+Bx+C$, where

$$A=\frac{1}{\rho+2a},\ B=\frac{abA+q}{\rho+a},\ C=\frac{abB+\frac{1}{2}\sigma^2A}{\rho}.$$

We set $a=1,\ b=1,\ \sigma=0.5,\ \rho=1.5,\ q=1,\ X_0=0$, and simulate the trajectories up to $T=2\times10^5$. In our experiment, we parameterize the value function by $J^\theta(x)=\frac{1}{2}\theta_0x^2+\theta_1x+\theta_2$. We implement CTD(0), CLSTD(0), and CGTD2 algorithms and report the results in Figure 10. Since the parametric functions lie in a linear space, CLSTD(0) explicitly solves the moment conditions, and hence converges fastest. CGTD2 converges faster than CTD(0) because the former is a true gradient-based algorithm.

6. Some Algorithmic Aspects

In this section we discuss two problems from the algorithmic perspective: the choice of test functions and the way to perform function approximation.

6.1 Choice of test functions

One of the most important implications of the martingale viewpoint is the introduction of the test functions. In Subsection 4.2, we show that the choice of test functions determines in what sense the true value function is approximated and, hence, a same algorithm with different test functions may converge differently, as illustrated in Examples 3 – 5. While this characterization remains abstract in theory and provides little guidance on how to actually

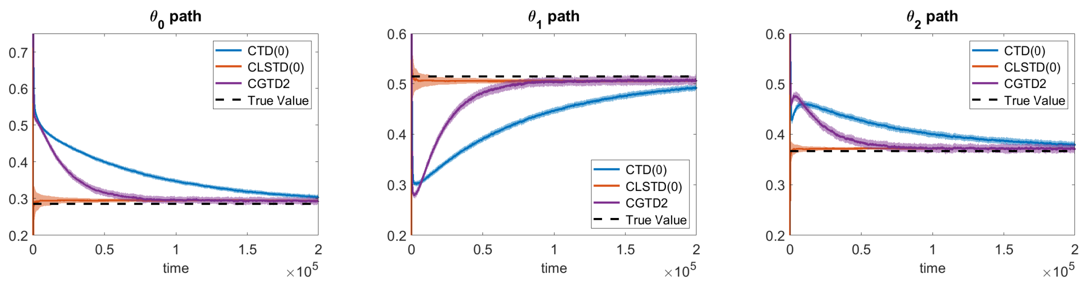

Figure 10: **Comparison of learned parameters with different online TD algorithms.** All the algorithms converge to the correct value function. Among them, CLSTD(0) converges the fastest and CTD(0) the slowest. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

select test functions, here we present a simple example to demonstrate how test functions may affect the convergence from an algorithmic perspective.

Consider the same LQ problem in Subsection 5.4, with $a = 0$, $\sigma = 1$, and $q = 0$. In this case, $X_t = X_0 + W_t$ now becomes a Brownian motion, which has no stationary distributions.[18]

As before, we parameterize the value function by $J^\theta = \frac{1}{2\rho}x^2 + \theta$. This parametric family contains the true value function with $\theta_{\text{true}} = \frac{1}{2\rho^2}$. The conventional choice of the test function in TD(0) is $\xi_t = \frac{\partial J^\theta}{\partial \theta}(X_t) = 1$, leading to the following updating rule on θ:

$$\theta \leftarrow \theta + \alpha[J^\theta(X_{t+\Delta t}) - J^\theta(X_t) + r(X_t)\Delta t - \rho J^\theta(X_t)\Delta t].$$

Denote by θ_t the learned parameter value at time t. Then, at the continuous-time limit, θ_t satisfies an SDE (ignoring the learning rate constant α):

$$\mathrm{d}\theta_t = \mathrm{d}J^{\theta_t}(X_t) + r(X_t)\mathrm{d}t - \rho J^{\theta_t}(X_t)\mathrm{d}t.$$

By Itô's lemma, $\mathrm{d}J^{\theta_t}(X_t) = \frac{X_t}{\rho}\mathrm{d}W_t + \frac{1}{2\rho}\mathrm{d}t$; hence

$$\mathrm{d}\theta_t = \left(\frac{1}{2\rho} - \rho\theta_t\right)\mathrm{d}t + \frac{X_t}{\rho}\mathrm{d}W_t.$$

Suppose the initial guess of θ is θ_0. Then

$$\mathbb{E}[\theta_t] = \frac{1}{2\rho^2}(2\theta_0\rho^2 e^{-\rho t} + 1 - e^{-\rho t}) \to \frac{1}{2\rho^2} = \theta_{true} \quad \text{as } t \to \infty.$$

That is, asymptotically, the conventional choice of the test function indeed leads to an unbiased estimate. Let us now calculate $\mathrm{Var}(\theta_t)$, the variance of θ_t. Set $z_t = \theta_t - \frac{1}{2\rho^2}(2\theta_0\rho^2 e^{-\rho t} + 1 - e^{-\rho t})$, which satisfies the SDE:

$$\mathrm{d}z_t = -\rho z_t \mathrm{d}t + \frac{X_t}{\rho}\mathrm{d}W_t, \ z_0 = 0.$$

18. Such non-stationary processes are common in practice. Due to the presence of the discount factor, the corresponding LQ problem is still well-posed.

Itô's lemma provides

$$\mathrm{d}z_t^2 = 2z_t[-\rho z_t \mathrm{d}t + \frac{X_t}{\rho}\mathrm{d}W_t] + \frac{X_t^2}{\rho^2}\mathrm{d}t, \ z_t = 0.$$

Hence

$$\mathrm{Var}(\theta_t) = \mathbb{E}[z_t^2] = \frac{1}{4\rho^4}\left(e^{-2\rho t} - 1 + \rho t - 2\rho X_0^2 e^{-2\rho t} + 2\rho X_0^2\right) \to \infty, \text{ as } t \to \infty.$$

So, the conventionally chosen test function does not produce a consistent estimator of θ due to the blow-up in variance, which in turn is caused by the non-stationarity of the underlying state process – a Brownian motion in this example – whose variance grows linearly in time.

However, this issue can be resolved by selecting a *tailored* test function. Recall the CTD(0) algorithm with a general test function ξ_t updates θ by

$$\theta \leftarrow \theta + \alpha \xi_t \left[J^\theta(X_{t+\Delta t}) - J^\theta(X_t) + r(X_t)\Delta t - \rho J^\theta(X_t)\Delta t\right].$$

Applying the same SDE approximation, we derive

$$\mathrm{d}\theta_t = \xi_t\left(\frac{1}{2\rho} - \rho\theta_t\right)\mathrm{d}t + \xi_t \frac{X_t}{\rho}\mathrm{d}W_t.$$

Intuitively, to reduce the variance of θ_t, we need to choose a test function that can cancel the growing trend in variance. There are many choices to achieve this goal, but a simple one is to take $\xi_t = \frac{1}{|X_t|+1}$ so that the volatility term above is bounded. We call this a "tailored choice" of test function for this particular LQ problem. The cost of this variance reduction method is the introduction of some bias in the mean as some correlation enters into the drift term.

Figure 11 visualizes the result of a simulation study that confirms our analysis. With the conventional test function $\xi_t = 1$, even though the average of the learned parameter values across different experiments tends to be close to the true value, these values become more volatile as time grows larger. On the other hand, with our tailored test function $\xi_t = \frac{1}{|X_t|+1}$, the variance is reduced dramatically, though the average is slightly off from the true value.

Overall, the study we provide in this subsection shows the promise of our martingale framework in designing more efficient algorithms with suitable choice of test functions, which may at the same time extend the existing literature on RL algorithms for MDPs.

6.2 Function approximation: global vs sectional

For a finite-horizon problem, the value function $J(t,x)$ is a *bivariate* function of time t and state x. Hitherto we have used a *global* approximator J^θ in the sense that we use the same parameter θ when approximating $J(t,\cdot)$ by $J^\theta(t,\cdot)$. Another way of function approximation is *sectional*, namely, we approximate $J(t,\cdot)$ by $J^{\theta_t}(t,\cdot)$ where the parameter θ_t may be time-varying. More precisely, let the time discretization be fixed with the grid points $0 = t_0 < t_1 < \cdots < t_K = T$, and let $J_0^{\theta_0}(x), \cdots, J_K^{\theta_K}(x)$ approximate the value function at these points, namely, $J_i^{\theta_i}(x) \approx J(t_i, x)$, $i = 0, 1, \cdots, K$.

