Neural feedback approximation for stochastic control with degenerate diffusions: error estimates and numerical analysis

Olivier Bokanowski* Jean-François Chassagneux† Marco Scaratti‡§
Xavier Warin¶

July 13, 2026

Abstract

We study finite-horizon stochastic optimal control problems and approximate the resulting time-discrete formulation by a direct policy-learning problem over neural-network feedback maps. We prove a quantitative convergence estimate, in an averaged sense, for the error between the time-discrete value and the value induced by an approximately optimized neural policy. The bound separates the approximation of near-optimal feedback policies, the localization of stochastic trajectories on compact sets, and the optimization tolerance in training. The analysis does not require transition-density assumptions and covers possibly degenerate diffusions and deterministic controlled dynamics in a unified framework. Numerical experiments are provided for a degenerate stochastic radial target problem, a Hamilton–Jacobi–Bellman benchmark, and a gas storage problem, illustrating the approach and separating the main error sources: time discretization, restriction to piecewise-constant policies, neural-network approximation, and Monte Carlo evaluation.

Keywords. Stochastic optimal control; degenerate diffusions; neural networks; feedback control approximation; error estimates; time-discrete approximation.

MSC 2020. Primary: 93E20, 49M25, 65C30. Secondary: 68T07, 60H10.

1 Introduction

Stochastic optimal control problems arise in many applications in which decisions must be made dynamically under uncertainty, including finance, economics, energy management, operations research, engineering, and models of large interacting populations. In continuous time, these problems are usually formulated through controlled stochastic differential equations and have been studied through several classical approaches: among others, the dynamic programming principle (DPP) and the associated Hamilton–Jacobi–Bellman (HJB) equation [17, 36], weak formulations based on martingale methods, changes of measure, backward stochastic differential equations (BSDEs) [16], and stochastic maximum principle methods based on Hamiltonians and adjoint equations [35, 52]. Despite this well-established theory, the numerical solution of stochastic control problems remains challenging in moderate and high dimensions. Grid-based

*Université Paris Cité, Laboratoire Jacques-Louis Lions, olivier.bokanowski@u-paris.fr. This research benefited from the support of the FMJH Program PGMO and from the support to this program from EDF.

†ENSAE, CREST and Institut Polytechnique de Paris, jean-francois.chassagneux@ensae.fr. This research has benefited from the support of the ANR Project ReLISCoP (ANR-21-CE40-0001).

‡Corresponding author: marco.scaratti@univr.it.

§University of Verona and Université Paris Cité, Laboratoire Jacques-Louis Lions.

¶EDF Lab Paris-Saclay and FiME, Laboratoire de Finance des Marchés de l'Energie, 91120 Palaiseau, France, xavier.warin@edf.fr.

approximations of HJB equations suffer from the curse of dimensionality, while computing accurate feedback controls is particularly difficult when the dynamics are degenerate or when the diffusion coefficients are controlled.

In recent years, neural-network methods have provided a flexible computational tool for high-dimensional stochastic control and related nonlinear PDEs. Broadly speaking, the existing literature can be divided into two main families. The first one is *value-based*: the neural network is used to approximate the value function, its gradient, a BSDE component, or the residual of a PDE. The second one is *policy-based*: the feedback control itself is parameterized by a neural network and optimized directly through simulated trajectories. These two points of view are related, but they lead to different algorithms and to different theoretical questions.

A first major line of work concerns neural-network solvers for high-dimensional PDEs and BSDEs. The deep BSDE method of [23] reformulates semilinear parabolic PDEs through BSDEs and approximates the unknown gradient, or equivalently the BSDE control process, by neural networks. Several extensions and variants of this probabilistic PDE viewpoint have since been developed; see, for instance, [15, 29, 6, 38, 20]. The convergence of the deep BSDE method was analyzed in [24], where a posteriori error estimates are proved for coupled FBSDEs under assumptions ensuring stability and neural-network approximation. More recent theoretical analyses have also addressed stochastic-control formulations based on the stochastic maximum principle; see, for instance, [27]. These works are closely related to stochastic control, but their primary object is the PDE/FBSDE representation: neural networks approximate the value function, its gradient, or adjoint variables, rather than directly approximating a Markov feedback policy. Other equation-based approaches include residual or collocation methods, such as physics-informed neural networks (PINNs) [42] and the deep Galerkin method [43], as well as actor-critic and generalized policy-iteration schemes for HJB equations [53, 13, 33]. These methods are complementary to the direct feedback-policy approximation studied here.

A second line of work, closer to the present paper, is based on dynamic programming in discrete time. [28, 4, 1] combine deep neural networks with the dynamic programming principle to solve finite-horizon stochastic control problems. In particular, in [28, 4] the proposed algorithms first approximate optimal policies by neural networks and then approximate value functions through Monte Carlo regression, using performance iteration, hybrid iteration, and regress-now methods. The convergence analysis is expressed in terms of neural-network approximation errors and statistical regression errors, while the optimization error is left aside. The two works are particularly relevant for the present paper because they also approximate feedback controls by neural networks and provide theoretical guarantees. The main difference is methodological. Their algorithms are backward, DPP-based, and local in time, whereas the method studied here is a global direct-policy method: all feedback policies on the time grid are optimized simultaneously by minimizing the expected cost induced by the simulated controlled dynamics. A further difference concerns the treatment of degenerate dynamics. The error analysis in [28] relies on a bounded-density assumption for the controlled transition kernels with respect to the training distribution. This assumption rules out singular transition kernels, and therefore excludes deterministic dynamics and degenerate stochastic schemes whose transition laws are not absolutely continuous with respect to the training distribution. By contrast, our estimates are based on moment stability estimates for controlled trajectories and localization arguments and do not require any transition-density assumption. This allows us to treat possibly degenerate diffusions as well as deterministic controlled dynamics.

The direct policy-learning approach for high-dimensional stochastic control was proposed in [22]. There, the time-dependent controls are represented by feedforward neural networks, stacked through the discretized dynamics, and the objective functional of the stochastic control problem is used directly as the training loss. Note that the underlying idea of optimizing a parametrized policy directly through simulated costs was already used, e.g., in [5] for gas storage and swing option problems, with low-dimensional parametric families of strategies in

place of neural networks. In both cases, this global forward approach is algorithmically close to the method analyzed in the present paper.

Related policy-based and continuous-time reinforcement-learning approaches have been developed in several directions, including continuous-time portfolio selection [49], actor-critic methods for mean-field control [19, 37], and simulation-free methods for stochastic control [26]. These works demonstrate the relevance of policy parameterization and trajectory-based optimization. However, many of them focus primarily on algorithm design, numerical performance, or the training procedure itself. They do not provide the type of quantitative feedback-policy approximation estimate developed here, in which the value error is decomposed into explicit policy approximation, localization, and optimization terms.

Our error analysis is closely related to recent weak error estimates for neural-network approximations of feedback strategies in deterministic control problems. In [9], neural networks are used to approximate feedback controls for first-order HJB equations, with error estimates in an averaged norm. The subsequent work [10] develops representation results and weak error estimates for differential games. In the spirit of these two works, our error estimate is measured in a weak or averaged sense. However, the stochastic setting requires new stability estimates for controlled stochastic numerical schemes and a localization argument to handle trajectories that may leave every compact subset of the state space with positive probability. Further related literature concerns neural approximation of feedback controls for stochastic partial differential equations (SPDEs); for instance, the works [46, 45] develop approximation results for optimal feedback controls in stochastic reaction-diffusion settings, including neural-network approximations.

Although the present paper is written in a classical finite-dimensional setting, the analysis is also relevant for high-dimensional control problems obtained from particle or finite-dimensional approximations of infinite-dimensional models, such as mean field games and mean field control; see, e.g., [25]. Finite-particle and symmetric finite-dimensional approximations provide one way to reduce such problems to high-dimensional stochastic control problems, where neural-network methods can then be applied; see, for example, [40, 44, 8].

This paper develops a quantitative convergence analysis for direct neural-network approximations of feedback policies in finite-horizon stochastic control, allowing for degenerate diffusions and deterministic controlled dynamics. The analysis is carried out for the value function of a fixed time-discretized control problem, obtained by restricting controls to the time grid and discretizing the controlled dynamics. The main result, Theorem 4.6, provides an explicit averaged error estimate for the value induced by an approximate neural feedback policy. The bound separates the approximation error of a near-optimal feedback policy by the neural-network class on compact sets, the localization error due to stochastic trajectories leaving these compact sets, and the optimization tolerance of the training problem. This decomposition is the central theoretical contribution of the paper: it identifies separately the roles of feedback approximation, stochastic propagation, and numerical optimization in the error of direct policy learning. A distinctive feature of the analysis is that it does not require transition-density assumptions or non-degeneracy of the diffusion coefficient. It is instead based on stability estimates for controlled stochastic numerical schemes, approximation of near-optimal measurable feedback policies by Lipschitz feedbacks in value, and probabilistic localization via moment bounds. The resulting framework therefore covers non-degenerate diffusions, degenerate diffusions, and deterministic controlled dynamics in a unified way. We stress that, while the error estimate of Theorem 4.6 is fully quantitative, the Lipschitz approximation of near-optimal measurable feedbacks is not: the Lipschitz constant of the approximating policy is not quantified, so that the resulting convergence result (Theorem 4.10) does not provide a rate in terms of the size of the approximation class; see Remarks 4.8 and 4.13 for a discussion.

The convergence theorem is deliberately formulated at the level of a fixed time grid. The discretization error between the continuous-time problem and the time-discrete problem is treated

separately, using standard estimates for piecewise-constant controls and numerical schemes [39, 3, 30, 31]. This separation makes explicit the two layers of approximation: first, the discretization of the original stochastic control problem; second, the neural-network approximation of the resulting discrete feedback problem.

The numerical experiments are designed to complement the theoretical analysis by isolating the main error mechanisms that arise in practice. We consider three examples. The first is a two-dimensional stochastic radial target problem with degenerate diffusion. The results show that the expected time-discretization orders are recovered when the neural-network and Monte Carlo errors remain below the discretization error. The second example is an HJB benchmark for which semi-analytic formulas for both the value and the optimal feedback are available, allowing for comparison of both the learned values and controls. The experiment identifies the regime in which neural-network approximation and optimization errors become dominant after the time-discretization error has been sufficiently reduced. In both examples, we additionally compare Euler–Maruyama with a weak second-order scheme. The third example is a constrained gas storage problem motivated by energy applications, providing a realistic test case for which no closed-form benchmark is available. In this case, the exogenous price factors are simulated exactly on the grid, so the observed convergence mainly reflects the refinement of the decision grid, together with residual neural-network and training errors.

Taken together, the numerical results support the theoretical error decomposition: the experiments isolate the effects of time discretization, restriction to piecewise-constant policies, neural-network approximation and optimization, and Monte Carlo evaluation. To the best of our knowledge, a comparable combination of quantitative error estimates for global direct neural feedback learning and numerical separation of the main error sources within a framework that covers degenerate diffusions and deterministic controlled dynamics is not available in the existing literature.

The paper is organized as follows. Sections 2 and 3 introduce the continuous-time control problem, the time-discrete approximation, and the feedback formulation. Section 4 proves the stability, localization, and approximation estimates leading to the main convergence theorem. Section 5 describes the direct policy-learning algorithm used in the experiments, and Section 6 presents the numerical examples.

Notation. Given two measurable spaces $(X, \mathcal{X})$ and $(Y, \mathcal{Y})$, we denote by $\mathcal{M}(X; Y)$ the set of measurable functions from X to Y. We adopt $|\cdot|$ for both the Euclidean norm and the Frobenius norm. Finally, for any $R \geq 0$, $B_R := B(0, R)$ denotes the (Euclidean) closed ball of radius R. If, in addition, $(X, |\cdot|_X), (Y, |\cdot|_Y)$ are normed spaces and $\varphi : X \to Y$, we denote by

$$[\varphi] := \sup_{x \neq x'} \frac{|\varphi(x) - \varphi(x')|_Y}{|x - x'|_X}$$

its Lipschitz constant, whenever finite.

2 Continuous-time setting

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space endowed with a filtration $\mathbb{F} = (\mathcal{F}_t)_{t \geq 0}$ satisfying the usual conditions ($\mathbb{P}$-completeness and right-continuity) and let $w = (w_t)_{t \geq 0}$ be an m-dimensional standard Brownian motion, adapted to $\mathbb{F}$. Finally, denote by $\mathbb{E}$ the expectation under $(\Omega, \mathcal{F}, \mathbb{P})$.

Fix $T > 0$. We consider a finite-horizon stochastic control problem. Let $A \subseteq \mathbb{R}^k$ and denote by $\mathcal{A}$ the set of admissible controls that are assumed to be A-valued $\mathbb{F}$-progressively measurable processes $\alpha : [0, T] \times \Omega \to A$. The evolution of a controlled d-dimensional state is governed by a stochastic differential equation, with measurable coefficients $b : [0, T] \times \mathbb{R}^d \times A \to \mathbb{R}^d$ and

$\sigma : [0,T] \times \mathbb{R}^d \times A \to \mathbb{R}^{d\times m}$. For an initial condition $(t,x) \in [0,T] \times \mathbb{R}^d$, we consider the state equation given by:

$$\begin{cases} \mathrm{d}y_s = b(s, y_s, \alpha_s)\,\mathrm{d}s + \sigma(s, y_s, \alpha_s)\,\mathrm{d}w_s, \quad s \in (t,T] \\ y_t = x. \end{cases} \tag{1}$$

We denote by $y^{t,x,\alpha}$ the corresponding state process starting from (t,x) and driven by the control $\alpha \in \mathcal{A}$. Note that when the initial t time is strictly positive, only the restriction of α to $[t,T]$ is relevant.

Assumption 2.1. The following conditions hold.

(i) A is a non-empty compact and convex subset of $\mathbb{R}^k$.