To compare these two methods of function approximation, the first thing to note is that the number of parameters to learn grows linearly in the number of time steps with the

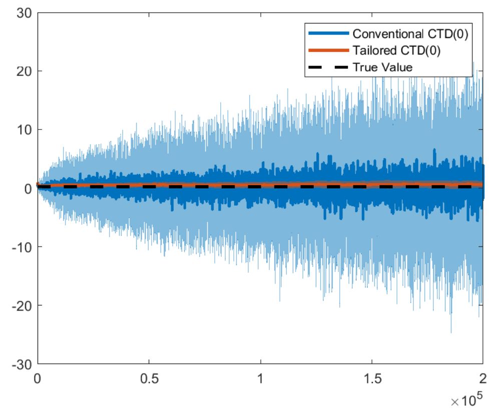

Figure 11: **Comparison of the learned parameters under the conventional test function and the tailored test function**. Conventional CTD(0) refers to the algorithm using test function $\xi_t = \frac{\partial J^\theta}{\partial \theta}(X_t) = 1$, and Tailored CTD(0) refers to the one using test function $\xi_t = \frac{1}{|X_t|+1}$. In the simulation, the problem primitives are $a = 0$, $\sigma = 1$, $q = 0$, $\rho = 1.5$, the initial state is $X_0 = 0$ and initial guess of the parameter θ is $\theta_0 = 1$. The true parameter is $\theta_{\text{true}} = \frac{2}{9} \approx 0.22$. The learning rate is $\alpha = 0.1$. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

sectional approximation, while remains the same with the global one. Hence, the latter has an edge in terms of computational cost when a finer time grid is used. Second, and indeed more importantly, the sectional approximation may become problematic for online learning. To see this, suppose we are now at (t_i, X_{t_i}) in the online setting. Applying the idea of the conventional TD(0) algorithm, one can update the parameter θ_{i-1} by

$$\theta_{i-1} \leftarrow \theta_{i-1} + \alpha \frac{\partial J_{i-1}^{\theta_{i-1}}}{\partial \theta_{i-1}}(X_{t_{i-1}}) \left[J_i^{\theta_i}(X_{t_i}) - J_{i-1}^{\theta_{i-1}}(X_{t_{i-1}}) \right].$$

The question is how to update θ_k for $k = i, i+1, \cdots, K$ without knowing the *future* states X_{t_k}? It seems the best we could do is to update θ_k according to

$$\theta_k \leftarrow \theta_k + \alpha \frac{\partial J_k^{\theta_k}}{\partial \theta_k}(X_{t_{i-1}}) \left[J_i^{\theta_i}(X_{t_i}) - J_{i-1}^{\theta_{i-1}}(X_{t_{i-1}}) \right]. \tag{30}$$

This form is less intuitive because we use the current and past states to update parameters for future value functions. In contrast, the global parameterization views the value function as a whole; hence a temporal advancement naturally leads to an update of the whole bivariate function, including a prediction into the future as well as an updated evaluation of the past.

Finally, we run a simulation for Example 1 to compare the learning results of the two function approximation approaches, both in offline learning (using martingale loss function) and online learning (using CTD(0) for the global approximation and (30) for the sectional one). Recall that the ground truth is $J(t, x) = x$, and we have used the global approximation with $J^\theta(t,x) = [\theta(1-t)+1]x$. For the sectional approximation, we consider a simple form of $J_i^{\theta_i}(x) = \theta_i x$, with unknown parameters $\theta_0, \cdots, \theta_{K-1}$ while it is known that $\theta_K = 1$ based on the terminal condition.

We evaluate the performance of the different approximation approaches by MSVE as defined in (21). For the global approximation, this error is

$$\mathbb{E}\int_0^T |J(t,X_t) - J^\theta(t,X_t)|^2 \mathrm{d}t = \mathbb{E}\int_0^T \theta^2(1-t)^2 W_t^2 \mathrm{d}t = \frac{1}{12}\theta^2.$$

For the sectional approximation, this error is calculated by

$$\mathbb{E}\sum_{i=0}^{K-1} |J(t_i, X_{t_i}) - J_i^{\theta_i}(X_{t_i})|^2 \Delta t = \sum_{i=0}^{K-1} (\theta_i - 1)^2 t_i \Delta t.$$

The results are presented in Figure 12. For this simple example, the number of unknown parameters in the sectional approach is small so the difference in computational cost is insignificant. Otherwise, we observe that the global approximation performs similarly as the sectional one in the offline setting (the ML method), but significantly better in the online setting (the CTD(0) method).

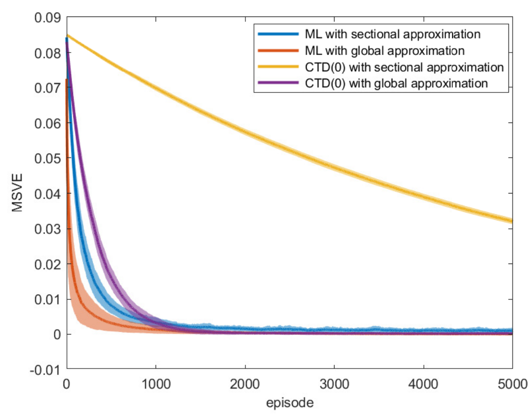

Figure 12: **Comparison of the mean-square value errors of the learned value function using globel and sectional approximation methods with ML and CTD(0) algorithms**. The initial guess is $\theta = -1$ for global approximation and $\theta_i = t_i$, $0 \le i \le K-1$, for sectional approximation so that the two methods are initialized to be the same function. The learning rate is $\alpha = 0.01$. We repeat the experiment for 100 times to calculate the standard deviations, which are represented as the shaded areas. The width of each shaded area is twice the corresponding standard deviation.

7. Conclusions

In this paper, we provide a unified theoretical framework for studying PE in RL with continuous time and space. The theory is premised upon the observation that PE is equivalent to enforcing the martingality of a stochastic process. Many existing and popular PE algorithms, which somewhat scatter around in the MDP literature, can find a common ground through this "martingale lens". These algorithms can be modified for solving PE in the continuous setting for actual implementation.

The martingale perspective is potentially useful for studying many important problems related to PE that have been well developed in the discrete setting but remain open in the continuous setting, including off-policy evaluation, state-action value estimation, and convergence analysis. Furthermore, it may inspire new questions that have not been posed by traditional RL research. For example, how to "optimally" choose test functions and how their choices affect the convergence rate in both discrete and continuous settings.

Finally, PE is formulated for Itô processes in this paper, mainly because such a process has convenient and well-studied properties and can reasonably model many real-life dynamics. The martingale view, however, is generalizable beyond Itô processes such as jump diffusions, non-Markov processes and semi-martingales.

Acknowledgments

We are grateful for comments from the seminar participants at University of Southern California, Boston University, Imperial College, University of Connecticut, the International Seminar on SDEs and Related Topics, the Joint Seminar by AIFT and Columbia University and the World Online Seminars on Machine Learning in Finance, and from the participants at the 6th Berlin Workshop for Young Researchers in Mathematical Finance and the IMSI Workshop on Advances in Optimal Decision Making under Uncertainty. We thank Jerome Detemple, Steven Kou, Huyên Pham, Moris Strub, Wenpin Tang, Nizar Touzi, Renyuan Xu and Jianfeng Zhang for helpful discussions and comments on the paper. We are also indebted to the Action Editor and three anonymous referees for constructive comments which have led to an improved version of the paper. Zhou gratefully acknowledges financial support through a start-up grant and the Nie Center for Intelligent Asset Management at Columbia University.

Appendix A: A Summary of Popular PE Methods

The following Table 1 summarizes popular PE methods and algorithms, and the interpretations we have discovered in this paper in terms of objectives (loss/error functions to be minimized or equations to be solved) and limiting points of convergent algorithms.