(ii) For any $t,s \in [0,T]$, $x,y \in \mathbb{R}^d$, and $a,a' \in A$,

$$|b(t,x,a) - b(s,y,a')| \le [b]_0|t-s|^{1/2} + [b]_1|x-y| + [b]_2|a-a'|,$$

for some non-negative constants $[b]_0, [b]_1, [b]_2$;

(iii) For any $t,s \in [0,T]$, $x,y \in \mathbb{R}^d$, and $a,a' \in A$,

$$|\sigma(t,x,a) - \sigma(s,y,a')| \le [\sigma]_0|t-s|^{1/2} + [\sigma]_1|x-y| + [\sigma]_2|a-a'|,$$

for some non-negative constants $[\sigma]_0, [\sigma]_1, [\sigma]_2$.

Notice that, since the horizon is finite and A is compact, the coefficients have linear growth uniformly in (s,a). More precisely, there exists a positive constant C such that

$$|b(s,x,a)| + |\sigma(s,x,a)| \le C(1+|x|), \qquad (s,x,a) \in [0,T] \times \mathbb{R}^d \times A.$$

Consequently, for every $(t,x) \in [0,T] \times \mathbb{R}^d$ and every $\alpha \in \mathcal{A}$, equation (1) admits a unique strong solution, as a standard consequence of the classical well-posedness theory for SDEs; see, for instance, [52, Ch.s 1-2].

Remark 2.2. The Lipschitz continuity of b and σ with respect to the control variable is not needed for the existence and uniqueness of the solution to (1). For that purpose, it is enough to assume uniform Lipschitz continuity in the state variable and a uniform linear growth condition. We impose Lipschitz continuity with respect to the control variable because it will later be used to derive quantitative error estimates.

To introduce the control problem, let $f : [0,T] \times \mathbb{R}^d \times A \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$ be measurable functions satisfying the following assumptions.

Assumption 2.3.

(i) There exist non-negative constants $[f]_0, [f]_1, [f]_2$ such that, for all $t,s \in [0,T]$, $x,y \in \mathbb{R}^d$, and $a,a' \in A$,

$$|f(t,x,a) - f(s,y,a')| \le [f]_0|t-s|^{1/2} + [f]_1|x-y| + [f]_2|a-a'|.$$

(ii) There exists a non-negative constant $[g]$ such that, for all $x,y \in \mathbb{R}^d$,

$$|g(x) - g(y)| \le [g]|x-y|.$$

For any $(t,x) \in [0,T] \times \mathbb{R}^d$ and any admissible control $\alpha \in \mathcal{A}$ we define the cost functional

$$J(t,x,\alpha) := \mathbb{E}\Big[\int_t^T f(s, y_s^{t,x,\alpha}, \alpha_s)\,\mathrm{d}s + g(y_T^{t,x,\alpha})\Big],$$

and the corresponding value function by

$$v(t,x) := \inf_{\alpha \in \mathcal{A}} J(t,x,\alpha).$$

We recall the following regularity estimate for the value function. It follows from the continuity property of the coefficients and standard estimates for stochastic differential equations (cf. [52, Ch. 1, Thm. 6.16 and Ch. 4, Prop. 3.1]).

Proposition 2.4. *Under Assumptions 2.1 and 2.3, the value function v is well-defined on $[0,T] \times \mathbb{R}^d$ and satisfies the bound*

$$|v(t,x) - v(s,y)| \le C(|x-y| + (1+|x|+|y|)|t-s|^{1/2}), \quad \forall\, t,s \in [0,T],\ x,y \in \mathbb{R}^d,$$

for some constant $C > 0$, depending only on T, and the constants appearing in the above assumptions.

For simplicity of exposition, we assume $t = 0$ from now on. Thus, the control problem is posed on the whole interval $[0,T]$; in this case, the process $(y_s)_{s\in[0,T]}$ is denoted by $y_s^{x,\alpha} := y_s^{0,x,\alpha}$.

3 Discrete-time setting and feedback control formulation

3.1 Discrete-time setting

We approach the discretization of the control problem in two steps: given a time grid, we first restrict the admissible controls to piecewise-constant policies, and then discretize the dynamics. Let $N \in \mathbb{N}\backslash\{0\}$ and define the time-step $\tau := T/N > 0$, so that the resulting time mesh is given by $t_k := k\tau$ for $k = 0, \dots, N$. We define

$$\mathcal{A}^\tau := \Big\{ \alpha \in \mathcal{A} \;\Big|\; \exists\,(a_0,\dots,a_{N-1}) \in \mathcal{M}(\Omega;A)^N,\ a_k \text{ are } \mathcal{F}_{t_k}\text{-measurable},$$
$$\text{such that } \alpha_s(\omega) = \sum_{k=0}^{N-1} a_k(\omega)\mathbf{1}_{s\in[t_k,t_{k+1})} \Big\}$$

(with the convention that $\alpha_T(\omega) = a_{N-1}(\omega)$). Thus, any control $\alpha \in \mathcal{A}^\tau$ can be written as

$$\alpha(\omega) = (a_0(\omega), \dots, a_{N-1}(\omega)), \quad \text{where each } a_k \text{ is } A\text{-valued, } \mathcal{F}_{t_k}\text{-measurable.}$$

We first define the value $v^\tau(t,x)$ by considering piecewise-constant policies, as follows:

$$v^\tau(0,x) := \inf_{\alpha\in\mathcal{A}^\tau} J(0,x,\alpha) = \inf_{\alpha\in\mathcal{A}^\tau} \mathbb{E}\Big[\int_0^T f(s, y_s^{x,\alpha}, \alpha_s)\,\mathrm{d}s + g(y_T^{x,\alpha})\Big].$$

It is clear that $v(0,x) \le v^\tau(0,x)$. The reverse bound quantifies the approximation of general controls by piecewise-constant policies: this question was first addressed by Krylov [32], who obtained, for bounded coefficients, an error of order $\tau^{1/6}$, subsequently improved to $C\tau^{1/4}$ in [30]. In our setting, with possibly unbounded (Lipschitz) coefficients, the arguments in [3] yield $v^\tau(0,x) \le v(0,x) + C(1+|x|)^q\,\tau^{1/4}$ for a positive constant C and some $q > 0$.

We then consider the discretization of the dynamics. For any given $\alpha \in \mathcal{A}^\tau$, we discretize the dynamics (1) of $y^{x,\alpha}$ by means of the Euler–Maruyama recursive relation

$$\begin{cases} X_{t_{k+1}}^{x,\alpha} &= X_{t_k}^{x,\alpha} + b(t_k, X_{t_k}^{x,\alpha}, a_k)\tau + \sigma(t_k, X_{t_k}^{x,\alpha}, a_k)\Delta W_{k+1}, \quad k = 0,\dots,N-1, \\ X_{t_0}^{x,\alpha} &= x, \end{cases}$$

where $\Delta W_{k+1} := w_{t_{k+1}} - w_{t_k}$ are i.i.d. random variables with distribution $\mathcal{N}(0, \tau\, \mathrm{I}_m)$. We then obtain a second approximation of the value function, hereafter denoted by V_0:

$$V_0(x) := \inf_{\alpha \in \mathcal{A}^\tau} \tilde{J}(0, x, \alpha) \tag{2}$$

where

$$\tilde{J}(0, x, \alpha) := \mathbb{E}\Big[\sum_{k=0}^{N-1} f(t_k, X_{t_k}^{x,\alpha}, a_k)\tau + g(X_{t_N}^{x,\alpha}) \Big]. \tag{3}$$

We also have that there exists a positive constant C such that $|v^\tau(0,x) - V_0(x)| \le C(1+|x|)\tau^{1/2}$ (cf. [39, Prop. 3.1]; see also [31, Thm. 10.2.2] for the classical strong order $1/2$ convergence of the Euler–Maruyama scheme). Therefore, from the triangular inequality, we conclude that there exists a positive constant C such that, with $q' := \max\{q, 1\}$,

$$|v(0,x) - V_0(x)| \le C(1+|x|)^{q'}\tau^{1/4}.$$

In the next sections, we deal with the approximation of the value V_0. Indeed, the focus of this work is the approximation of V_0 by deep neural networks: roughly speaking, denoting the approximation by $\hat{V}_0$, we are interested in providing explicit error estimates, in a weak sense, between $\hat{V}_0$ and V_0, allowing for possibly degenerate diffusions.

3.2 Feedback control formulation and algorithm

We first recall that the discrete-time control problem associated with V_0 admits an equivalent Markov feedback formulation. We denote by $\mathcal{G}$ the set of measurable functions from $\mathbb{R}^d$ to A, i.e.,

$$\mathcal{G} := \mathcal{M}(\mathbb{R}^d; A).$$

For a given feedback control policy $\alpha \in \mathcal{G}^N$, that is, $\alpha = (\alpha_0, \dots, \alpha_{N-1})$, where each α_k belongs to $\mathcal{G}$, and given $x \in \mathbb{R}^d$, we consider the process $\widetilde{X}_{t_k}^{x,\alpha}$ expressed by the recursive relation, for $k = 0, \dots, N-1$,

$$\begin{cases} \widetilde{X}_{t_{k+1}}^{x,\alpha} &= \widetilde{X}_{t_k}^{x,\alpha} + b(t_k, \widetilde{X}_{t_k}^{x,\alpha}, \alpha_k(\widetilde{X}_{t_k}^{x,\alpha}))\tau + \sigma(t_k, \widetilde{X}_{t_k}^{x,\alpha}, \alpha_k(\widetilde{X}_{t_k}^{x,\alpha}))\sqrt{\tau} Z_{k+1}, \\ \widetilde{X}_{t_0}^{x,\alpha} &= x, \end{cases} \tag{4}$$

where Z_{k+1} are i.i.d. standard normal random vectors in $\mathbb{R}^m$. By the finite-horizon dynamic programming principle (DPP), the value V_0 can be equivalently written by restricting the minimization to nonrandomized Markov feedback policies (i.e., $\alpha \in \mathcal{G}^N$), then

$$V_0(x) = \inf_{\alpha \in \mathcal{G}^N} \mathbb{E}\Big[\sum_{k=0}^{N-1} f(t_k, \widetilde{X}_{t_k}^{x,\alpha}, \alpha_k(\widetilde{X}_{t_k}^{x,\alpha}))\tau + g(\widetilde{X}_{t_N}^{x,\alpha}) \Big].$$

Moreover, under the compactness of A and the above continuity assumptions on the coefficients, an optimal feedback policy exists (see [7, Prop. 8.1, 8.2, 8.5 and Cor. 8.1.1]).

We recall the following commutation formula, also known as a "randomization formula". Let $(\Omega_0, \mathcal{F}_0, \mathbb{P}_0)$ be an additional probability space, under which the expectation is indicated by $\mathbb{E}_0$, and let $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$, that is $\mathbb{E}_0[|\xi|^2] < \infty$. We consider $\widetilde{X}^{\xi,\alpha}$ on the product probability space $(\Omega_0 \times \Omega, \mathcal{F}_0 \otimes \mathcal{F}, \mathbb{P}_0 \otimes \mathbb{P})$, with initial condition $\widetilde{X}_{t_0}^{\xi,\alpha} = \xi$.

Proposition 3.1 (Randomization formula). *Let Assumptions 2.1 and 2.3 hold true, and let $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$. Then,*

1. *The following commutation formula holds:*

$$\begin{aligned}\mathbb{E}_0[V_0(\xi)] &= \inf_{\alpha\in\mathcal{G}^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)] \\ &= \inf_{\alpha\in\mathcal{G}^N} \mathbb{E}_0\Big[\mathbb{E}\Big[\sum_{k=0}^{N-1} f(t_k, \widetilde{X}^{\xi,\alpha}_{t_k}, \alpha_k(\widetilde{X}^{\xi,\alpha}_{t_k}))\tau + g(\widetilde{X}^{\xi,\alpha}_{t_N})\Big]\Big].\end{aligned} \tag{5}$$

2. *There exists an optimal feedback policy $\bar{\alpha} \in \mathcal{G}^N$ minimizing (5). Moreover, for any $\bar{\alpha} \in \mathcal{G}^N$, the following two are equivalent:*

   (a) $\bar{\alpha} \in \operatorname{arginf}_{\alpha\in\mathcal{G}^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)]$,

   (b) $\tilde{J}(0,x,\bar{\alpha}) = V_0(x)$ *(i.e.,* $\bar{\alpha} \in \operatorname{arginf}_{\alpha\in\mathcal{G}^N} \tilde{J}(0,x,\alpha)$*), for $\mathcal{L}(\xi)$-a.e. $x \in \mathbb{R}^d$, where $\mathcal{L}(\xi)$ denotes the law of ξ.*

Proof. By the DPP and a measurable selection argument, using the continuity of the coefficients and of the costs with respect to the control variable, together with the compactness of A (see [7]), there exists an optimal feedback policy $\alpha^* \in \mathcal{G}^N$ such that $V_0(x) = \tilde{J}(0,x,\alpha^*)$, for every $x \in \mathbb{R}^d$. By the definition of V_0 as the infimum over piecewise-constant policies, we get $V_0(x) \le \tilde{J}(0,x,\alpha)$, for $\alpha \in \mathcal{G}^N$, and integrating with respect to $\mathcal{L}(\xi)$ gives $\mathbb{E}_0[V_0(\xi)] \le \inf_{\alpha\in\mathcal{G}^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)]$. The reverse inequality follows by choosing $\alpha = \alpha^*$.

For the characterization of minimizers, note that $\tilde{J}(0,x,\alpha) - V_0(x) \ge 0$, thus $\mathbb{E}_0[\tilde{J}(0,\xi,\bar{\alpha}) - V_0(\xi)] = 0$ if and only if $\tilde{J}(0,x,\bar{\alpha}) = V_0(x)$ for $\mathcal{L}(\xi)$-a.e. $x \in \mathbb{R}^d$. □

In this work, we are interested in approximating the value $\mathbb{E}_0[V_0(\xi)]$ and a corresponding optimal feedback control strategy. Given an approximation space $\hat{\mathcal{G}}$ for $\mathcal{G}$ (with $\hat{\mathcal{G}} \subseteq \mathcal{G}$), such as a neural network class, we consider the following approximation procedure.

Algorithm 3.2. Let $\eta > 0$ be a given margin of error.