Method	Representative algorithms	Online	Objective	Converging point
Monte Carlo[a]	gradient Monte Carlo	No	minimize martingale loss function	minimizers of mean-square value function error
Residual gradient[b]	naïve residual gradient	Yes	minimize mean-square TD error	minimizers of quadratic variation
Semi-gradient TD learning[c]	TD(λ) LSTD(λ)	Yes	solve moment conditions	zeros to moment conditions
Gradient TD learning[d]	GTD(0) GTD2 TDC	Yes	minimize quadratic form of moment conditions	minimizers of mean-square projected Bellman error

a. Sutton and Barto (2018).
b. Baird (1995).
c. Sutton (1988); Bradtke and Barto (1996). This terminology is taken from Sutton and Barto (2018, Chapter 9).
d. Sutton et al. (2008, 2009).

Table 1: **Summary of popular PE methods in RL literature.** The table summarizes different PE methods. The first three columns indicate the names of the methods, those of the representative algorithms, and whether applicable online and/or offline. The last two columns reveal the objectives and the converging points of the corresponding algorithms.

Appendix B: Stochastic Control Formulation of RL

Let d, m, n be given positive integers, $T > 0$, and $b : [0,T] \times \mathbb{R}^d \times \mathbb{R}^n \mapsto \mathbb{R}^d$ and $\sigma : [0,T] \times \mathbb{R}^d \times \mathbb{R}^n \mapsto \mathbb{R}^{d\times m}$ be given functions. A stochastic control problem is to control the *state* (or *feature*) dynamic governed by an SDE:

$$\mathrm{d}X_s = b\big(s, X_s, \boldsymbol{u}(s, X_s)\big)\mathrm{d}t + \sigma\big(s, X_s, \boldsymbol{u}(s, X_s)\big)\mathrm{d}W_s,\ s \in [0,T], \tag{31}$$

where $\boldsymbol{u} : (t,x) \in [0,T] \times \mathbb{R}^d \mapsto \boldsymbol{u}(t,x) \in \mathcal{U}$ is a given (measurable) feedback control policy, with $\mathcal{U} \subseteq \mathbb{R}^n$ being the *action space* representing the constraints on an agent's decisions (*controls* or *actions*).

Given a policy $\boldsymbol{u}$ and an initial time–state pair $(t,x) \in [0,T]\times\mathbb{R}^d$, let $\{X_s^{t,x,\boldsymbol{u}}, t \le s \le T\}$ be the solution to (31) with $X_t = x$. The *value function* under the policy $\boldsymbol{u}$ is

$$J(t,x;\boldsymbol{u}) = \mathbb{E}\left[\int_t^T r(s, X_s^{t,x,\boldsymbol{u}}, \boldsymbol{u}(s, X_s^{t,x,\boldsymbol{u}}))\mathrm{d}s + h(X_T^{t,x,\boldsymbol{u}})\Big| X_t^{t,x,\boldsymbol{u}} = x\right], \tag{32}$$

where $r : [0,T] \times \mathbb{R}^d \times \mathbb{R}^n \mapsto \mathbb{R}$ and $h : \mathbb{R}^d \mapsto \mathbb{R}$ are given reward functions.

A policy $\boldsymbol{u}$ is called admissible if (31) has a unique weak solution and (32) is finite for any $(t,x) \in [0,T] \times \mathbb{R}^d$. A typical RL problem is to maximize (minimize) $J(t,x;\boldsymbol{u})$ over all admissible policies $\boldsymbol{u}$. In the classical (model-based) stochastic control literature, the functional forms of b, σ, r and h are known, and there are well-developed theories to solve the problem; see, e.g., Yong and Zhou (1999); Fleming and Soner (2006). However, in the RL context, these functional forms are typically unknown, although in some applications that of h may be known because it may be interpreted as a given target the agent specifies (e.g. in option pricing h is the payoff function of an option, which is typically given and known; see Subsection 5.3).

The PE task as a part of the general RL problem is, for a given policy $\boldsymbol{u}$, to devise a numerical procedure to find $J(t,x;\boldsymbol{u})$ as a function of (t,x) using multiple sample trajectories of the process $\{s, X_s^{t,x,\boldsymbol{u}}, r\big(s, X_s^{t,x,\boldsymbol{u}}, \boldsymbol{u}(t, X_s^{t,x,\boldsymbol{u}})\big)\}_{t\le s\le T}$, without the knowledge of the model parameters (the functional forms of b, σ, r, h).

If we suppress $\boldsymbol{u}$, which is fixed in PE, then we recover the formulation (1)–(2). Note that the formulation also covers the "exploratory" setting of Wang et al. (2020) in which the admissible control policies are probability-distribution-valued, because the value function therein is of the same form as (32) under a fixed distributional control policy.

Appendix C: Martingale in Discrete-time Markov Reward Processes

We show that there is also a martingale property in the classical discrete-time RL MDP formulation. To be consistent with the main setting of this paper, we consider only the finite horizon episodic tasks; the infinite horizon continuing tasks can be studied similarly.

Let $X = \{X_t, t = 0, 1, \cdots, T\}$ be a discrete-time Markov process adapted to $\{\mathcal{F}_t\}_{t=0,1,\cdots,T}$ in a filtered probability space $(\Omega, \mathbb{P}, \mathcal{F}, \{\mathcal{F}_t\}_{t=0,1,\cdots,T})$. One is interested in finding the value function v defined by

$$v(t,x) = \mathbb{E}\left[\sum_{s=t}^{T-1} r(s, X_s) + h(X_T)\Big| X_t = x\right],$$

where $r(t,x)$ is the expected reward at time t conditioned on being at state x, and h is the final reward.

When the state space is finite and discrete, X is referred to as a Markov reward process (MRP) or alternatively as an MDP with a fixed policy. When the state space is infinite or typically continuous, it is usually called a semi-MRP or a semi-MDP.

Set $M_t = v(t, X_t) + \sum_{s=0}^{t-1} r(s, X_s)$ with $M_T = h(X_T) + \sum_{s=0}^{T-1} r(s, X_s)$. Then, for any $t = 0, 1, \cdots, T-1$, by Markov property, we obtain

$$
\begin{aligned}
\mathbb{E}[M_{t+1}|\mathcal{F}_t] &= \sum_{s=0}^{t} r(s, X_s) + \mathbb{E}[v(t+1, X_{t+1})|\mathcal{F}_t] \\
&= \sum_{s=0}^{t-1} r(s, X_s) + \mathbb{E}[r(t, X_t) + v(t+1, X_{t+1})|\mathcal{F}_t] \\
&= \sum_{s=0}^{t-1} r(s, X_s) + v(t, X_t) = M_t,
\end{aligned}
$$

where the last equality is due to

$$
\mathbb{E}[r(t, X_t) + v(t+1, X_{t+1})|\mathcal{F}_t] = \mathbb{E}[r(t, X_t) + v(t+1, X_{t+1})|X_t] = v(t, X_t),
$$

which is the well-known Bellman equation for a discrete-time MRP.

So, M being a martingale is equivalent to the value function satisfying the Bellman equation, which in turn can be used to characterize PE. From this martingale perspective, we can develop parallel approaches such as the martingale loss function and the martingale orthogonality condition that will recover various conventional PE algorithms for discete-time MRPs.

Appendix D: Proofs of Statements

Proof of Proposition 1

Proof To show M is a martingale, observe that based on (2), we have

$$
M_s = \mathbb{E}\left[\int_s^T r\big(s', X_{s'}\big)\mathrm{d}s' + h(X_T)|X_s\right] + \int_t^s r\big(s', X_{s'}\big)\mathrm{d}s' = \mathbb{E}[M_T|\mathcal{F}_s],
$$

where we have used the Markov property of the process $\{X_s, t \le s \le T\}$. This establishes that M is a martingale.

Conversely, if $\tilde{M}$ is a martingale, then $\tilde{M}_s = \mathbb{E}[\tilde{M}_T|\mathcal{F}_s]$, which is equivalent to

$$
\begin{aligned}
\tilde{J}(s, X_s) = &\ \mathbb{E}\left[\int_s^T r\big(s', X_{s'}\big)\mathrm{d}s' + \tilde{J}(T, X_T)|\mathcal{F}_s\right] \\
= &\ \mathbb{E}\left[\int_s^T r\big(s', X_{s'}\big)\mathrm{d}s' + h(X_T)|\mathcal{F}_s\right] \\
= &\ J(s, X_s), \quad s \in [t, T].
\end{aligned}
$$

Letting $s = t$, we conclude $\tilde{J}(t, x) = J(t, x)$.

■

Proof of Theorem 2

We first present two lemmas that will be useful for the proof of Theorem 2 and also other theorems later.

Lemma 8 *Let $f_h(x) = f(x) + r_h(x)$, where f is a continuous function and r_h converges to 0 uniformly on any compact set as $h \to 0$.*

(a) Suppose $x_h^* \in \arg\min_x f_h(x) \neq \emptyset$ and $\lim_{h\to 0} x_h^* = x^*$. Then $x^* \in \arg\min_x f(x)$. Moreover, if there exists $\alpha > 0$ such that $|r_h(x)| \leq Ch^\alpha$ for some constant C, then $|f(x_h^*) - f(x^*)| \leq 2Ch^\alpha$.

(b) Suppose $f_h(x_h^*) = 0$ and $\lim_{h\to 0} x_h^* = x^*$. Then $f(x^*) = 0$. Moreover, if there exists $\alpha > 0$ such that $|r_h(x)| \leq Ch^\alpha$ for some constant C, then $|f(x_h^*)| \leq Ch^\alpha$.

Proof

(a) For any y, we have $f(x_h^*) + r_h(x_h^*) = f_h(x_h^*) \leq f_h(y)$. The sequence $\{x_h^*\}$ forms a compact set; hence $r_h(x_h^*) \to 0$ as $h \to 0$. Letting $h \to 0$, since $x_h^* \to x^*$ and f is continuous, we obtain $f(x^*) \leq f(y)$. Since y is arbitrary, $x^* \in \arg\min_x f(x)$.

Moreover, we have

$$0 \leq f(x_h^*) - f(x^*) = f_h(x_h^*) - r_h(x_h^*) - f_h(x^*) + r_h(x^*) \leq -r_h(x_h^*) + r_h(x^*) \leq 2Ch^\alpha.$$

(b) Since $f(x_h^*) + r_h(x_h^*) = f_h(x_h^*) = 0$, $|f(x_h^*)| = |r_h(x_h^*)|$. The sequence $\{x_h^*\}$ forms a compact set; hence $r_h(x_h^*) \to 0$ as $h \to 0$. Letting $h \to 0$, since $x_h^* \to x^*$ and f is continuous, we obtain $|f(x^*)| = 0$.

The second statement is straightforward since $|f(x_h^*)| = |r_h(x_h^*)| \leq Ch^\alpha$.

■

Lemma 9 *Under Assumptions 1 and 4, we have*

$$\mathbb{E}\left[\int_t^{t+h} |r(s, X_s) - r(t, X_t)|^2 \,\mathrm{d}s\right] \leq Ch^{2\mu_1+\mu_2+1}.$$

Proof By Assumption 4, for $s \in [t, t+h]$, we have

$$|r(s, X_s) - r(t, X_t)|^2 \leq Ch^{2\mu_1}|X_s - X_t|^{2\mu_2}(|X_s|^{2\mu_3} + |X_t|^{2\mu_3}).$$

When $\mu_2 > 0$, we take $p > 1$ sufficiently large such that $2\mu_2 p \geq 1$, and $q > 1$ such that $\frac{1}{p} + \frac{1}{q} = 1$.

Under Assumption 1, we have the usual moment estimate of the solution to an SDE, e.g., Yong and Zhou (1999, Chapter 1, Theorem 6.1). Together with Hölder's inequality, we have

$$\begin{aligned}
&\mathbb{E}\Big[\int_t^{t+h}|X_s-X_t|^{2\mu_2}(|X_s|^{2\mu_3}+|X_t|^{2\mu_3})\mathrm{d}s\Big]\\
\le&\left(\mathbb{E}\Big[\int_t^{t+h}|X_s-X_t|^{2\mu_2 p}\mathrm{d}s\Big]\right)^{1/p}\left(\mathbb{E}\Big[\int_t^{t+h}(|X_s|^{2\mu_3}+|X_t|^{2\mu_3})^q\mathrm{d}s\Big]\right)^{1/q}\\
\le&C\left(\mathbb{E}\Big[\int_t^{t+h}|X_s-X_t|^{2\mu_2 p}\mathrm{d}s\Big]\right)^{1/p}\left(h\max_{t\le s\le t+h}\mathbb{E}\left[|X_s|^{2\mu_3 q}\right]\right)^{1/q}\\
\le&C\left(\int_t^{t+h}(s-t)^{\mu_2 p}\mathrm{d}s\right)^{1/p}h^{1/q}\\
\le&Ch^{(\mu_2 p+1)/p}h^{1/q}=Ch^{\mu_2+1}.
\end{aligned}$$

When $\mu_2=0$, the above inequality also holds true as $\mathbb{E}\Big[\int_t^{t+h}|X_s-X_t|^{2\mu_2}(|X_s|^{2\mu_3}+|X_t|^{2\mu_3})\mathrm{d}s\Big]\le Ch.$ ■

Now we are already to prove Theorem 2.

Proof By Itô's lemma, we have

$$\begin{aligned}
&\sum_{i=0}^{K-1}\left(\frac{J^\theta(t_{t+1},X_{t_{i+1}})-J^\theta(t_i,X_{t_i})}{t_{i+1}-t_i}+r_{t_i}\right)^2\Delta t\\
=&\frac{1}{\Delta t}\sum_{i=0}^{K-1}\left(J^\theta(t_{i+1},X_{t_{i+1}})-J^\theta(t_i,X_{t_i})+\int_{t_i}^{t_{i+1}}r_{t_i}\mathrm{d}s\right)^2\\
=&\frac{1}{\Delta t}\sum_{i=0}^{K-1}\left(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s+\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\right)^2\\
=&\frac{1}{\Delta t}\sum_{i=0}^{K-1}\Bigg\{\left(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\right)^2+\left(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\right)^2\\
&+2\left(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\right)\left(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\right)\Bigg\}.
\end{aligned}$$

Itô's isometry implies

$$\mathbb{E}\Big[\left(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\right)^2\Big]=\mathbb{E}\int_{t_i}^{t_{i+1}}\Big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\Big|^2\mathrm{d}s.$$

Thus,

$$\begin{aligned}
\mathrm{MSTDE}_{\Delta t}(\theta)=&\frac{1}{\Delta t}\mathbb{E}\int_0^T\Big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\Big|^2\mathrm{d}s+\frac{1}{\Delta t}\sum_{i=0}^{K-1}\mathbb{E}\Big[\left(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\right)^2\Big]\\
&+\frac{2}{\Delta t}\sum_{i=0}^{K-1}\mathbb{E}\Big[\left(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\right)\left(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\right)\Big].
\end{aligned}$$

We write $\mathrm{MSTDE}_{\Delta t}(\theta)\Delta t = \mathrm{QV}(\theta) + R(\theta)$, where

$$\mathrm{QV}(\theta) := \mathbb{E}\int_0^T \big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top \sigma(s,X_s)\big|^2 \mathrm{d}s$$

and

$$\begin{aligned} R(\theta) := \sum_{i=0}^{K-1}\mathbb{E}\Big[&\Big(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\Big)^2 \\ &+2\Big(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\Big)\Big(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\Big)\Big]. \end{aligned}$$