(i) Compute an η-suboptimal minimizer $\hat{\alpha} \in \hat{\mathcal{G}}^N$ of the above expectation formula, in the sense that

$$\mathbb{E}_0[\tilde{J}(0,\xi,\hat{\alpha})] \le \inf_{\alpha\in\hat{\mathcal{G}}^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)] + \eta. \tag{6}$$

(ii) Define the approximate value induced by $\hat{\alpha}$ as

$$\hat{V}_0(x) := \tilde{J}(0,x,\hat{\alpha}).$$

Remark 3.3. The approximate control $\hat{\alpha}$ is not necessarily unique.

Remark 3.4. Later on, we will choose $\hat{\alpha} \in \hat{\mathcal{G}}_L^N$, where $\hat{\mathcal{G}}_L$ is a class of neural networks with Lipschitz constant bounded by L. Step (i) can be performed, up to the tolerance η, by using a stochastic gradient descent algorithm (SGD).

This procedure reduces the approximation of $\mathbb{E}_0[V_0(\xi)]$ to an optimization problem over the restricted class $\hat{\mathcal{G}}^N$. The goal of the next section is to estimate the error between the induced approximation $\hat{V}_0$ and the true value V_0 in some averaged norm, in terms of the expressive power of $\hat{\mathcal{G}}$ (that is, how close $\hat{\mathcal{G}}$ is to the true space $\mathcal{G}$), and the optimization tolerance η.

4 Convergence analysis

Let $L \ge 0$ be a given constant and denote by $\mathcal{G}_L \subseteq \mathcal{G}$ the set of measurable functions $\alpha : \mathbb{R}^d \to A$ whose Lipschitz constant is bounded above by L, that is,

$$\mathcal{G}_L := \{\alpha \in \mathcal{G} \,:\, [\alpha] \le L\}.$$

The first result is a discrete Gronwall-type estimate for the second moment of the difference between two controlled trajectories.

Lemma 4.1 (Gronwall estimate). *Under assumption 2.1, given $\bar{\alpha} \in \mathcal{G}^N$, $\alpha \in \mathcal{G}_L^N$, and $x, y \in \mathbb{R}^d$, it holds*

$$\max_{0\le k\le N} \mathbb{E}\Big[\big|\widetilde{X}_{t_k}^{y,\alpha} - \widetilde{X}_{t_k}^{x,\bar{\alpha}}\big|^2\Big] \le e^{C_{1,L}T}\Big(|y-x|^2 + C_{2,L}\tau \sum_{k=0}^{N-1} \mathbb{E}\Big[\big|\alpha_k(\widetilde{X}_{t_k}^{x,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{x,\bar{\alpha}})\big|^2\Big]\Big), \tag{7}$$

with constants $C_{1,L}, C_{2,L} \ge 0$ which depend only on T, L and the data.

Proof. For $0 \le k \le N$, let $y_k := \widetilde{X}_{t_k}^{y,\alpha}$ and $x_k := \widetilde{X}_{t_k}^{x,\bar{\alpha}}$, which satisfy the recursive equations

$$\begin{aligned} y_{k+1} &= y_k + b_k(y_k, \alpha_k(y_k))\tau + \sigma_k(y_k, \alpha_k(y_k))\sqrt{\tau} Z_{k+1}, \\ x_{k+1} &= x_k + b_k(x_k, \bar{\alpha}_k(x_k))\tau + \sigma_k(x_k, \bar{\alpha}_k(x_k))\sqrt{\tau} Z_{k+1}, \end{aligned}$$

where $b_k(x,a) := b(t_k, x, a)$, $\sigma_k(x,a) := \sigma(t_k, x, a)$, and $\{Z_i\}_{i=1}^N$ are i.i.d. standard normal random vectors in $\mathbb{R}^m$. We first write

$$\begin{aligned} y_{k+1} - x_{k+1} = y_k - x_k &+ \big(b_k(y_k, \alpha_k(y_k)) - b_k(x_k, \bar{\alpha}_k(x_k))\big)\tau \\ &+ \big(\sigma_k(y_k, \alpha_k(y_k)) - \sigma_k(x_k, \bar{\alpha}_k(x_k))\big)\sqrt{\tau} Z_{k+1}. \end{aligned}$$

Taking the conditional expectation with respect to $\mathcal{F}_{t_k}$ and using $\mathbb{E}[Z_{k+1}] = 0$ and $\mathbb{E}[|\Pi Z_{k+1}|^2 \mid \mathcal{F}_{t_k}] = |\Pi|^2$ for every $\mathcal{F}_{t_k}$-measurable matrix Π, where $|\cdot|$ denotes Frobenius norm on matrices, we deduce

$$\begin{aligned} \mathbb{E}[|y_{k+1} - x_{k+1}|^2 \mid \mathcal{F}_{t_k}] = |y_k - x_k + (b_k(y_k, \alpha_k(y_k)) - b_k(x_k, \bar{\alpha}_k(x_k)))\tau|^2 \\ + \tau|\sigma_k(y_k, \alpha_k(y_k)) - \sigma_k(x_k, \bar{\alpha}_k(x_k))|^2 \end{aligned} \tag{8}$$

Let $C_b := [b]_1 + L[b]_2$ and $C_\sigma := [\sigma]_1 + L[\sigma]_2$. The last term of Equation (8) can be bounded as follows:

$$\begin{aligned} &|\sigma_k(y_k, \alpha_k(y_k)) - \sigma_k(x_k, \bar{\alpha}_k(x_k))| \\ &\qquad \le |\sigma_k(y_k, \alpha_k(y_k)) - \sigma_k(x_k, \alpha_k(x_k))| + |\sigma_k(x_k, \alpha_k(x_k)) - \sigma_k(x_k, \bar{\alpha}_k(x_k))| \\ &\qquad \le C_\sigma |y_k - x_k| + [\sigma]_2 |\alpha_k(x_k) - \bar{\alpha}_k(x_k)|. \end{aligned}$$

Hence

$$|\sigma_k(y_k, \alpha_k(y_k)) - \sigma_k(x_k, \bar{\alpha}_k(x_k))|^2 \le 2C_\sigma^2 |y_k - x_k|^2 + 2[\sigma]_2^2 |\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2. \tag{9}$$

Similarly,

$$|y_k - x_k + (b_k(y_k, \alpha_k(y_k)) - b_k(x_k, \bar{\alpha}_k(x_k)))\tau| \le (1 + C_b\tau)|y_k - x_k| + [b]_2\tau|\alpha_k(x_k) - \bar{\alpha}_k(x_k)|,$$

and we get

$$\begin{aligned} &|y_k - x_k + (b_k(y_k, \alpha_k(y_k)) - b_k(x_k, \bar{\alpha}_k(x_k)))\tau|^2 \\ &\le (1 + C_b\tau)^2 |y_k - x_k|^2 + [b]_2^2\tau^2 |\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2 + [b]_2(1 + C_b\tau)\tau(|y_k - x_k|^2 + |\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2) \\ &\le ((1 + C_b\tau)^2 + [b]_2(1 + C_b\tau)\tau)|y_k - x_k|^2 + ([b]_2^2\tau^2 + [b]_2(1 + C_b\tau)\tau)|\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2 \\ &\le (1 + \tau(TC_b^2 + 2C_b + [b]_2(1 + C_bT)))|y_k - x_k|^2 + \tau([b]_2^2 T + [b]_2(1 + C_bT))|\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2, \end{aligned}$$

where, in the last inequality, we used the fact that $\tau \le T$. We then have

$$\begin{aligned} &|y_k - x_k + (b_k(y_k, \alpha_k(y_k)) - b_k(x_k, \bar{\alpha}_k(x_k)))\tau|^2 \\ &\qquad \le (1 + B_{1,L}\tau)|y_k - x_k|^2 + \tau B_{2,L}|\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2 \end{aligned} \tag{10}$$

with constants $B_{1,L} := TC_b^2 + 2C_b + [b]_2(1 + C_bT)$ and $B_{2,L} := [b]_2^2 T + [b]_2(1 + C_bT)$.

Combining estimates (9) and (10) into Equation (8) we obtain:

$$\mathbb{E}[|y_{k+1} - x_{k+1}|^2 \,|\, \mathcal{F}_{t_k}] \le |y_k - x_k|^2(1 + C_{1,L}\tau) + C_{2,L}\tau|\alpha_k(x_k) - \bar{\alpha}_k(x_k)|^2,$$

with $C_{1,L} := B_{1,L} + 2C_\sigma^2$ and $C_{2,L} := B_{2,L} + 2[\sigma]_2^2$. By induction and the tower property of conditional expectation, for $0 < n \le N$, we get

$$\mathbb{E}\Big[|\widetilde{X}^{y,\alpha}_{t_n} - \widetilde{X}^{x,\bar{\alpha}}_{t_n}|^2\Big] \le (1 + C_{1,L}\tau)^n \Big(|y - x|^2 + C_{2,L}\tau \sum_{k=0}^{n-1} \mathbb{E}\Big[|\alpha_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k}) - \bar{\alpha}_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k})|^2\Big] \Big).$$

By noting that $(1 + C_{1,L}\tau)^n \le e^{C_{1,L}T}$, for $t_n = n\tau \le T$, we obtain the desired result. □

Remark 4.2. The estimate is asymmetric with respect to the two controls: only α is required to be Lipschitz, while the reference control $\bar{\alpha}$ is merely measurable. This comes from decomposing the difference of the controls along the two trajectories as

$$\alpha_k(y_k) - \bar{\alpha}_k(x_k) = (\alpha_k(y_k) - \alpha_k(x_k)) + (\alpha_k(x_k) - \bar{\alpha}_k(x_k)),$$

so that only the first term is controlled by the Lipschitz constant of α_k, whereas the second one is the approximation error along the trajectory driven by the reference control $\bar{\alpha}$.

We next derive an estimate for the sum term appearing on the right-hand side of Equation (7). We first have the following bound.

Lemma 4.3. *Let assumption 2.1 hold and $x \in \mathbb{R}^d$. Let $\bar{\alpha} \in \mathcal{G}^N$ and $\alpha \in \mathcal{G}_L^N$. For every Borel set $D \subseteq \mathbb{R}^d$, we have*

$$\max_{0\le k\le N} \mathbb{E}\Big[|\widetilde{X}^{x,\alpha}_{t_k} - \widetilde{X}^{x,\bar{\alpha}}_{t_k}|^2\Big] \le C_{T,L}\Big(\tau \sum_{k=0}^{N-1} \|\alpha_k - \bar{\alpha}_k\|^2_{\infty,D} + T\,\mathrm{diam}(A)^2 \max_{0\le k\le N-1} \mathbb{P}[\widetilde{X}^{x,\bar{\alpha}}_{t_k} \notin D] \Big), \tag{11}$$

where $C_{T,L}$ is a positive constant depending only on T, L and the data, and $\mathrm{diam}(A)$ denotes the diameter of A, which is finite since A is assumed compact. Moreover, by $\|\cdot\|_{\infty,D}$ we denote $\|\phi\|_{\infty,D} := \sup_{z\in D} |\phi(z)|$.

Proof. By Lemma 4.1, we have

$$\max_{0\le k\le N} \mathbb{E}\Big[|\widetilde{X}^{x,\alpha}_{t_k} - \widetilde{X}^{x,\bar{\alpha}}_{t_k}|^2\Big] \le C_{2,L} e^{C_{1,L}T}\, \tau \sum_{k=0}^{N-1} \mathbb{E}\Big[|\alpha_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k}) - \bar{\alpha}_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k})|^2\Big].$$

Let $\beta_k := \alpha_k - \bar{\alpha}_k$, so that $|\beta_k(x)| \le \mathrm{diam}(A)$ (because both α_k and $\bar{\alpha}_k$ take their values in A). By separating the regions $\{\widetilde{X}^{x,\bar{\alpha}}_{t_k} \in D\}$ and $\{\widetilde{X}^{x,\bar{\alpha}}_{t_k} \notin D\}$, we have

$$\begin{aligned} |\beta_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k})|^2 &\le \|\beta_k\|^2_{\infty,D} \cdot \mathbf{1}_{\{\widetilde{X}^{x,\bar{\alpha}}_{t_k} \in D\}} + \mathrm{diam}(A)^2 \cdot \mathbf{1}_{\{\widetilde{X}^{x,\bar{\alpha}}_{t_k} \notin D\}} \\ &\le \|\beta_k\|^2_{\infty,D} + \mathrm{diam}(A)^2 \cdot \mathbf{1}_{\{\widetilde{X}^{x,\bar{\alpha}}_{t_k} \notin D\}}. \end{aligned}$$

Thus, taking the expectation

$$\mathbb{E}[|\beta_k(\widetilde{X}^{x,\bar{\alpha}}_{t_k})|^2] \le \|\beta_k\|^2_{\infty,D} + \mathrm{diam}(A)^2 \cdot \mathbb{P}[\widetilde{X}^{x,\bar{\alpha}}_{t_k} \notin D]. \tag{12}$$

This proves the claim, with $C_{T,L} := C_{2,L} e^{C_{1,L}T}$. □

We recall the following classical estimate, in the discrete-time case.

Lemma 4.4. *Let assumption 2.1 hold true and $x \in \mathbb{R}^d$. For every $\alpha \in \mathcal{G}^N$, and for any $p \geq 2$,*

$$\mathbb{E}\Big[\max_{0\leq k\leq N} |\widetilde{X}^{x,\alpha}_{t_k} - x|^p\Big] \leq C_{T,p}(1+|x|^p),$$

where $C_{T,p}$ is a constant that depends only on T, p, and on the data.

Proof. The proof follows from the regularity assumptions on b and σ, the discrete-time Burkholder-Davis-Gundy inequality, and a discrete Gronwall argument. See, for instance, [31, Thm. 4.5.4], or [52, Ch. 1, Thm. 6.16] in the continuous-time setting. For completeness, we report the proof in Section A.1. □

We then obtain the following intermediate estimate.

Lemma 4.5. *Let assumption 2.1 hold. Let $R > 0$ and let $\xi \in L^p((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$ for some $p \geq 2$ be such that $|\xi| \leq R$, $\mathbb{P}_0$-a.s. Let $M > R$ and set $B_M := B(0, M)$. Then, for every $\alpha \in \mathcal{G}^N$, there exists a constant $C_{T,p} \geq 0$, depending only on T, p, and the data, such that*

$$\mathbb{E}_0\Big[\max_{0\leq k\leq N} \mathbb{P}\Big[\widetilde{X}^{\xi,\alpha}_{t_k} \notin B_M\Big]\Big] \leq \frac{C_{T,p}}{(M-R)^p}(1+\mathbb{E}_0[|\xi|^p]).$$

Proof. By the estimate in Lemma 4.4, applied conditionally on ξ, for $\mathbb{P}_0$-a.e. realization of ξ, there exists a positive constant $C_{T,p}$ such that

$$\mathbb{E}\Big[\max_{0\leq k\leq N} |\widetilde{X}^{\xi,\alpha}_{t_k} - \xi|^p\Big] \leq C_{T,p}(1+|\xi|^p).$$

Since $|\xi| \leq R$ and $M > R$, the implication

$$\widetilde{X}^{\xi,\alpha}_{t_k} \notin B_M \quad \Longrightarrow \quad |\widetilde{X}^{\xi,\alpha}_{t_k} - \xi| \geq M - |\xi| \geq M - R$$

holds for every k. Hence,

$$\begin{aligned}\max_{0\leq k\leq N} \mathbb{P}\Big[\widetilde{X}^{\xi,\alpha}_{t_k} \notin B_M\Big] &\leq \mathbb{P}\Big[\max_{0\leq k\leq N} |\widetilde{X}^{\xi,\alpha}_{t_k} - \xi| \geq M - R\Big] \\ &\leq \frac{\mathbb{E}\big[\max_{0\leq k\leq N} |\widetilde{X}^{\xi,\alpha}_{t_k} - \xi|^p\big]}{(M-R)^p} \leq \frac{C_{T,p}}{(M-R)^p}(1+|\xi|^p),\end{aligned}$$

where we used Markov's inequality. The result follows by taking the $\mathbb{E}_0$-expectation. □

We can now state our main result.

Theorem 4.6 (Error estimate). *Assume that assumptions 2.1 and 2.3 hold. Let $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$, such that $|\xi| \leq R$, $\mathbb{P}_0$-a.s., for some $R \geq 0$. Let $\varepsilon > 0$ and let $\bar{\alpha}^\varepsilon \in \mathcal{G}^N$ be an ε-optimal feedback policy for the randomized problem related to $V_0(\xi)$, namely $\mathbb{E}_0[\tilde{J}(0,\xi,\bar{\alpha}^\varepsilon)] \leq \mathbb{E}_0[V_0(\xi)] + \varepsilon$. Let $L \geq 0$ and let $\hat{\mathcal{G}}_L \subseteq \mathcal{G}_L$ be the approximation class. For any $\eta > 0$, consider an η-suboptimal control policy $\hat{\alpha} \in \hat{\mathcal{G}}_L^N$ obtained as in Algorithm 3.2, and define $\hat{V}_0(x) := \tilde{J}(0, x, \hat{\alpha})$. Then there exists a positive constant $C_{T,L}$, depending only on T, L and the data, and independent of N, such that, for any $M > R$,*

$$\begin{aligned}&\mathbb{E}_0[|\hat{V}_0(\xi) - V_0(\xi)|] \\ &\qquad \leq C_{T,L}\left[\left(\tau \sum_{0\leq k\leq N-1} d^2_{\infty,B_M}(\bar{\alpha}^\varepsilon_k, \hat{\mathcal{G}}_L)\right)^{1/2} + \frac{\operatorname{diam}(A)}{M-R}(1+\mathbb{E}_0[|\xi|^2])^{1/2}\right] + \varepsilon + \eta,\end{aligned}$$

where $d_{\infty,B_M}(\phi, \hat{\mathcal{G}}_L) := \inf_{\psi\in\hat{\mathcal{G}}_L} \sup_{z\in B_M} |\phi(z) - \psi(z)|$.

Remark 4.7 (Error terms interpretation). On the right-hand side of the above error estimate:

- The first term measures how well the reference policy $\bar{\alpha}^\varepsilon$ can be approximated, on the localization ball B_M, by L-Lipschitz policies in the neural network class $\hat{\mathcal{G}}_L \subseteq \mathcal{G}_L$.
- The estimate holds for any ε-optimal $\bar{\alpha}^\varepsilon \in \mathcal{G}^N$, but is informative only when $\bar{\alpha}^\varepsilon$ can be chosen with Lipschitz components: for a merely measurable policy, such as a discontinuous bang-bang feedback, the first term may remain of order $\mathrm{diam}(A)$. The existence of such Lipschitz ε-optimal policies is the object of Proposition 4.11 below.
- The parameter ε measures the suboptimality of the chosen reference policy $\bar{\alpha}^\varepsilon$. Proposition 4.11 below shows that, for every $\varepsilon > 0$, one can choose an ε-optimal reference policy in $\mathcal{G}_L^N$, for a suitable L. In the convergence proof, ε will be used precisely in this sense: it measures the cost of replacing measurable feedback policies by Lipschitz feedback policies. In particular, if an optimal feedback policy belongs to $\mathcal{G}_L^N$, then one may take $\varepsilon = 0$.
- The term proportional to $(M - R)^{-1}$ accounts for the probability that the stochastic process, starting in B_R, leaves the localization ball B_M over the time horizon T. In deterministic settings, such as in [9, 10], this localization term can often be made zero by choosing B_M large enough to contain the reachable set over the time horizon. In the stochastic case, one instead controls the probability of leaving B_M.
- The term η accounts for the optimization error in the computation of the approximate policy $\hat{\alpha}$, (e.g., the convergence of an SGD algorithm).

Remark 4.8 (Dependence on the Lipschitz constant). The constants in the stability estimate have the form $C_{T,L} = C_{2,L} \exp(C_{1,L} T)$, with $C_{1,L}$ containing quadratic terms in L, hence $C_{T,L}$ may grow exponentially as L increases. This dependence should be kept in mind when interpreting the approximation term and the parameter ε. If an optimal, or near-optimal, feedback policy is Lipschitz with a moderate Lipschitz constant, then one may choose the reference policy with fixed L, and the estimate yields a meaningful quantitative bound. For example, in the linear–convex setting of [21], under additional structural assumptions, the existence of Lipschitz continuous optimal feedback controls is proved, together with stability properties. On the other hand, when the optimal feedback is discontinuous, for instance of bang–bang type, making ε small may require Lipschitz approximations with increasingly large Lipschitz constants (cf. Remark 4.13). In that case, the decrease of the approximation or suboptimality error must be balanced against the growth of $C_{T,L}$.

Remark 4.9 (Degenerate dynamics and pointwise approximation). The distance d_{∞,B_M} is defined through the pointwise supremum norm on B_M, rather than a Lebesgue essential supremum or an L^p norm. This is not a technical detail: since the controlled dynamics may be degenerate or deterministic, the law of the discrete-time state process may be singular with respect to the Lebesgue measure (e.g., supported on a lower-dimensional set). Two policies that coincide Lebesgue-a.e. may then differ precisely on the support of the state process and induce different values, so that only a pointwise uniform approximation of $\bar{\alpha}_k^\varepsilon$ guarantees a control of the error, whatever the law of the state process. This is also why our estimate does not rely on the existence of a transition probability (from step $\widetilde{X}_{t_n}$ to $\widetilde{X}_{t_{n+1}}$) associated with an explicit bounded density, unlike density-based estimates such as those used in [28]. The framework therefore covers possibly degenerate diffusions and includes the purely deterministic case, where the transition law is generally not associated with a density.

Proof of Theorem 4.6. Since $\hat{\mathcal{G}}_L \subseteq \mathcal{G}$, the policy $\hat{\alpha}$ is admissible for the discrete feedback problem, hence $\hat{V}_0(\xi) - V_0(\xi) \geq 0$, $\mathbb{P}_0$-a.s., and therefore $\mathbb{E}_0[|\hat{V}_0(\xi) - V_0(\xi)|] = \mathbb{E}_0[\hat{V}_0(\xi) - V_0(\xi)]$. The argument reduces this $L^1(\mathbb{P}_0)$ error to an $L^2(\mathbb{P}_0)$ estimate for differences of cost functionals, by the Cauchy–Schwarz inequality.

Throughout the proof, we omit the superscript ε and simply write $\bar{\alpha}$ for the ε-optimal feedback control $\bar{\alpha}^\varepsilon$. By its ε-suboptimality, we have

$$\mathbb{E}_0[\hat{V}_0(\xi) - V_0(\xi)] \le \mathbb{E}_0[\hat{V}_0(\xi) - \tilde{J}(0,\xi,\bar{\alpha})] + \varepsilon,$$

which, together with the η-suboptimality of the control policy $\hat{\alpha} \in \hat{\mathcal{G}}_L^N$, leads to

$$\begin{aligned}\mathbb{E}_0[\hat{V}_0(\xi) - V_0(\xi)] &\le \inf_{\alpha\in\hat{\mathcal{G}}_L^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha) - \tilde{J}(0,\xi,\bar{\alpha})] + \varepsilon + \eta \\ &\le \inf_{\alpha\in\hat{\mathcal{G}}_L^N} (\mathbb{E}_0[|\tilde{J}(0,\xi,\alpha) - \tilde{J}(0,\xi,\bar{\alpha})|^2])^{1/2} + \varepsilon + \eta. \end{aligned} \tag{13}$$

Let $\mathcal{E}_\alpha := |\tilde{J}(0,\xi,\alpha) - \tilde{J}(0,\xi,\bar{\alpha})|^2$, where $\alpha \in \hat{\mathcal{G}}_L^N$. We aim to bound $\inf_{\alpha\in\hat{\mathcal{G}}_L^N}(\mathbb{E}_0[\mathcal{E}_\alpha])^{1/2}$. By Jensen's and Cauchy–Schwarz's inequalities:

$$\begin{aligned}\mathcal{E}_\alpha &\le \mathbb{E}\Big[\Big|\sum_{k=0}^{N-1} \tau(f(t_k,\widetilde{X}_{t_k}^{\xi,\alpha},\alpha_k(\widetilde{X}_{t_k}^{\xi,\alpha})) - f(t_k,\widetilde{X}_{t_k}^{\xi,\bar{\alpha}},\bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}))) + (g(\widetilde{X}_{t_N}^{\xi,\alpha}) - g(\widetilde{X}_{t_N}^{\xi,\bar{\alpha}}))\Big|^2\Big] \\ &\le 2\mathbb{E}\Big[N\tau^2 \sum_{k=0}^{N-1} |f(t_k,\widetilde{X}_{t_k}^{\xi,\alpha},\alpha_k(\widetilde{X}_{t_k}^{\xi,\alpha})) - f(t_k,\widetilde{X}_{t_k}^{\xi,\bar{\alpha}},\bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}))|^2\Big] + 2\mathbb{E}\Big[|g(\widetilde{X}_{t_N}^{\xi,\alpha}) - g(\widetilde{X}_{t_N}^{\xi,\bar{\alpha}})|^2\Big] \\ &\le 2T\mathbb{E}\Big[\tau \sum_{k=0}^{N-1} |f(t_k,\widetilde{X}_{t_k}^{\xi,\alpha},\alpha_k(\widetilde{X}_{t_k}^{\xi,\alpha})) - f(t_k,\widetilde{X}_{t_k}^{\xi,\bar{\alpha}},\bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}))|^2\Big] + 2\mathbb{E}\Big[|g(\widetilde{X}_{t_N}^{\xi,\alpha}) - g(\widetilde{X}_{t_N}^{\xi,\bar{\alpha}})|^2\Big].\end{aligned}$$

By considering the Lipschitz continuity assumptions of f, g and the control policies, we get

$$|f(t_k,\widetilde{X}_{t_k}^{\xi,\alpha},\alpha_k(\widetilde{X}_{t_k}^{\xi,\alpha})) - f(t_k,\widetilde{X}_{t_k}^{\xi,\bar{\alpha}},\bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}))| \le C_1\,|\widetilde{X}_{t_k}^{\xi,\alpha} - \widetilde{X}_{t_k}^{\xi,\bar{\alpha}}| + [f]_2\,|\alpha_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}})|,$$

where $C_1 := [f]_1 + L[f]_2$, and then

$$\begin{aligned}\mathcal{E}_\alpha &\le 2T\cdot\mathbb{E}\Big[\tau\sum_{k=0}^{N-1}\Big(2C_1^2\,|\widetilde{X}_{t_k}^{\xi,\alpha} - \widetilde{X}_{t_k}^{\xi,\bar{\alpha}}|^2 + 2[f]_2^2\,|\alpha_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}})|^2\Big)\Big] + 2[g]^2\mathbb{E}\Big[|\widetilde{X}_{t_N}^{\xi,\alpha} - \widetilde{X}_{t_N}^{\xi,\bar{\alpha}}|^2\Big] \\ &\le (4T^2C_1^2 + 2[g]^2)\max_{0\le k\le N}\mathbb{E}\Big[|\widetilde{X}_{t_k}^{\xi,\alpha} - \widetilde{X}_{t_k}^{\xi,\bar{\alpha}}|^2\Big] + 4T[f]_2^2\,\tau\sum_{k=0}^{N-1}\mathbb{E}\Big[|\alpha_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}})|^2\Big].\end{aligned} \tag{14}$$

We bound the term $\mathbb{E}[|\alpha_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}})|^2]$ as in (12) and obtain

$$\tau\sum_{k=0}^{N-1}\mathbb{E}\Big[|\alpha_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}}) - \bar{\alpha}_k(\widetilde{X}_{t_k}^{\xi,\bar{\alpha}})|^2\Big] \le \tau\sum_{k=0}^{N-1}\|\alpha_k - \bar{\alpha}_k\|_{\infty,B_M}^2 + T\,\mathrm{diam}(A)^2\max_{0\le k\le N-1}\mathbb{P}[\widetilde{X}_{t_k}^{\xi,\bar{\alpha}} \notin B_M]. \tag{15}$$

Combining (14), (15) and Lemma 4.3 applied conditionally on ξ, for $\mathbb{P}_0$-a.e. realization of ξ, yields

$$\mathcal{E}_\alpha \le C_{T,L}\,\Big(\tau\sum_{k=0}^{N-1}\|\alpha_k - \bar{\alpha}_k\|_{\infty,B_M}^2 + T\,\mathrm{diam}(A)^2\max_{0\le k\le N-1}\mathbb{P}[\widetilde{X}_{t_k}^{\xi,\bar{\alpha}} \notin B_M]\Big),$$

for a constant $C_{T,L}$ depending only on T, L, and the data. By integrating with respect to $\mathbb{P}_0$, we get from Lemma 4.5

$$\mathbb{E}_0[\mathcal{E}_\alpha] \le C_{T,L}\,\Big(\tau\sum_{k=0}^{N-1}\|\alpha_k - \bar{\alpha}_k\|_{\infty,B_M}^2 + T\,\mathrm{diam}(A)^2\frac{C_{T,2}}{(M-R)^2}(1+\mathbb{E}_0[|\xi|^2])\Big).$$

Finally, in the bound (13), taking the infimum over $\alpha \in \hat{\mathcal{G}}_L^N$ corresponds to

$$\inf_{\alpha \in \hat{\mathcal{G}}_L^N} \tau \sum_{k=0}^{N-1} \|\alpha_k - \bar{\alpha}_k\|_{\infty,B_M}^2 = \tau \sum_{k=0}^{N-1} d_{\infty,B_M}^2(\bar{\alpha}_k, \hat{\mathcal{G}}_L).$$

Using $\sqrt{u+v} \leq \sqrt{u}+\sqrt{v}$ for $u, v \geq 0$ to bound $(\mathbb{E}_0[\mathcal{E}_\alpha])^{1/2}$, we obtain the stated estimate (upon enlarging the $C_{T,L}$ constant if necessary). □

The previous error estimate now allows us to prove the convergence of the approximated value $\hat{V}_0$ (computed with Algorithm 3.2), towards V_0, in $L^1(\mathbb{P}_0)$ for compactly supported initial distributions.