We apply Cauchy-Schwarz inequality and obtain

$$\begin{aligned} |R(\theta)| \leq & \sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]^2\mathrm{d}s(\Delta t)^2 \\ & + 2\sum_{i=0}^{K-1}\left\{\mathbb{E}\Big[\Big(\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]\mathrm{d}s\Big)^2\Big]\mathbb{E}\Big[\Big(\int_{t_i}^{t_{i+1}}\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\mathrm{d}W_s\Big)^2\Big]\right\}^{1/2} \\ \leq & \sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]^2\mathrm{d}s(\Delta t)^2 \\ & + 2\sum_{i=0}^{K-1}\left\{\Big[\mathbb{E}\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+r_{t_i}]^2\mathrm{d}s(\Delta t)^2\Big]\Big[\mathbb{E}\int_{t_i}^{t_{i+1}}\big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\big|^2\mathrm{d}s\Big]\right\}^{1/2} \\ = & (\Delta t)^2\mathbb{E}\int_0^T[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s \\ & + 2\Delta t\sum_{i=0}^{K-1}\left\{\Big[\mathbb{E}\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s\Big]\Big[\mathbb{E}\int_{t_i}^{t_{i+1}}\big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\big|^2\mathrm{d}s\Big]\right\}^{1/2} \\ \leq & (\Delta t)^2\mathbb{E}\int_0^T[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s \\ & + 2\Delta t\left\{\sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s\right\}^{1/2}\left\{\sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}\big|\left(\frac{\partial J^\theta}{\partial x}\right)^\top\sigma(s,X_s)\big|^2\mathrm{d}s\right\}^{1/2} \\ = & (\Delta t)^2\mathbb{E}\int_0^T[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s + 2\Delta t\left\{\mathbb{E}\int_0^T[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s\right\}^{1/2}\sqrt{\mathrm{QV}(\theta)}, \end{aligned}$$

where $\bar{r}_s := r_{t_i}$ for the unique i such that $t_i \leq s < t_{i+1}$.

It follows from the triangle inequality that

$$\left\{\mathbb{E}\int_0^T[\mathcal{L}J^\theta(s,X_s)+\bar{r}_s]^2\mathrm{d}s\right\}^{1/2} = ||\mathcal{L}J^\theta(\cdot,X_\cdot)+\bar{r}_\cdot||_{L^2} \leq ||\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot||_{L^2} + ||r-\bar{r}||_{L^2}.$$

Hence,

$$
\begin{aligned}
|R(\theta)| \leq &(\Delta t)^2\big(||\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot||_{L^2}+||r-\bar{r}||_{L^2}\big)^2+2\Delta t\big(||\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot||_{L^2}+||r-\bar{r}||_{L^2}\big)\sqrt{QV(\theta)}\\
\leq &4(\Delta t)^2\big(2\,\mathrm{MSBE}(\theta)+||r-\bar{r}||_{L^2}^2\big)+2\Delta t\sqrt{\mathrm{QV}(\theta)}\big(\sqrt{2\,\mathrm{MSBE}(\theta)}+||r-\bar{r}||_{L^2}\big),
\end{aligned}
$$

where $\mathrm{MSBE}(\theta)=||\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot||_{L^2}^2$ is the *mean-square Bellman error*.

Because $\bar{r}$ is a simple process approximating r, we have $||r-\bar{r}||_{L^2}\to 0$ as $\Delta t\to 0$, which is independent of θ. For an arbitrary compact set Γ, Assumption 3 yields that $\mathrm{MSBE}(\theta)$ and $\mathrm{QV}(\theta)$ are both continuous functions of θ; hence $\sup_{\theta\in\Gamma}\mathrm{MSBE}(\theta)$ and $\sup_{\theta\in\Gamma}\mathrm{QV}(\theta)$ are both finite. Consequently,

$$
\begin{aligned}
\sup_{\theta\in\Gamma}|R(\theta)| \leq &4(\Delta t)^2\big(2\sup_{\theta\in\Gamma}\mathrm{MSBE}(\theta)+||r-\bar{r}||_{L^2}^2\big)\\
&+2\Delta t\sqrt{\sup_{\theta\in\Gamma}\mathrm{QV}(\theta)}\Big(\sqrt{2\sup_{\theta\in\Gamma}\mathrm{MSBE}(\theta)}+||r-\bar{r}||_{L^2}\Big)\to 0 \text{ as } \Delta t\to 0.
\end{aligned}
$$

The desired result now follows from Lemma 8.

Moreover, under Assumption 4, it follows from Lemma 9 that

$$
||r-\bar{r}||_{L^2}^2=\sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}(r(s,X_s)-r(t_i,X_{t_i}))^2\,\mathrm{d}s\leq K(\Delta t)^{2\mu_1+\mu_2+1}=(\Delta t)^{2\mu_1+\mu_2}.
$$

Therefore, our analysis above yields

$$
\sup_{\theta\in\Gamma}|R(\theta)|\leq C_1\Delta t+C_2(\Delta t)^2+C_3(\Delta t)^{\mu_1+\mu_2/2+1}+C_4(\Delta t)^{2\mu_1+\mu_2+2},
$$

where the leading term in the right hand side is $O(\Delta t)$. The desired result again follows from Lemma 8.

■

Proof of Theorem 3

Proof Since $M_t=J(t,X_t)+\int_0^t r(s,X_s)\mathrm{d}s$ is a martingale, we have

$$
\begin{aligned}
ML(\theta)=&\mathbb{E}\int_0^T|M_T-M_t^\theta|^2\mathrm{d}t\\
=&\mathbb{E}\int_0^T(M_T-M_t+M_t-M_t^\theta)^2\mathrm{d}t\\
=&\mathbb{E}\int_0^T[(M_T-M_t)^2+(M_t-M_t^\theta)^2+2(M_T-M_t)(M_t-M_t^\theta)]\mathrm{d}t\\
=&\mathbb{E}\int_0^T(M_T-M_t)^2\mathrm{d}t+\mathbb{E}\int_0^T(M_t-M_t^\theta)^2\mathrm{d}t+2\int_0^T\mathbb{E}\Big((M_t-M_t^\theta)\mathbb{E}[(M_T-M_t)|\mathcal{F}_t]\Big)\,\mathrm{d}t\\
=&\mathbb{E}\int_0^T|J(t,X_t)-J^\theta(t,X_t)|^2\mathrm{d}t+\mathbb{E}\int_0^T|M_T-M_t|^2\mathrm{d}t.
\end{aligned}
$$

The second term does not rely on θ. This proves our first statement.

Next, let us estimate the difference between the continuous-time and the discretized martingale loss functions. Denote

$$m(t,\theta) = \mathbb{E}[(M_T - M_t^\theta)^2] = \mathbb{E}[\big(h(X_T) - J^\theta(t, X_t) + \int_t^T r_s \mathrm{d}s\big)^2],$$

and

$$\Delta\tilde{M}_{t_i}^\theta = h(X_T) - J^\theta(t_i, X_{t_i}) + \sum_{j=i}^{K-1} r(t_j, X_{t_j})\Delta t = h(X_T) - J^\theta(t, X_t) + \int_t^T \bar{r}_s \mathrm{d}s,$$

where $\bar{r}_s := r_{t_i}$ for the unique i such that $t_i \le s < t_{i+1}$.

Then

$$\begin{aligned}\mathrm{ML}(\theta) - \mathrm{ML}_{\Delta t}(\theta) =& \int_0^T m(t,\theta)\mathrm{d}t - \sum_{i=0}^{K-1} m(t_i,\theta)\Delta t + \sum_{i=0}^{K-1} m(t_i,\theta)\Delta t - ML_{\Delta t}(\theta) \\ =& \sum_{i=0}^{K-1} \int_{t_i}^{t_{i+1}} [m(t,\theta) - m(t_i,\theta)]\mathrm{d}t + \Delta t \sum_{i=0}^{K-1} \mathbb{E}[(M_T - M_{t_i}^\theta)^2 - (\Delta\tilde{M}_{t_i}^\theta)^2].\end{aligned}$$

The first term is bounded by

$$\left|\sum_{i=0}^{K-1} \int_{t_i}^{t_{i+1}} [m(t,\theta) - m(t_i,\theta)]\mathrm{d}t\right| \le \sum_{i=0}^{K-1} \sup_{t\in[0,T]} |\frac{\partial m}{\partial t}(t,\theta)| \int_{t_i}^{t_{i+1}} (t - t_i)\mathrm{d}t = \frac{T}{2} \sup_{t\in[0,T]} |\frac{\partial m}{\partial t}(t,\theta)|\Delta t.$$