Theorem 4.10 (Convergence). *Assume that assumptions 2.1 and 2.3 hold. Let $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$, compactly supported. Let the time grid, equivalently $N \geq 1$, be fixed. Let $(\hat{\mathcal{G}}_L^\Theta)_{\Theta \geq 1}$, with $\hat{\mathcal{G}}_L^\Theta \subseteq \mathcal{G}_L$ for every Θ, be a family of approximation classes such that, for every $L \geq 0$, every $M \geq 0$, and every L-Lipschitz function $\alpha : B_M \to A$,*

$$\lim_{\Theta \to \infty} d_{\infty,B_M}(\alpha, \hat{\mathcal{G}}_L^\Theta) = 0. \tag{16}$$

Then, for every $\delta > 0$, there exist $L \geq 0$, $\bar{\Theta} \geq 1$, and $\bar{\eta} > 0$ such that, for every $\Theta \geq \bar{\Theta}$ and every $\eta \in (0, \bar{\eta}]$, the η-suboptimal approximation $\hat{V}_0^{\Theta,\eta}$ computed by Algorithm 3.2 over $(\hat{\mathcal{G}}_L^\Theta)^N$ (that is, satisfying, as in (6), $\mathbb{E}_0[\hat{V}_0^{\Theta,\eta}] \leq \inf_{\alpha \in (\hat{\mathcal{G}}_L^\Theta)^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)] + \eta$) satisfies

$$\mathbb{E}_0[|\hat{V}_0^{\Theta,\eta}(\xi) - V_0(\xi)|] \leq \delta.$$

In other words, there exist sequences $L_j \geq 0$, $\Theta_j \to \infty$, and $\eta_j \downarrow 0$ such that

$$\mathbb{E}_0[|\hat{V}_0^{\Theta_j,\eta_j}(\xi) - V_0(\xi)|] \xrightarrow[j\to\infty]{} 0.$$

In order to prove this convergence result, we first need an intermediate result showing that the value of the discrete-time control problem can be approximated arbitrarily well by using Lipschitz-continuous feedback controls.

Proposition 4.11. *Assume that assumptions 2.1 and 2.3 hold. Let $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$. Then, for any $\varepsilon > 0$, there exists a constant $L \geq 0$ and $\bar{\alpha} \in \mathcal{G}_L^N$, such that*

$$\mathbb{E}_0[\tilde{J}(0,\xi,\bar{\alpha})] \leq \mathbb{E}_0[V_0(\xi)] + \varepsilon.$$

Remark 4.12. Equivalently, for every $\varepsilon > 0$ there exists $L \geq 0$ such that

$$\mathbb{E}_0[V_0(\xi)] \geq \inf_{\alpha \in \mathcal{G}_L^N} \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)] - \varepsilon.$$

In other words, Lipschitz feedback policies are dense at the level of the value $V_0(\xi)$, in the class of measurable feedback policies.

Proof. We prove the result by a density argument, presented in Appendix Lemma A.1 with respect to the laws of the discretized controlled process. In this proof, the convexity of $A \subset \mathbb{R}^k$ is needed to ensure that the approximation can be chosen to be A-valued.

By the randomization formula (Proposition 3.1), there exists a measurable feedback policy $\alpha = (\alpha_0, \ldots, \alpha_{N-1}) \in \mathcal{G}^N$ such that

$$\mathbb{E}_0[V_0(\xi)] \leq \mathbb{E}_0[\tilde{J}(0,\xi,\alpha)] \leq \mathbb{E}_0[V_0(\xi)] + \frac{\varepsilon}{2}, \tag{17}$$

(the left inequality is just a consequence of the definitions). We show that this policy can be approximated, in value, by a Lipschitz feedback policy.

We now regularize the components of α backward in time. Set $\alpha^{(N)} := \alpha$. Suppose that, for some $j \in \{0, \dots, N-1\}$, the feedbacks at times $j+1, \dots, N-1$ have already been replaced by Lipschitz feedbacks, while the previous ones are left unchanged. We denote the corresponding policy by

$$\alpha^{(j+1)} := (\alpha_0, \dots, \alpha_j, \bar{\alpha}_{j+1}, \dots, \bar{\alpha}_{N-1}), \qquad \text{where } \bar{\alpha}_{j+1}, \dots, \bar{\alpha}_{N-1} \text{ are Lipschitz.}$$

For a given $\bar{\alpha}_j$ (to be defined later on), let us consider the policy obtained by replacing the j-th component by $\bar{\alpha}_j$:

$$\alpha^{(j)} := (\alpha_0, \dots, \alpha_{j-1}, \bar{\alpha}_j, \bar{\alpha}_{j+1}, \dots, \bar{\alpha}_{N-1}).$$

Since the policies $\alpha^{(j)}$ and $\alpha^{(j+1)}$ coincide up to time t_j, the corresponding discretized trajectories also coincide up to time t_j. The first difference between them is produced at the step from t_j to t_{j+1}, and is controlled by Equation (19). Since the feedback controls at times $j+1, \dots, N-1$ are Lipschitz, the discrete stability estimate for the Euler scheme propagates this error from t_{j+1} to T. Consequently, using also the Lipschitz continuity of f and g, there exists a finite constant $C_j \geq 0$, depending only on the data, on T, and on the Lipschitz constants of $\bar{\alpha}_{j+1}, \dots, \bar{\alpha}_{N-1}$ (and independent neither of $\bar{\alpha}_j$ nor of $\alpha_0, \dots, \alpha_{j-1}$) such that

$$\mathbb{E}_0[|\tilde{J}(0, \xi, \alpha^{(j)}) - \tilde{J}(0, \xi, \alpha^{(j+1)})|] \leq C_j \Big(\mathbb{E}_0 \mathbb{E}\Big[\Big| \bar{\alpha}_j(\widetilde{X}_{t_j}^{\xi, \alpha^{(j+1)}}) - \alpha_j(\widetilde{X}_{t_j}^{\xi, \alpha^{(j+1)}}) \Big|^2 \Big] \Big)^{1/2}. \tag{18}$$

Let $\mu_j = \mathcal{L}(\widetilde{X}_{t_j}^{\xi, \alpha^{(j+1)}}) = \mathcal{L}(\widetilde{X}_{t_j}^{\xi, \alpha})$ denote the law under $\mathbb{P}_0 \otimes \mathbb{P}$. Applying Lemma A.1, with $\mu = \mu_j$ and $\phi = \alpha_j$, we can choose a Lipschitz continuous map $\bar{\alpha}_j : \mathbb{R}^d \to A$ such that

$$\Big(\int_{\mathbb{R}^d} |\bar{\alpha}_j(x) - \alpha_j(x)|^2 \, \mu_j(\mathrm{d}x) \Big)^{1/2} \equiv \Big(\mathbb{E}_0 \mathbb{E}\Big[\Big| \bar{\alpha}_j(\widetilde{X}_{t_j}^{\xi, \alpha^{(j+1)}}) - \alpha_j(\widetilde{X}_{t_j}^{\xi, \alpha^{(j+1)}}) \Big|^2 \Big] \Big)^{1/2} \leq \frac{\varepsilon}{2 C_j N}. \tag{19}$$

Then we have

$$\mathbb{E}_0[|\tilde{J}(0, \xi, \alpha^{(j)}) - \tilde{J}(0, \xi, \alpha^{(j+1)})|] \leq \frac{\varepsilon}{2N}.$$

The procedure is applied backward, successively for $j = N-1, N-2, \dots, 1, 0$. Suming up the bounds, and using the fact that $\alpha^{(N)} \equiv \alpha$, we obtain an approximate feedback control $\bar{\alpha} := (\bar{\alpha}_0, \dots, \bar{\alpha}_{N-1}) \in \mathcal{G}_L^N$, with $L := \max_{0 \leq k \leq N-1} [\bar{\alpha}_k]$, and such that

$$\mathbb{E}_0[|\tilde{J}(0, \xi, \bar{\alpha}) - \tilde{J}(0, \xi, \alpha)|] \leq \frac{\varepsilon}{2}.$$

Therefore, together with Equation (17), this leads to the desired estimate. □

Remark 4.13 (Non-quantitative nature of the Lipschitz approximation). The construction above is qualitative. It shows that, for every $\varepsilon > 0$, one may choose an ε-optimal feedback policy with L-Lipschitz components, but it does not provide a quantitative bound on the corresponding Lipschitz constant $L = L(\varepsilon)$, which may blow up as $\varepsilon \to 0$. Consequently, the convergence result of Theorem 4.10 does not provide a rate in terms of the size of the approximation class; we refer to Remark 4.8 for structural situations in which L can be fixed a priori, allowing for quantitative bounds.

Proof of Theorem 4.10. Let $\delta > 0$. Set $\varepsilon = \delta/4$. By Proposition 4.11, there exist $L \geq 0$ and an ε-suboptimal Lipschitz control $\bar{\alpha} \in \mathcal{G}_L^N$. Since ξ is compactly supported, there exists $R \geq 0$ such that $\xi \in B_R$, $\mathbb{P}_0$-a.s. Choose then $M > R$ large enough, such that

$$C_{T,L} \frac{\mathrm{diam}(A)}{M - R} (1 + \mathbb{E}_0[|\xi|^2])^{1/2} \leq \frac{\delta}{4}.$$

Set $\bar{\eta} := \delta/4$. By the approximation property of $(\hat{\mathcal{G}}_L^\Theta)_{\Theta\geq 1}$, there exists $\bar{\Theta} \geq 1$ such that, for every $\Theta \geq \bar{\Theta}$,

$$C_{T,L}\Big(\tau \sum_{k=0}^{N-1} d^2_{\infty,B_M}(\bar{\alpha}_k, \hat{\mathcal{G}}_L^\Theta)\Big)^{1/2} \leq \frac{\delta}{4}.$$

Applying Theorem 4.6, for any $\Theta \geq \bar{\Theta}$, $\eta \in (0, \bar{\eta}]$, we obtain $\mathbb{E}_0[|\hat{V}_0^{\Theta,\eta}(\xi) - V_0(\xi)|] \leq \delta$, which proves the convergence result. □

Remark 4.14 (Higher-order time discretizations). The convergence analysis has been presented for the Euler–Maruyama scheme to keep the notation simple, but the same approach extends to other explicit one-step discretization schemes. In particular, the discrete Gronwall estimate, the localization argument, and the policy-approximation estimate remain valid (up to different constants) for any scheme whose one-step map is stable with respect to both the state variable and the feedback control evaluations. Higher-order weak schemes could also be considered in this framework; see, e.g., [31, Chs. 14-15], [34], and [47].

Remark 4.15 (Lipschitz-constrained architectures). The approximation assumption (16) can be satisfied by known Lipschitz-constrained neural network architectures. In particular, norm-constrained GroupSort networks, introduced in [2], are universal approximators of Lipschitz functions on compact sets. Quantitative approximation results for Lipschitz functions by GroupSort networks are further developed in [48]. Therefore, the class $\hat{\mathcal{G}}_L$ appearing in the convergence theorem can be realized by existing neural-network architectures with an explicit (a priori) control of the Lipschitz constant.

In the numerical experiments below, we consider standard feedforward neural networks with smooth Lipschitz activation functions, such as `sigmoid`, `tanh`, and `SiLU`. For fixed finite parameters, these networks define Lipschitz feedback maps, although their Lipschitz constants are not explicitly a priori constrained during training. This choice is made for computational convenience.

5 Neural-network algorithm

We now describe the neural-network implementation used to approximate the time-discrete stochastic control problem associated with the value function V_0 introduced above. The method is based on a direct parametrization of the feedback policy by a neural network and on the minimization of the corresponding simulated cost.

For a given feedback control $\alpha \in \mathcal{G}^N$ and given $\xi \in L^2((\Omega_0, \mathcal{F}_0, \mathbb{P}_0); \mathbb{R}^d)$, compactly supported on a ball B_R of radius $R > 0$, we consider the stochastic process $\widetilde{X}_{t_k}^{\xi,\alpha}$ over the time grid $\{t_k := k\tau\}_{k=0}^N$, for $N \in \mathbb{N}\backslash\{0\}$ and $\tau := T/N$, generated by the time-discrete dynamics introduced in Equation (4). Here, for simplicity, we present the Euler–Maruyama scheme; the generalization to higher-order schemes follows according to Remark 4.14. In view of the commutation formula in Proposition 3.1, the training objective is the randomized cost $\mathbb{E}_0[\tilde{J}(0, \xi, \alpha)]$.

The convergence analysis of Section 4 applies to piecewise-constant feedback policies $\alpha = (\alpha_0, \ldots, \alpha_{N-1}) \in \mathcal{G}^N$, independently of the neural parametrization used to represent them. One may either use a single time-state network $\alpha^\theta(t, x)$ and set $\alpha_k = \alpha^\theta(t_k, \cdot)$ (Examples 1 and 2), or a separate network per time step, $\alpha_k = \alpha_k^{\theta_k}$ (Example 3). The former shares parameters across time steps, reduces the number of trainable parameters, and introduces an implicit regularization in time, which can be advantageous when the optimal feedback depends regularly on time. The latter yields a larger and more flexible approximation class, which can be preferable when the feedback or the admissible control constraints vary irregularly in time, at the price of a number of trainable parameters that typically grows linearly with the number of time steps. In both cases, the error estimates apply to the induced discrete policy, provided it belongs to the

approximation class of the convergence theorem. We stress that, even with time-step-specific networks, the training remains global in time: all feedback components are optimized simultaneously through the simulated total cost, rather than by a backward DPP recursion as in [4, 28].

Let Θ denote the trainable parameters of a neural parametrization and set

$$\alpha^\Theta = (\alpha_0^\Theta, \dots, \alpha_{N-1}^\Theta), \qquad \alpha_k^\Theta : \mathbb{R}^d \to A.$$

This notation covers both parametrizations discussed above (in the time-state network, $\alpha_k^\Theta(x) = \alpha^\theta(t_k, x)$ with $\Theta = \theta$, whereas in the time-step-specific network, $\alpha_k^\Theta(x) = \alpha_k^{\theta_k}(x)$ with $\Theta = (\theta_0, \dots, \theta_{N-1})$).

The expectation in the training objective is estimated using $M_{\text{batch}} > 0$ simulated trajectories for the process $\widetilde{X}^{\xi,\alpha^\Theta}$. The Monte Carlo estimator associated with the cost functional for a given parametrized policy α^Θ is then given by

$$\widehat{J}(0, \xi, \alpha^\Theta) := \frac{1}{M_{\text{batch}}} \sum_{m=1}^{M_{\text{batch}}} \left[\sum_{k=0}^{N-1} f(t_k, \widetilde{X}_{t_k,m}^{\xi,\alpha^\Theta}, \alpha_k^\Theta(\widetilde{X}_{t_k,m}^{\xi,\alpha^\Theta}))\tau + g(\widetilde{X}_{t_N,m}^{\xi,\alpha^\Theta}) \right], \tag{20}$$

where $(\widetilde{X}_{t_k,m}^{\xi,\alpha^\Theta})_{k=0}^N$ denotes the m-th sampled trajectory, starting at the point $\widetilde{X}_{t_0,m}^{\xi,\alpha^\Theta} \in B_R$, sampled from the distribution $\mathcal{L}(\xi)$.

The training of the parametrized feedback policy α^Θ updates the parameters Θ according to a stochastic gradient-based approximation method, minimizing the empirical loss (20). The resulting training procedure is summarized in Algorithm 1.

Algorithm 1: Direct neural-network policy learning

Input : Dynamics b, σ; costs f, g; step-size τ; $\mathcal{L}(\xi)$ s.t. $\text{supp}(\mathcal{L}(\xi)) \subseteq B_R$ for $R \geq 0$
Input : Control α^Θ; learning rate γ; M_{batch}; N_{epoch}
for $i = 1, \dots, N_{\text{epoch}}$ **do**
  Initialize $v^i = 0$ (cost of the batch)
  for $m = 1, \dots, M_{\text{batch}}$ **do**
    Sample $x^{i,m}$ in B_R according to $\mathcal{L}(\xi)$
    Generate i.i.d. standard Gaussian random vectors $\{Z_k^{i,m}\}_{k=1}^N$
    Initialize $v^{i,m} = 0$ (cost of the trajectory)
    for $k = 0, \dots, N-1$ **do**
      $t_k \leftarrow k\tau$
      $a_k^{i,m} \leftarrow \alpha_k^\Theta(x^{i,m})$
      $v^{i,m} \leftarrow v^{i,m} + f(t_k, x^{i,m}, a_k^{i,m})\tau$
      $x^{i,m} \leftarrow x^{i,m} + b(t_k, x^{i,m}, a_k^{i,m})\tau + \sigma(t_k, x^{i,m}, a_k^{i,m})\sqrt{\tau} Z_{k+1}^{i,m}$
    end
    $v^{i,m} \leftarrow v^{i,m} + g(x^{i,m})$
  end
  $v^i \leftarrow \frac{1}{M_{\text{batch}}} \sum_{m=1}^{M_{\text{batch}}} v^{i,m}$
  $\Theta \leftarrow \Theta - \gamma \nabla_\Theta v^i$ (update parameters, where $v^i = v^i(\Theta)$)
end
Output : $\widehat{\Theta}$ trained parameters; $\widehat{\alpha} = \alpha^{\widehat{\Theta}}$ approximate feedback policy

Remark 5.1. Although the theoretical formulation distinguishes between the auxiliary expectation $\mathbb{E}_0$, associated with the randomized initial condition ξ, and the expectation $\mathbb{E}$, associated with the noise of the controlled dynamics, the numerical implementation estimates the resulting product expectation directly. More precisely, at each training iteration we sample independent

initial conditions $\xi^m \sim \mathcal{L}(\xi)$ and, conditionally on each ξ^m, independent noise increments driving the time-discrete dynamics. Thus, from Algorithm 1, $\widehat{J}(0, \xi, \alpha^\Theta)$ is a Monte Carlo estimator of $\mathbb{E}_0[\tilde{J}(0, \xi, \alpha^\Theta)]$. Moreover, in the theoretical analysis, the optimization error is represented by the parameter η. In practice, finite-sample effects from the Monte Carlo approximation and the convergence of the stochastic optimization procedure both contribute to the observed training error.

6 Numerical examples

We present three numerical examples illustrating the proposed methodology. The experiments are designed to identify, and separate as far as possible, the different error components: the time discretization of the controlled dynamics (Examples 1 and 2, comparing the Euler–Maruyama scheme with the weak second-order scheme of Section C), the restriction to piecewise-constant policies (Example 3), and the neural-network approximation and optimization errors. In all experiments, after training, the reported values are computed by re-evaluating the trained policy on a large independent sample, thereby reducing the Monte Carlo error.

We recall that the classical weak second-order convergence theory (cf. [31, Ch. 15]) requires smooth coefficients, including the feedback policy inserted into the dynamics. In our experiments the trained network policy is smooth, but the optimal feedback may be discontinuous; the second-order behavior observed below (in the first two examples) is therefore not covered by the classical theory and should be regarded as empirical.

Computational setup. All numerical experiments were performed on a Linux GPU server using one `NVIDIA Tesla V100-PCIE-16GB` GPU per run. The server was equipped with three such GPUs and used NVIDIA driver version `535.309.01` and CUDA version `12.2`.

6.1 Example 1: Stochastic radial target problem

This first example is a benchmark target problem with a controlled SDE involving a degenerate diffusion. We aim to observe the convergence order of time discretization schemes.

The model. Let B be a standard real-valued Brownian motion. For $0 \le t \le T$, we consider the 2-dimensional process $X_s = (X_s^1, X_s^2)$, defined by the coupled system of stochastic differential equations

$$\begin{cases} \mathrm{d}X_s^1 &= -X_s^2 \mathrm{d}B_s + \alpha_s^1 \mathrm{d}s, \quad s \in (t, T] \\ \mathrm{d}X_s^2 &= \quad X_s^1 \mathrm{d}B_s + \alpha_s^2 \mathrm{d}s, \quad s \in (t, T] \\ X_t &= x \in \mathbb{R}^2, \end{cases}$$

where the control processes $\alpha_s = (\alpha_s^1, \alpha_s^2) = (\alpha^1(s, X_s), \alpha^2(s, X_s))$ are given in feedback form. When needed, we will denote $X_s^{t,x,\alpha} = X_s$. Set $A := B_M$ for a given positive constant $M > 0$, to constrain the policies into the set $\mathcal{A}_M := \{\alpha \in \mathcal{A} \,:\, |\alpha_s| \le M, \text{ for all } s \in [t, T]\}$. Given a target radius $r_0 > 0$, we define

$$J(t, x, \alpha) := \mathbb{E}[g(X_T^{t,x,\alpha})], \qquad g(x) := (|x| - r_0)^2, \qquad V(t, x) = \inf_{\alpha \in \mathcal{A}_M} J(t, x, \alpha).$$

This model can be interpreted as a target problem, in which we aim to reach the target radius r_0, within a finite time horizon $T > 0$ and bounded control. Indeed, for $t \in [0, T]$, this defines a *backward reachable set*, described as a *zero-level set* for the value function, given by

$$\mathcal{Z}_t := \{x \in \mathbb{R}^2 \,:\, V(t, x) = 0\}.$$

The condition $x \in \mathcal{Z}_t$ represents the states at time t from which the terminal target can be reached exactly, namely those for which there exists an admissible control α, with $|\alpha_s| \leq M$, for all $s \in [t, T]$, such that $|X_T^{t,x,\alpha}| = r_0$. See, for instance, Figure 1.

The value function is characterized, in the viscosity sense, by the HJB equation

$$\begin{cases} -\partial_t V(t,x) & = \inf\limits_{|a|\leq M} \left\{ \langle a, \nabla V(t,x) \rangle + \dfrac{1}{2} \mathrm{Tr}[\sigma\sigma^\top(t,x) \nabla^2 V(t,x)] \right\}, \quad x \in \mathbb{R}^2, \ t \in [0,T] \\ V(T,x) & = (|x| - r_0)^2, \quad x \in \mathbb{R}^2. \end{cases}$$

An analytic expression for V, together with $\mathcal{Z}_t$, can be obtained, see Section B. Notice that, whenever $\nabla V(t,x) \neq 0$, a possible optimal control satisfies

$$\alpha^*(t,x) = -M \frac{\nabla V(t,x)}{|\nabla V(t,x)|}. \tag{21}$$

In contrast, in the interior of the backward reachable set, V is locally constant (hence $\nabla V = 0$ wherever the classical gradient is defined), therefore the HJB does not select a unique optimal control there. In that region, the terminal target can be reached exactly, and any admissible control whose radial component steers the radius to r_0 is optimal; tangential components may be added as long as the constraint $|\alpha_s| \leq M$ remains satisfied.

Numerical implementation. We consider the time horizon $T = 0.5$, the target radius $r_0 = 1$, and the control bound $M = 1$. The feedback control is parametrized by a deep feedforward neural network with input (t, x_1, x_2), 3 hidden layers of 20 neurons each, and using the `tanh` activation function. The output layer $\Pi_A(u) := \frac{u}{\max\{1,|u|\}}$ is used in order to generate controls in the unit ball.

For each value of N and for each time-discretization scheme, the policy is trained independently. During training, the initial condition is sampled uniformly on the centered ball $B_{2.5}$. We use the Adam optimizer with learning rate $\gamma = 10^{-3}$, together with large Monte Carlo batches of size 10^5, over 10^5 training iterations. This choice is made in order to reduce the stochastic noise in the optimization and to make the error due to the time-discretization scheme clearly visible in the convergence study.

We perform simulations using both the Euler–Maruyama first order scheme and the explicit weak second order scheme of Platen. Since the Brownian noise is one-dimensional, we use the simplified version of the second-order scheme (see (31) in Appendix C; see also [31, Eq. (15.1.1)]). After training, the value induced by the learned policy is recomputed pointwise, for each fixed initial condition, using 10^5 independent Monte Carlo trajectories. This significantly reduces the Monte Carlo contribution in the final evaluation, so that the observed errors primarily reflect the approximation properties of the time-discretization scheme and, to a lesser extent, the neural-network approximation and optimization errors.

Figure 1 provides a first qualitative validation of the learned policy. The simulated trajectories show the expected radial behavior: initial states lying outside the backward reachable set are driven as far as possible toward the target circle, while those starting inside the reachable annulus can be steered so as to attain the target exactly within the terminal time.

The contour plot of the value induced by the trained neural-network policy is also in very good agreement with the theoretical structure of the problem, exhibiting an approximately rotationally invariant profile, with nearly concentric level sets. This is a nontrivial qualitative feature, since no symmetry is explicitly imposed in the neural-network architecture. The learned approximation therefore recovers not only the correct magnitude of the value function, but also its expected geometric structure.

The results reported in Figure 2 show a clear and robust convergence pattern. The slices of the value function along the section $x = (r, 0)$ already indicate that the second-order scheme is

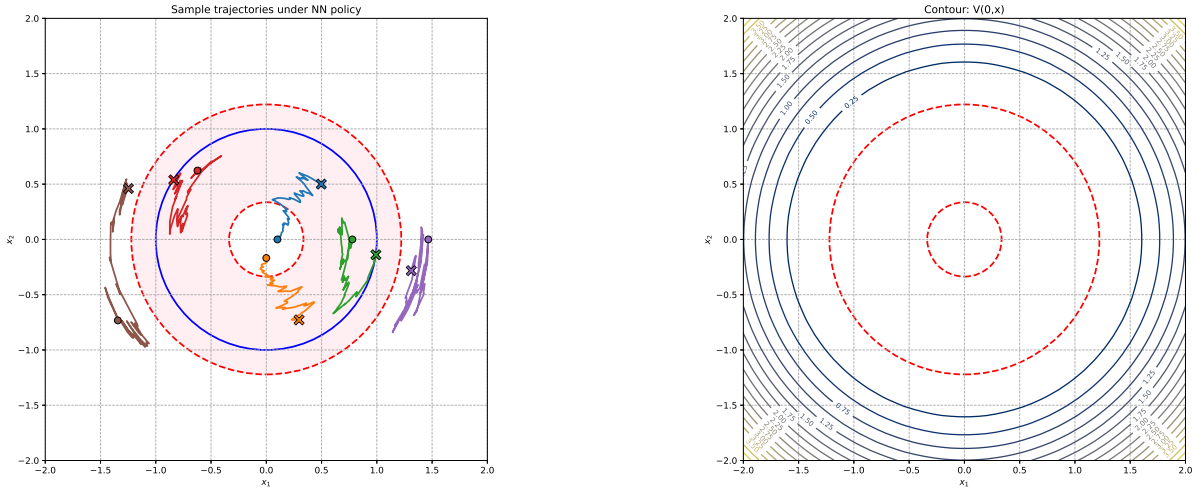


Figure 1: (Radial target problem) *Left*: Simulated independent trajectories over the time horizon $[0, T = 0.5]$ under the trained neural-network policy, with control constraint $M = 1$. Dots and crosses represent the initial and terminal states, respectively. The blue circle is the target of radius $r_0 = 1$. The pink annular region represents the backward reachable set at time $t = 0$, that is, $\mathcal{Z}_0$. *Right*: Contour plot of the Monte Carlo estimate of the value induced by the trained neural-network policy at time $t = 0$. The shape of the level sets is consistent with the rotational invariance of the underlying control problem.

substantially more accurate than the first-order one, even for small values of N. In particular, for $N = 4$, the approximation obtained with the Platen scheme is already very close to the exact value function, whereas the Euler–Maruyama approximation still exhibits a visible bias. As the time grid is refined, both schemes converge to the exact profile, with the second-order method remaining systematically more accurate.