To estimate the second term, recall that

$$M_T - M_{t_i}^\theta - \Delta\tilde{M}_{t_i}^\theta = \int_{t_i}^T (r_s - \bar{r}_s)\mathrm{d}s.$$

Hence

$$\begin{aligned}\big|\mathbb{E}[(M_T - M_{t_i}^\theta)^2 - (\Delta\tilde{M}_{t_i}^\theta)^2]\big| =& \left|\mathbb{E}[2\int_{t_i}^T (r_s - \bar{r}_s)\mathrm{d}s(M_T - M_{t_i}^\theta) - \big(\int_{t_i}^T (r_s - \bar{r}_s)\mathrm{d}s\big)^2]\right| \\ \le& 2m(t_i,\theta)^{\frac{1}{2}} (\mathbb{E}[(\int_{t_i}^T (r_s - \bar{r}_s)\mathrm{d}s)^2])^{\frac{1}{2}} + \mathbb{E}[(\int_{t_i}^T (r_s - \bar{r}_s)\mathrm{d}s)^2] \\ \le& 2T \sup_{t\in[0,T]} |m(t,\theta)|^{\frac{1}{2}} ||r - \bar{r}||_{L^2} + T^2 ||r - \bar{r}||_{L^2}^2.\end{aligned}$$

Therefore, we have proved

$$|\,\mathrm{ML}(\theta) - \mathrm{ML}_{\Delta t}(\theta)| \le \frac{T}{2} \sup_{t\in[0,T]} |\frac{\partial m}{\partial t}(t,\theta)|\Delta t + 2T^2 \sup_{t\in[0,T]} |m(t,\theta)|^{\frac{1}{2}} ||r - \bar{r}||_{L^2} + T^3 ||r - \bar{r}||_{L^2}^2.$$

For an arbitrary compact set Γ, under Assumption 3, $\sup_{t\in[0,T],\theta\in\Gamma} |\frac{\partial m}{\partial t}(t,\theta)| + \sup_{t\in[0,T],\theta\in\Gamma} |m(t,\theta)| < \infty$, and $||r - \bar{r}||_{L^2} \to 0$. Hence as $\Delta t \to 0$,

$$\sup_{\theta\in\Gamma} |\,\mathrm{ML}(\theta) - \mathrm{ML}_{\Delta t}(\theta)| \to 0.$$

By Lemma 8, we obtain the desired conclusion.

Moreover, under Assumption 4, it follows from Lemma 9 and the proof of Theorem 2 that

$$\sup_{\theta\in\Gamma} |\,\mathrm{ML}(\theta) - \mathrm{ML}_{\Delta t}(\theta)| \le C_1\Delta t + C_2(\Delta t)^{\mu_1+\mu_2/2} + C_3(\Delta t)^{2\mu_1+\mu_2},$$

where the leading term in the right hand side is $O\left((\Delta t)^{\min\{1,\mu_1+\frac{\mu_2}{2}\}}\right)$. The desired result again follows from Lemma 8.

■

Proof of Proposition 4

Proof The "only if" part is evident. To prove the "if" part, assume that $\mathrm{d}M_t^\theta = A_t\mathrm{d}t + B_t\mathrm{d}W_t$. In particular, in our case, $A_t = \mathcal{L}J^\theta(t, X_t) + r_t$ and $B_t = (\frac{\partial J^\theta}{\partial x})^\top\sigma(t, X_t)$. $A, B \in L^2_{\mathcal{F}}([0, T])$ follows from Assumption 3. For any $0 \le s < s' \le T$, take $\xi_t = sgn(A_t)$ if $t \in [s, s']$ and $\xi_t = 0$ otherwise. Then

$$0 = \mathbb{E}\int_s^{s'} \xi_t\mathrm{d}M_t^\theta = \mathbb{E}\int_s^{s'} (|A_t|\mathrm{d}t + \xi_tB_t\mathrm{d}W_t) = \mathbb{E}\int_s^{s'} |A_t|\mathrm{d}t,$$

where the expectation of the second term vanishes because $|\xi B| \le |B| \in L^2_{\mathcal{F}}([0, T])$ and hence $\mathbb{E}\int_0^{\cdot} \xi_tB_t\mathrm{d}W_t$ is a martingale. This yields $A_t = 0$ almost surely, and thus M^θ is a martingale. ■

Proof of Theorem 5

Proof Based on Lemma 8, it suffices to examine the difference

$$
\begin{aligned}
&\big|\mathbb{E}\int_0^T \xi_t \mathrm{d}M_t^\theta - \mathbb{E}\sum_{i=0}^{K-1} \xi_{t_i}(M_{t_{i+1}}^\theta - M_{t_i}^\theta)\big| = \big|\mathbb{E}\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}(\xi_t - \xi_{t_i})\mathrm{d}M_t^\theta\big| + \big|\mathbb{E}\sum_{i=0}^{K-1}\xi_{t_i}\int_{t_i}^{t_{i+1}}(r_s - r_{t_i})\mathrm{d}s\big| \\
\leq & \mathbb{E}\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}} |\xi_t - \xi_{t_i}|\cdot|\mathcal{L}J^\theta(t,X_t) + r_t|\mathrm{d}t + \mathbb{E}\Big[\Big(\sum_{i=0}^{K-1}\xi_{t_i}^2\Big)^{1/2}\Big(\sum_{i=0}^{K-1}\Big(\int_{t_i}^{t_{i+1}}(r_s - r_{t_i})\mathrm{d}s\Big)^2\Big)^{1/2}\Big] \\
\leq & \sum_{i=0}^{K-1}\big(\mathbb{E}\int_{t_i}^{t_{i+1}}|\xi_t - \xi_s|^2\mathrm{d}t\big)^{1/2}\Big(\mathbb{E}\int_{t_i}^{t_{i+1}}\big(\mathcal{L}J^\theta(t,X_t) + r_t\big)^2\mathrm{d}t\Big)^{1/2} \\
& \mathbb{E}\Big[\Big(\sum_{i=0}^{K-1}\xi_{t_i}^2\Big)^{1/2}\Big(\sum_{i=0}^{K-1}\Big(\int_{t_i}^{t_{i+1}}(r_s - r_{t_i})^2\mathrm{d}s\Big)\Delta t\Big)^{1/2}\Big] \\
\leq & \sum_{i=0}^{K-1}\Big(\mathbb{E}\int_{t_i}^{t_{i+1}}\big(\mathcal{L}J^\theta(t,X_t) + r_t\big)^2\mathrm{d}t\Big)^{1/2}\Big(\int_{t_i}^{t_{i+1}}C(t-t_i)^\alpha\mathrm{d}t\Big)^{1/2} + (\Delta t)^{1/2}||r - \bar{r}||_{L^2}\Big(\sum_{i=0}^{K-1}\mathbb{E}[\xi_{t_i}^2]\Big)^{1/2} \\
\leq & \sum_{i=0}^{K-1}\Big(\mathbb{E}\int_{t_i}^{t_{i+1}}\big(\mathcal{L}J^\theta(t,X_t) + r_t\big)^2\mathrm{d}t\Big)^{1/2}\sqrt{\frac{C}{1+\alpha}}(\Delta t)^{\frac{1+\alpha}{2}} + ||r - \bar{r}||_{L^2}||\bar{\xi}||_{L^2} \\
\leq & \Big(\sum_{i=0}^{K-1}\mathbb{E}\int_{t_i}^{t_{i+1}}\big(\mathcal{L}J^\theta(t,X_t) + r_t\big)^2\mathrm{d}t\Big)^{1/2}K^{1/2}\sqrt{\frac{C}{1+\alpha}}(\Delta t)^{\frac{1+\alpha}{2}} + ||\bar{\xi}||_{L^2}(\Delta t)^{\mu_1+\mu_2/2} \\
\leq & ||\mathcal{L}J^\theta(\cdot, X_\cdot) + r_\cdot||_{L^2}\sqrt{\frac{CT}{1+\alpha}}(\Delta t)^{\frac{\alpha}{2}} + ||\bar{\xi}||_{L^2}(\Delta t)^{\mu_1+\mu_2/2}.
\end{aligned}
$$

Hence, for an arbitrary compact set Γ, under Assumption 3, we have

$$
\begin{aligned}
&\sup_{\theta\in\Gamma}\big|\mathbb{E}\int_0^T \xi_t\mathrm{d}M_t^\theta - \mathbb{E}\sum_{i=0}^{K-1}\xi_{t_i}(M_{t_{i+1}}^\theta - M_{t_i}^\theta)\big| \\
\leq & \sup_{\theta\in\Gamma}||\mathcal{L}J^\theta(\cdot, X_\cdot) + r_\cdot||_{L^2}\sqrt{\frac{CT}{1+\alpha}}(\Delta t)^{\frac{\alpha}{2}} + \sup_{\theta\in\Gamma}||\bar{\xi}||_{L^2}(\Delta t)^{\mu_1+\mu_2/2} \to 0,
\end{aligned}
$$

as $\Delta t \to 0$.