This qualitative behavior is fully confirmed by Table 1 and Figure 3. For the Euler–Maruyama scheme, the empirical convergence rates in the L^1 and L^2 norms are close to 1, consistently with a first-order behavior in this benchmark. For the Platen weak second order scheme, the empirical rates are close to 2, especially in the L^1 and L^2 norms. The errors are evaluated along the section $x = (r, 0)$, $r \in (0, 2]$, by comparing the exact value function $V(0, x)$ with the Monte Carlo estimate induced by the trained policy. The reported L^1, L^2, and L^∞ errors are computed on a uniform grid in r and correspond to quadrature approximations of the continuous norms along this one-dimensional section.

The fact that the expected rates are recovered so accurately is particularly informative. It shows that, in the present experiment, the dominant contribution to the observed error is indeed the time-discretization error of the numerical scheme. In contrast, the neural-network approximation and optimization errors remain below this level across the tested range of discretizations. Moreover, since the final value is recomputed pointwise with 10^5 independent Monte Carlo trajectories, the Monte Carlo contribution in the final evaluation is strongly reduced. The results also suggest that restricting to piecewise-constant feedback policies does not dominate the convergence behavior in this benchmark, as otherwise one would not observe the theoretical orders of the underlying schemes so clearly.

6.2 Example 2: Hamilton–Jacobi–Bellman benchmark

The second example is a Hamilton–Jacobi–Bellman benchmark inspired by [23], and illustrates the improved accuracy of the second-order weak scheme, as well as the regime in which the neural-network approximation error becomes dominant.

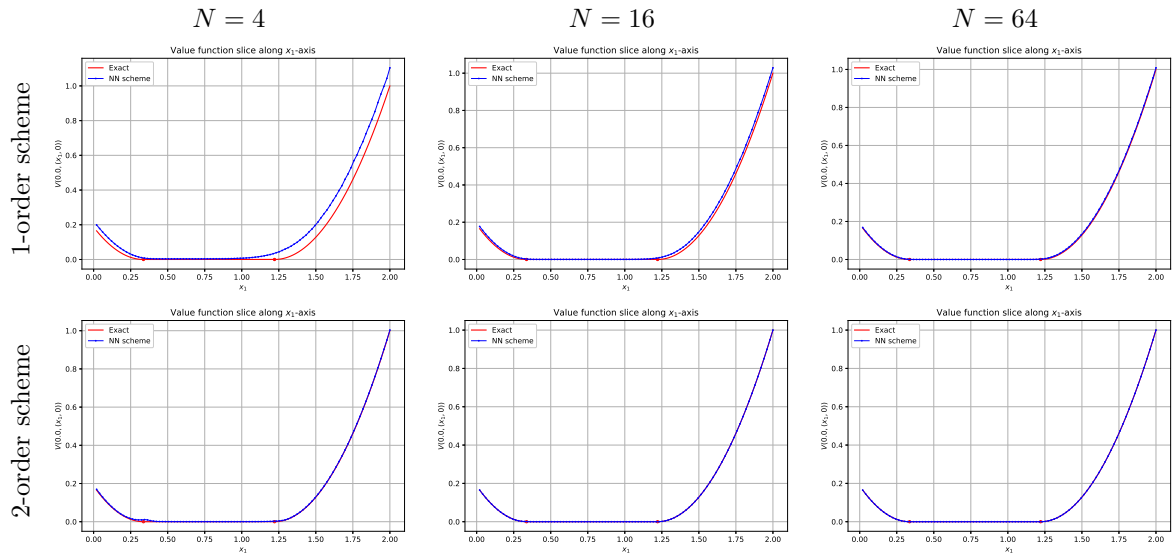


Figure 2: (Radial target problem) Comparison between the exact value function $V(0, x)$ and the Monte Carlo estimate of the value induced by the trained neural-network policy, along the section $x = (r, 0)$, $r \in (0, 2]$. The value induced by the trained policy is recomputed pointwise using 10^5 independent Monte Carlo trajectories. *Top row:* Euler–Maruyama scheme. *Bottom row:* Platen weak second order scheme.

Table 1: (Radial target problem) Error tables in discrete L^1, L^2, and L^∞ norms along the section $x = (r, 0)$, $r \in (0, 2]$. The error is computed between the exact value function $V(0, x)$ and the Monte Carlo estimate of the value induced by the trained neural-network policy, recomputed pointwise with 10^5 independent trajectories.

	N	$\|e\|_{L^1}$	Rate	$\|e\|_{L^2}$	Rate	$\|e\|_{L^\infty}$	Rate
1-order	2	1.38×10^{-1}	–	1.30×10^{-1}	–	1.74×10^{-1}	–
	4	7.66×10^{-2}	0.849	7.36×10^{-2}	0.819	1.03×10^{-1}	0.756
	8	4.06×10^{-2}	0.915	3.98×10^{-2}	0.887	5.77×10^{-2}	0.840
	16	2.12×10^{-2}	0.941	2.11×10^{-2}	0.916	3.02×10^{-2}	0.934
	32	1.07×10^{-2}	0.982	1.08×10^{-2}	0.966	1.64×10^{-2}	0.882
	64	5.40×10^{-3}	0.987	5.49×10^{-3}	0.976	8.71×10^{-3}	0.911
2-order	2	2.19×10^{-2}	–	2.04×10^{-2}	–	5.26×10^{-2}	–
	4	4.12×10^{-3}	2.411	4.05×10^{-3}	2.332	1.11×10^{-2}	2.248
	8	9.79×10^{-4}	2.075	1.07×10^{-3}	1.924	3.51×10^{-3}	1.657
	16	1.70×10^{-4}	2.526	2.15×10^{-4}	2.310	9.56×10^{-4}	1.878
	32	3.96×10^{-5}	2.103	5.78×10^{-5}	1.895	3.03×10^{-4}	1.657
	64	9.60×10^{-6}	2.043	1.30×10^{-5}	2.157	5.96×10^{-5}	2.345

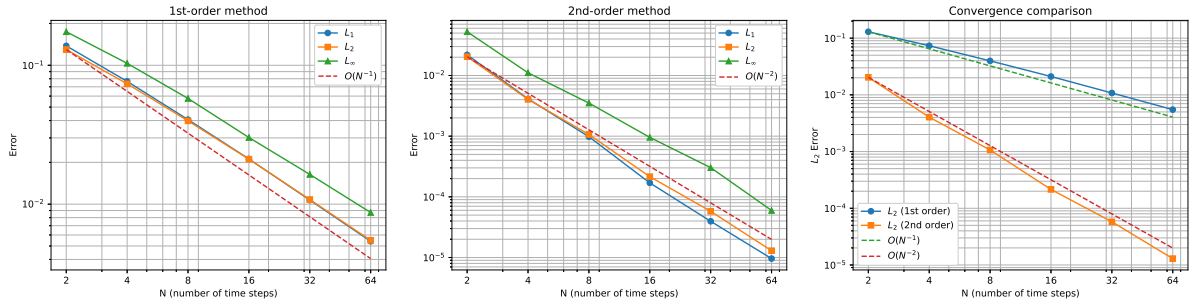


Figure 3: (Radial target problem) Log–log error for the value induced by the trained neural-network policy along the section $x = (r, 0)$, $r \in (0, 2]$. The dashed reference lines show a very clear first- and second-order convergence rates, as expected.

The model. We consider the HJB benchmark introduced in [23, Equations (12)–(14)], which we reformulate here as a stochastic optimal control problem. The d-dimensional controlled dynamics are described by the following stochastic differential equation

$$\mathrm{d}X_t = 2m_t\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W_t, \quad X_0 = x_0, \quad t \in [0, T],$$

where $m = (m_t)_{t\in[0,T]}$ represents the d-dimensional control process. The associated cost functional is given by

$$J(0, x_0, m) := \mathbb{E}\Big[\int_0^T \|m_t\|^2\,\mathrm{d}t + g(X_T)\Big],$$

with $g(x) := \ln\big(\frac{1+\|x\|^2}{2}\big)$, and $\|\cdot\|$ denoting the Euclidean norm. The infimum is taken over square-integrable progressively measurable controls.

Semi-analytic representation. The HJB equation of the problem is given by

$$\begin{cases} -\partial_t v(t,x) &= \Delta v(t,x) + \inf_{m\in\mathbb{R}^d}\{2\langle\nabla v(t,x), m\rangle + \|m\|^2\}, \quad (t,x) \in [0,T)\times\mathbb{R}^d \\ v(T,x) &= g(x), \quad x \in \mathbb{R}^d \end{cases}$$

with minimizer $m^*(t,x) = -\nabla v(t,x)$, which results in the following HJB equation

$$\begin{cases} -\partial_t v(t,x) &= \Delta v(t,x) - \|\nabla v(t,x)\|^2, \quad (t,x) \in [0,T)\times\mathbb{R}^d \\ v(T,x) &= g(x), \quad x \in \mathbb{R}^d. \end{cases}$$

By applying the Hopf–Cole transformation $u = \exp(-v)$ (cf. [12, Sec. 4.2]), u satisfies the backward heat equation $\partial_t u + \Delta u = 0$, with terminal condition $u(T,x) = \exp\{-g(x)\}$, and we deduce the following semi-analytic formulas for v and m^*:

$$v(t,x) = -\ln\Big(\mathbb{E}\Big[\exp\Big\{-g(x+\sqrt{2}W_{T-t})\Big\}\Big]\Big), \tag{22}$$

$$m^*(t,x) = -\frac{\mathbb{E}[\exp\{-g(x+\sqrt{2}W_{T-t})\}\nabla g(x+\sqrt{2}W_{T-t})]}{\mathbb{E}[\exp\{-g(x+\sqrt{2}W_{T-t})\}]}. \tag{23}$$

We observe that $\|\nabla g\| \le 1$, therefore, the unconstrained optimal feedback satisfies $\|m^*(t,\cdot)\| \le 1$, $\forall t \in (0,T)$. Thus, restricting the admissible controls to the compact set $A = B_1$ does not change the value function and allows us to apply the theoretical framework developed in the previous sections.

Numerical implementation. In this example, we consider the time horizon $T = 1$ in dimension $d = 6$. The feedback control is parametrized by a deep feedforward neural network taking as input the $(d+1)$-dimensional vector (t, x). The network has five hidden layers with 80 neurons each and uses the `SiLU` activation function, which provides smoother control outputs with respect to `ReLU` for example. Since the optimal feedback satisfies $\|m^*(t,x)\| \leq 1$, the control constraint is enforced through the output activation

$$\sigma_{\text{out}}(z) = \begin{cases} \frac{z}{\|z\|} \sin^4\left(\frac{\pi}{2}\|z\|\right), & 0 < \|z\| < 1, \\ \frac{z}{\|z\|}, & \|z\| \geq 1, \\ 0, & z = 0. \end{cases}$$

This activation maps the network output into the unit ball and provides a smooth radial saturation of the control norm, having a flat derivative for both $\|z\| \to 0$ and $\|z\| \to 1$. For each value of N and for each time-discretization scheme, the policy is trained independently. The training uses the Adam optimizer with initial learning rate $\gamma = 10^{-3}$, together with the adaptive learning-rate scheduler `ReduceLROnPlateau`, from `PyTorch` [41], applied to the average training loss. The scheduler reduces the learning rate by a factor of 0.5 after a plateau, with patience 1000, cooldown 500, and threshold 10^{-4}, helping achieve better convergence. We train over 10^4 iterations, using Monte Carlo batches of size 10^4. During training, the initial condition is sampled from a compactly supported isotropic distribution in $B_{2.5}$, by drawing a random direction uniformly on the sphere and an independent radius uniformly in $[0, 2.5]$. We perform simulations with both the Euler–Maruyama scheme and the Platen weak order 2.0 scheme. Since the diffusion coefficient is constant, the second-order scheme reduces to the form reported in (32) from Section C. After training, the value induced by the learned policy is recomputed pointwise along the state-space diagonal, at time $t = 0$, using 10^6 independent Monte Carlo trajectories for each point.

	1-order		2-order	
N	$\|e\|_{L^2}$	Rate	$\|e\|_{L^2}$	Rate
2	8.94×10^{-2}	–	1.29×10^{-2}	–
4	4.91×10^{-2}	0.865	3.34×10^{-3}	1.949
8	2.57×10^{-2}	0.934	1.26×10^{-3}	1.406
16	1.34×10^{-2}	0.940	9.72×10^{-4}	0.374
32	7.06×10^{-3}	0.924	8.66×10^{-4}	0.167
64	3.85×10^{-3}	0.875	8.23×10^{-4}	0.073

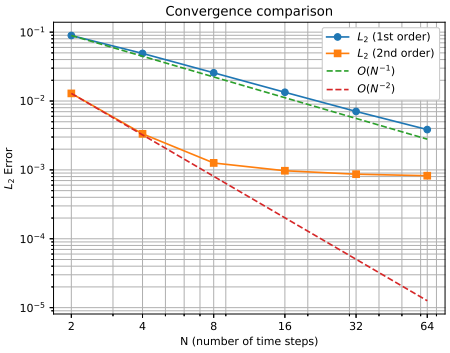


Figure 4: (HJB benchmark problem) Error tables in discrete L^2 norm and corresponding log–log plot for the value induced by the trained neural-network policy. The error is computed along the state-space diagonal $x = (\xi, \ldots, \xi)$, $\xi \in [0, 2]$, by comparison with the semi-analytic value function. The value induced by the trained policy is recomputed using 10^6 independent Monte Carlo trajectories for each point on the diagonal. The dashed lines indicate first- and second-order reference rates.