Since the leading term above is $O\big((\Delta t)^{\min\{\alpha/2,\ \mu_1+\mu_2/2\}}\big)$, we obtain the convergence rate in view of Lemma 8. ■

Proof of Theorem 6

We first prove an error estimate of the following form:

$$
\begin{aligned}
&\left|(b+\Delta b)^\top (D+\Delta D)(b+\Delta b) - b^\top D b\right| \\
=&\left|D\circ[(b+\Delta b)(b+\Delta b)^\top - bb^\top] + \Delta D \circ (b+\Delta b)(b+\Delta b)^\top\right| \\
\le&|D||(b+\Delta b)(b+\Delta b)^\top - bb^\top| + |\Delta D||(b+\Delta b)(b+\Delta b)^\top| \\
=&|D||\Delta b \Delta b^\top + b\Delta b^\top + \Delta b b^\top| + |\Delta D||b+\Delta b|^2 \\
=&|D||\Delta b|^2 + 2|D||b||\Delta b| + 2|\Delta D||b|^2 + 2|\Delta D||\Delta b|^2.
\end{aligned}
$$

Based on the proof of Theorem 5, we have that for an arbitrary compact set Γ,

$$
\begin{aligned}
&\sup_{\theta\in\Gamma}\left|\mathbb{E}\int_0^T \xi_t \mathrm{d}M_t^\theta - \mathbb{E}\sum_{i=0}^{K-1}\xi_{t_i}(M_{t_{i+1}}^\theta - M_{t_i}^\theta)\right| \\
&\le \sup_{\theta\in\Gamma}||\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot||_{L^2}\sqrt{\frac{CT}{1+\alpha}}(\Delta t)^{\frac{\alpha}{2}} + \sup_{\theta\in\Gamma}||\bar{\xi}||_{L^2}(\Delta t)^{\mu_1+\mu_2/2} \to 0.
\end{aligned}
$$

Given that $|A_{\Delta t} - A| \le \tilde{C}(\theta)|\Delta t|^\beta$, we get

$$
\sup_{\theta\in\Gamma}|\operatorname{GMM}_{\Delta t}(\theta) - \operatorname{GMM}(\theta)| \to 0,
$$

as $\Delta t \to 0$. By Lemma 8, we obtain the desired results.

Moreover, based on the error estimate of the quadratic form, we obtain

$$
\begin{aligned}
&\sup_{\theta\in\Gamma}|\operatorname{GMM}_{\Delta t}(\theta) - \operatorname{GMM}(\theta)| \\
\le& C\left[O\left((\Delta t)^{\alpha/2}\right) + O\left((\Delta t)^{\mu_1+\mu_2/2}\right) + O\left((\Delta t)^{\beta}\right) + o\left((\Delta t)^{\alpha/2} + (\Delta t)^{\mu_1+\mu_2/2} + (\Delta t)^{\beta}\right)\right],
\end{aligned}
$$

where the leading term is $O\left((\Delta t)^{\min\{\alpha/2,\ \mu_1+\mu_2/2,\ \beta\}}\right)$.

In particular, when $A = \left[\mathbb{E}\int_0^T \xi_t^\theta(\xi_t^\theta)^\top \mathrm{d}t\right]^{-1}$ and $A_{\Delta t} = \left[\mathbb{E}\sum_{i=0}^{K-1}\xi_{t_i}^\theta(\xi_{t_i}^\theta)^\top \Delta t\right]^{-1}$, we claim the condition $|A_{\Delta t} - A| \le \tilde{C}(\theta)|\Delta t|^\beta$ holds true. To see this, recall that

$$
|(D+\Delta D)^{-1} - D^{-1}| = |\sum_{k=0}^\infty (D^{-1}\Delta D)^k D^{-1} - D^{-1}| \le \sum_{k=0}^\infty |D^{-1}\Delta D||(D^{-1}\Delta D)^k D^{-1}| \le \frac{|D^{-1}|^2|\Delta D|}{1-|D^{-1}||\Delta D|}.
$$

Thus, it suffices to estimate the difference

$$
\begin{aligned}
&\big|\mathbb{E}\int_0^T \xi_t^\theta(\xi_t^\theta)^\top \mathrm{d}t - \mathbb{E}\sum_{i=0}^{K-1}\xi_{t_i}^\theta(\xi_{t_i}^\theta)^\top\Delta t\big| = \big|\mathbb{E}\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}[\xi_t^\theta(\xi_t^\theta)^\top - \xi_{t_i}^\theta(\xi_{t_i}^\theta)^\top]\mathrm{d}t\big|\\
\le&\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}\mathbb{E}[\xi_t^\theta(\xi_t^\theta)^\top - \xi_{t_i}^\theta(\xi_{t_i}^\theta)^\top]\mathrm{d}t\\
\le&\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}\mathbb{E}[|\xi_t^\theta - \xi_{t_i}^\theta|^2 + 2|\xi_t^\theta - \xi_{t_i}^\theta||\xi_t^\theta|]\mathrm{d}t\\
\le&\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}C(\theta)(t-t_i)^\alpha\mathrm{d}t + 2\sum_{i=0}^{K-1}\int_{t_i}^{t_{i+1}}\mathbb{E}[|\xi_t^\theta - \xi_{t_i}^\theta||\xi_t^\theta|]\mathrm{d}t\\
\le&\frac{TC(\theta)}{1+\alpha}(\Delta t)^\alpha + 2\sum_{i=0}^{K-1}\big(\mathbb{E}\int_{t_i}^{t_{i+1}}|\xi_t^\theta - \xi_{t_i}^\theta|^2\mathrm{d}t\big)^{1/2}\big(\mathbb{E}\int_{t_i}^{t_{i+1}}|\xi_t^\theta|^2\mathrm{d}t\big)^{1/2}\\
\le&\frac{TC(\theta)}{1+\alpha}(\Delta t)^\alpha + 2\sqrt{\frac{C(\theta)}{1+\alpha}}(\Delta t)^{\frac{1}{2}+\frac{\alpha}{2}}\sum_{i=0}^{K-1}\big(\mathbb{E}\int_{t_i}^{t_{i+1}}|\xi_t^\theta|^2\mathrm{d}t\big)^{1/2}\\
\le&\frac{TC(\theta)}{1+\alpha}(\Delta t)^\alpha + 2\sqrt{\frac{TC(\theta)}{1+\alpha}}(\Delta t)^{\frac{\alpha}{2}}||\xi^\theta||_{L^2}.
\end{aligned}
$$

Proof of Theorem 7

Proof Denote by $\langle\kappa,\tilde\kappa\rangle_{L^2} := \mathbb{E}\int_0^T\kappa_t\tilde\kappa_t\mathrm{d}t$ the inner product in $L^2_{\mathcal{F}}([0,T])$. It follows from the property of projection that $\langle\kappa - \Pi_\theta\kappa, \xi^{\theta,(j)}\rangle_{L^2} = 0$ for any $\kappa\in L^2_{\mathcal{F}}([0,T])$ and all $j=1,\cdots,L'$.

As a stochastic process, $\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\in L^2_{\mathcal{F}}([0,T])$. Write

$$
\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big) = \sum_{i=1}^{L'}\alpha^{(i)}(\theta)\xi_\cdot^{\theta,(i)} =: \alpha(\theta)^\top\xi_\cdot^\theta.
$$

Then

$$
\begin{aligned}
&\langle\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big),\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big)\rangle_{L^2}\\
=&\sum_{1\le i,j\le L'}\alpha^{(i)}(\theta)\alpha^{(j)}(\theta)\langle\xi_\cdot^{\theta,(i)},\xi_\cdot^{\theta,(j)}\rangle_{L^2} = \alpha(\theta)^\top A^\theta\alpha(\theta),
\end{aligned}
$$

where the ij-th entry of the $L'\times L'$ matrix A^θ is $\langle\xi_\cdot^{\theta,(i)},\xi_\cdot^{\theta,(j)}\rangle_{L^2}$.

On the other hand,

$$
\mathbb{E}\left[\int_0^T\left(\Pi_\theta\left(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\right)\right)\xi_t^\theta\mathrm{d}t\right] = \mathbb{E}\left[\int_0^T\left(\mathcal{L}J^\theta(t,X_t)+r_t\right)\xi_t^\theta\mathrm{d}t\right] = A^\theta\alpha(\theta).
$$

Therefore,

$$
\begin{aligned}
&\frac{1}{2}\mathbb{E}\left[\int_0^T \big(\mathcal{L}J^\theta(t,X_t)+r_t\big)\xi_t^\theta \mathrm{d}t\right]^\top \left[\mathbb{E}\int_0^T \xi_t^\theta(\xi_t^\theta)^\top \mathrm{d}t\right]^{-1} \mathbb{E}\left[\int_0^T \big(\mathcal{L}J^\theta(t,X_t)+r_t\big)\xi_t^\theta \mathrm{d}t\right]\\
=&\frac{1}{2}\alpha(\theta)^\top A^\theta (A^\theta)^{-1} A^\theta \alpha(\theta)\\
=&\frac{1}{2}\mathbb{E}\int_0^T \left|\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big)\right|^2 \mathrm{d}t = \frac{1}{2}||\Pi_\theta\big(\mathcal{L}J^\theta(\cdot,X_\cdot)+r_\cdot\big)||_{L^2}^2 = \mathrm{MSPBE}(\theta).
\end{aligned}
$$

■

References

L. C. Baird. Advantage updating. Technical report, Write Lab Wright-Patterson Air Force Base, OH 45433-7301, USA, 1993.

L. C. Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pages 30–37. Elsevier, 1995.

E. Barnard. Temporal-difference methods and Markov models. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(2):357–365, 1993.

O. E. Barndorff-Nielsen and N. Shephard. Estimating quadratic variation using realized variance. *Journal of Applied Econometrics*, 17(5):457–477, 2002.

C. Beck, M. Hutzenthaler, and A. Jentzen. On nonlinear feynman–kac formulas for viscosity solutions of semilinear parabolic partial differential equations. *Stochastics and Dynamics*, 21(08):2150048, 2021.

J. A. Boyan. Technical update: Least-squares temporal difference learning. *Machine Learning*, 49(2):233–246, 2002.

S. J. Bradtke and A. G. Barto. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22(1):33–57, 1996.

M. G. Crandall, H. Ishii, and P.-L. Lions. Users guide to viscosity solutions of second order partial differential equations. *Bulletin of the American mathematical society*, 27(1):1–67, 1992.

M. Dai, Y. Dong, and Y. Jia. Learning equilibrium mean-variance strategy. *SSRN preprint SSRN:3770818*, 2020.

K. Doya. Reinforcement learning in continuous time and space. *Neural Computation*, 12 (1):219–245, 2000.

S. S. Du, J. Chen, L. Li, L. Xiao, and D. Zhou. Stochastic variance reduction methods for policy evaluation. In *International Conference on Machine Learning*, pages 1049–1058. PMLR, 2017.

N. El Karoui, S. Peng, and M. C. Quenez. Backward stochastic differential equations in finance. *Mathematical Finance*, 7(1):1–71, 1997.

W. H. Fleming and H. M. Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer Science & Business Media, 2006.

N. Frémaux, H. Sprekeler, and W. Gerstner. Reinforcement learning using a continuous time actor-critic framework with spiking neurons. *PLoS Computational Biology*, 9(4): e1003024, 2013.

X. Gao, Z. Q. Xu, and X. Y. Zhou. State-dependent temperature control for Langevin diffusions. *SIAM Journal on Control and Optimization*, pages 1–26, 2020.

A. Geramifard, M. Bowling, and R. S. Sutton. Incremental least-squares temporal difference learning. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 356. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

X. Guo, R. Xu, and T. Zariphopoulou. Entropy regularization for mean field games with learning. *Mathematics of Operations Research*, 2022.

J. Han, A. Jentzen, and W. E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.

L. P. Hansen. Large sample properties of generalized method of moments estimators. *Econometrica*, pages 1029–1054, 1982.

L. P. Hansen, J. Heaton, and A. Yaron. Finite-sample properties of some alternative GMM estimators. *Journal of Business & Economic Statistics*, 14(3):262–280, 1996.

S. Hochreiter and J. Schmidhuber. LSTM can solve hard long time lag problems. *Advances in neural information processing systems*, pages 473–479, 1997.

C. Huré, H. Pham, and X. Warin. Some machine learning schemes for high-dimensional nonlinear PDEs. *arXiv preprint arXiv:1902.01599*, 33, 2019.

I. Karatzas and S. Shreve. *Brownian Motion and Stochastic Calculus*, volume 113. Springer, 2014.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations.* Springer, 1992.

J. Lee and R. S. Sutton. Policy iterations for reinforcement learning problems in continuous time and space – Fundamental theory and methods. *Automatica*, 126:109421, 2021.

B. Liu, J. Liu, M. Ghavamzadeh, S. Mahadevan, and M. Petrik. Proximal gradient temporal difference learning algorithms. In *IJCAI*, pages 4195–4199, 2016.

L. Ljung and T. Söderström. *Theory and practice of recursive identification.* MIT press, 1983.

H. R. Maei, C. Szepesvari, S. Bhatnagar, D. Precup, D. Silver, and R. S. Sutton. Convergent temporal-difference learning with arbitrary smooth function approximation. In *NIPS*, pages 1204–1212, 2009.

B. B. Mandelbrot and J. W. Van Ness. Fractional Brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, 1968.

M. Raissi. Deep hidden physics models: Deep learning of nonlinear partial differential equations. *Journal of Machine Learning Research*, 19(1):932–955, 2018.

H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

T. Sottinen and L. Viitasaari. Prediction law of fractional Brownian motion. *Statistics & Probability Letters*, 129:155–166, 2017.

D. W. Stroock and S. R. S. Varadhan. *Multidimensional diffusion processes*, volume 233 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences].* Springer-Verlag, Berlin-New York, 1979.

R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* Cambridge, MA: MIT Press, 2018.

R. S. Sutton, C. Szepesvári, and H. R. Maei. A convergent o(n) temporal-difference algorithm for off-policy learning with linear function approximation. In *NIPS*, 2008.

R. S. Sutton, H. R. Maei, D. Precup, S. Bhatnagar, D. Silver, C. Szepesvári, and E. Wiewiora. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 993–1000, 2009.

K. G. Vamvoudakis and F. L. Lewis. Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, 46(5):878–888, 2010.

H. Wang and X. Y. Zhou. Continuous-time mean–variance portfolio selection: A reinforcement learning framework. *Mathematical Finance*, 30(4):1273–1308, 2020.

H. Wang, T. Zariphopoulou, and X. Y. Zhou. Reinforcement learning in continuous time and space: A stochastic control approach. *Journal of Machine Learning Research*, 21 (198):1–34, 2020.

X. Xu, H.-g. He, and D. Hu. Efficient reinforcement learning using recursive least-squares methods. *Journal of Artificial Intelligence Research*, 16:259–292, 2002.

J. Yong and X. Y. Zhou. *Stochastic Controls: Hamiltonian Systems and HJB Equations.* New York, NY: Spinger, 1999.