The convergence results are reported in Figure 4. The error is computed along the state-space diagonal $x = (\xi, \ldots, \xi)$ for $\xi \in [0, 2]$, by comparing the semi-analytic value function in (22) with the Monte Carlo estimate of the value induced by the trained policy. The reported L^2 errors are computed on a uniform grid in ξ, as quadrature approximations of the corresponding continuous norm along this one-dimensional section. The Euler–Maruyama scheme exhibits a convergence rate close to 1 across the tested time steps. The Platen scheme is significantly more accurate already on coarse grids and initially shows a second-order decay of the error. However,

after a few refinements, the second-order curve flattens and no longer follows the reference $O(N^{-2})$ slope. We think this behavior is consistent with the neural-network approximation error becoming dominant once the time-discretization error has been sufficiently reduced, suggesting the saturation of the capacity of the considered network.

This example therefore complements the radial target benchmark: previously, the neural network error was small enough to recover the expected orders of both schemes over the full range of discretizations. Here, instead, the second-order scheme rapidly reduces the discretization error to a level at which the residual error is mainly governed by the neural networks' approximation capacity and training accuracy.

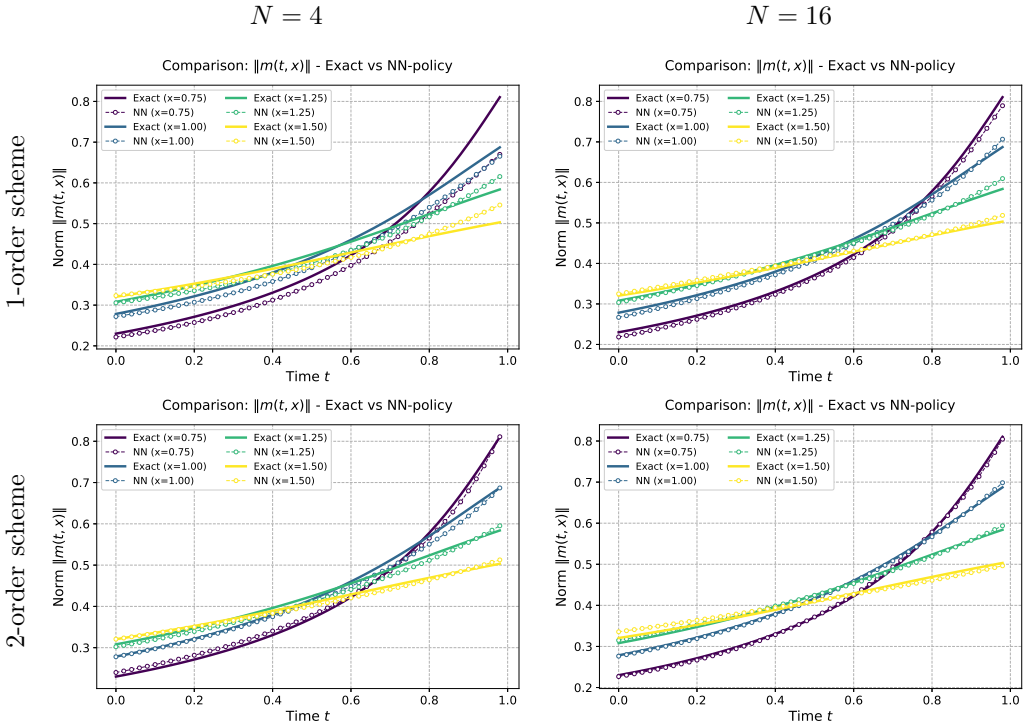


Figure 5: (HJB benchmark problem) Comparison between the norm of the semi-analytic optimal feedback and the norm of the learned neural-network feedback along the diagonal $x = (\xi, \dots, \xi)$, evaluated on a uniform grid of time points in $[0, 1]$ and selected values $\xi \in \{0.75, 1.0, 1.25, 1.5\}$.

Figure 5 compares the norm of the learned feedback control with the semi-analytic optimal one along the diagonal $x = (\xi, \dots, \xi)$, for selected values of ξ in the region typically explored during training. This comparison is possible in the present example because the optimal feedback is explicitly characterized by (23). The plots show that the learned control captures the qualitative time profile of the optimal control. The agreement improves as the number of time steps used in training increases, especially in the region where the controlled trajectories are more frequently observed. Moreover, for the same value of N, the policy trained with the second-order scheme generally provides a closer approximation of the optimal control norm than the one trained with the Euler–Maruyama scheme. This is consistent with the value function errors from Figure 4, where the second-order scheme reduces discretization bias, while the remaining discrepancy in the control reflects residual neural-network approximation and optimization errors.

6.3 Example 3: Gas storage problem

The third example concerns a gas storage problem (cf. [5, 11, 50]) where prices are driven by Ornstein–Uhlenbeck processes. In this case, the exact simulation of the Ornstein–Uhlenbeck factors allows us to eliminate the time-discretization error in the exogenous stochastic factors

and to focus on the effect of the piecewise-constant policy approximation with respect to the number of time steps.

The model. We examine a realistic gas storage problem in which the reservoir manager aims to maximize expected profit by optimally scheduling injections and withdrawals and purchasing the commodity when prices are low and selling it when prices are high. We consider a standard model for futures prices with short and long-term volatility factors. For each delivery maturity $\mathcal{T} > 0$, the futures price $F(t, \mathcal{T})$, for $t \in [0, \mathcal{T}]$, follows the stochastic differential equation

$$\frac{\mathrm{d}F(t,\mathcal{T})}{F(t,\mathcal{T})} = e^{-a(\mathcal{T}-t)}\sigma^S \mathrm{d}W_t^S + \sigma^L \mathrm{d}W_t^L, \tag{24}$$

for a, σ^S, σ^L positive, with W^S, W^L independent real-valued Brownian motions. The explicit solution to the linear SDE (24) is given by

$$F(t,\mathcal{T}) = F(0,\mathcal{T}) \exp\left\{ \int_0^t e^{-a(\mathcal{T}-u)}\sigma^S \mathrm{d}W_u^S + \sigma^L W_t^L - \frac{1}{2}\int_0^t \left(e^{-2a(\mathcal{T}-u)}(\sigma^S)^2 + (\sigma^L)^2\right)\mathrm{d}u \right\}.$$

Denoting the spot price $S_t := F(t, \mathcal{T} = t)$, we get

$$S_t = F(0,t) \exp\left\{ \int_0^t e^{-a(t-u)}\sigma^S \mathrm{d}W_u^S + \sigma^L W_t^L - \frac{1}{2}\int_0^t \left(e^{-2a(t-u)}(\sigma^S)^2 + (\sigma^L)^2\right)\mathrm{d}u \right\}, \tag{25}$$

where we observe, by defining $\hat{W}_t^S := \int_0^t e^{-a(t-u)}\sigma^S \mathrm{d}W_u^S$, i.e., the solution to the Ornstein–Uhlenbeck process $\mathrm{d}\hat{W}_t^S = -a\hat{W}_t^S \mathrm{d}t + \sigma^S \mathrm{d}W_t^S$, and $\hat{W}_t^L := \sigma^L W_t^L$, that we are able to recover the Markovian setting of S_t as a function of $(t, \hat{W}_t^S, \hat{W}_t^L)$.

The reservoir manager, according to the spot price S_t, aims to maximize the expected profit over a finite time horizon $T > 0$, typically one year, given by $J(u) := -\mathbb{E}[\int_0^T S_t u_t \mathrm{d}t]$, where $(u_t)_{t\in[0,T]}$ represents the control process, i.e., injection ($u_t > 0$) or withdrawal ($u_t < 0$) rate. The optimization problem is subject to constraints concerning the characteristics of the storage:

- $(Q_t)_{t\in[0,T]}$, the gas level in the storage, described by $\mathrm{d}Q_t = u_t \mathrm{d}t$, with initial inventory level $Q_0 = Q_{\text{init}}$;
- withdrawal $C_W(Q)$ and injection $C_I(Q)$ rates, depending on the gas level in the storage Q;
- $Q_{\min}(t)$, $Q_{\max}(t)$, as the minimum and maximum levels of gas in the storage, at time t.

This results in a non-linear control problem, with flow constraints, for $t \in [0, T]$

$$\begin{cases} -C_W(Q_t) \le u_t \le C_I(Q_t), \\ Q_{\min}(t) \le Q_t \le Q_{\max}(t), \end{cases}$$

where the first is a constraint on the control, and the second is a state constraint.

Time discretization and reduction to control constraints. The controlled Markov state is given by $(t, Q_t, \hat{W}_t^S, \hat{W}_t^L)_{t\in[0,T]}$, where the dynamics are degenerate in the inventory component Q. We measure time in days and consider the storage problem over one year, setting $T = 365$, and $t_n = n\tau$, with $\tau = T/N$, for $n = 0, \dots, N$. The parameters appearing in the continuous-time futures model are rescaled accordingly. More precisely, if a_{year}, σ^S_{year}, and σ^L_{year} denote the annualized parameters, we set

$$a = \frac{a_{\text{year}}}{365}, \qquad \sigma^S = \frac{\sigma^S_{\text{year}}}{\sqrt{365}}, \qquad \sigma^L = \frac{\sigma^L_{\text{year}}}{\sqrt{365}}.$$

In the sequel, to simplify notation, we keep writing a, σ^S, σ^L for these daily-scaled parameters. The exogenous short-term Ornstein–Uhlenbeck factor and the long-term Brownian factor are simulated exactly on the grid, consistent with (25), and the discrete-time dynamics are given by the recursive relation

$$\begin{cases} \hat{W}^S_{n+1} & = e^{-a\tau}\hat{W}^S_n + \sigma^S\sqrt{\frac{1-e^{-2a\tau}}{2a}}Z^S_{n+1}, \\ \hat{W}^L_{n+1} & = \hat{W}^L_n + \sigma^L\sqrt{\tau}Z^L_{n+1}, \\ Q_{n+1} & = Q_n + U_n, \\ S_n & = F(0,t_n)\exp\Big\{\hat{W}^S_n + \hat{W}^L_n - \frac{(\sigma^S)^2}{4a}(1-e^{-2at_n}) - \frac{1}{2}(\sigma^L)^2 t_n\Big\}, \end{cases}$$

where (Z^S_{n+1}, Z^L_{n+1}) are independent standard normal random variables. The discrete control U_n represents the net amount of gas injected into the storage during the period $[t_n, t_{n+1})$ and is related to the continuous-time rate by $U_n = \tau u_n$. Hence, the discretized reward functional is $\tilde{J}(U) = -\mathbb{E}[\sum_{n=0}^{N-1} S_n U_n]$, which corresponds to the expected trading profit associated with the discrete injection/withdrawal strategy.

The flow constraints take the form

$$U_n \in [-C_W(Q_n)\tau,\ C_I(Q_n)\tau] =: [U_W(Q_n),\ U_I(Q_n)], \qquad n = 0, \dots, N-1, \tag{26}$$

$$Q_n \in [Q_{\min}(t_n),\ Q_{\max}(t_n)], \qquad n = 0, \dots, N, \tag{27}$$

where $U_W(Q) \le 0$ denotes the withdrawal lower bound and $U_I(Q) \ge 0$ the injection upper bound. The parametrization of the control process, using neural networks, is introduced as follows. The state constraint (27) is equivalent to using

$$U_n \in [Q_{\min}(t_{n+1}) - Q_n,\ Q_{\max}(t_{n+1}) - Q_n], \qquad n = 0, \dots, N-1. \tag{28}$$

The intersection between (26) and (28) yields a state-dependent admissible interval for the discrete control, which enforces both the flow and the inventory constraints at the next grid point:

$$U_n \in [\underline{U}_n(Q_n),\ \overline{U}_n(Q_n)], \quad \text{where} \quad \begin{cases} \underline{U}_n(Q) & := \max\{Q_{\min}(t_{n+1}) - Q,\ U_W(Q)\}, \\ \overline{U}_n(Q) & := \min\{Q_{\max}(t_{n+1}) - Q,\ U_I(Q)\}. \end{cases} \tag{29}$$

Since the problem is Markovian in $(Q, \hat{W}^S, \hat{W}^L)$ at each time step, we restrict to feedback controls of the form $U_n = \alpha_n(Q_n, \hat{W}^S_n, \hat{W}^L_n)$. We introduce a network per time step, $\phi^{\theta_n}_n$ with parameters θ_n, as an operator from $\mathbb{R}^3$ to $[0,1]$ such that, following [51], the feedback policies are approximated by

$$U^{\theta_n}_n = \underline{U}_n(Q_n) + \Big(\overline{U}_n(Q_n) - \underline{U}_n(Q_n)\Big)\phi^{\theta_n}_n(Q_n, \hat{W}^S_n, \hat{W}^L_n), \qquad n = 0, \dots, N-1.$$

Noting $\theta = (\theta_n)_{n=0}^{N-1}$, we approximate the optimal reservoir management by solving

$$\theta^* = \operatorname*{argmax}_\theta \tilde{J}(U^\theta) = \operatorname*{argmin}_\theta \mathbb{E}\Big[\sum_{n=0}^{N-1} S_n U^\theta_n\Big].$$

Numerical implementation. In this example, the exogenous price factors driving the spot price are simulated exactly on the time grid. Hence, unlike in the first two examples, there is no time-discretization error associated with the stochastic price factors. The observed error is therefore mainly related to the restriction to piecewise-constant policies, together with the neural-network approximation and optimization errors.

The storage horizon runs from April 1, 2024, to April 1, 2025, corresponding to $T = 365$ days. The annualized parameters used for the two-factor price model are $a_{\text{year}} = 1.9641$, $\sigma^S_{\text{year}} = 0.7877$,

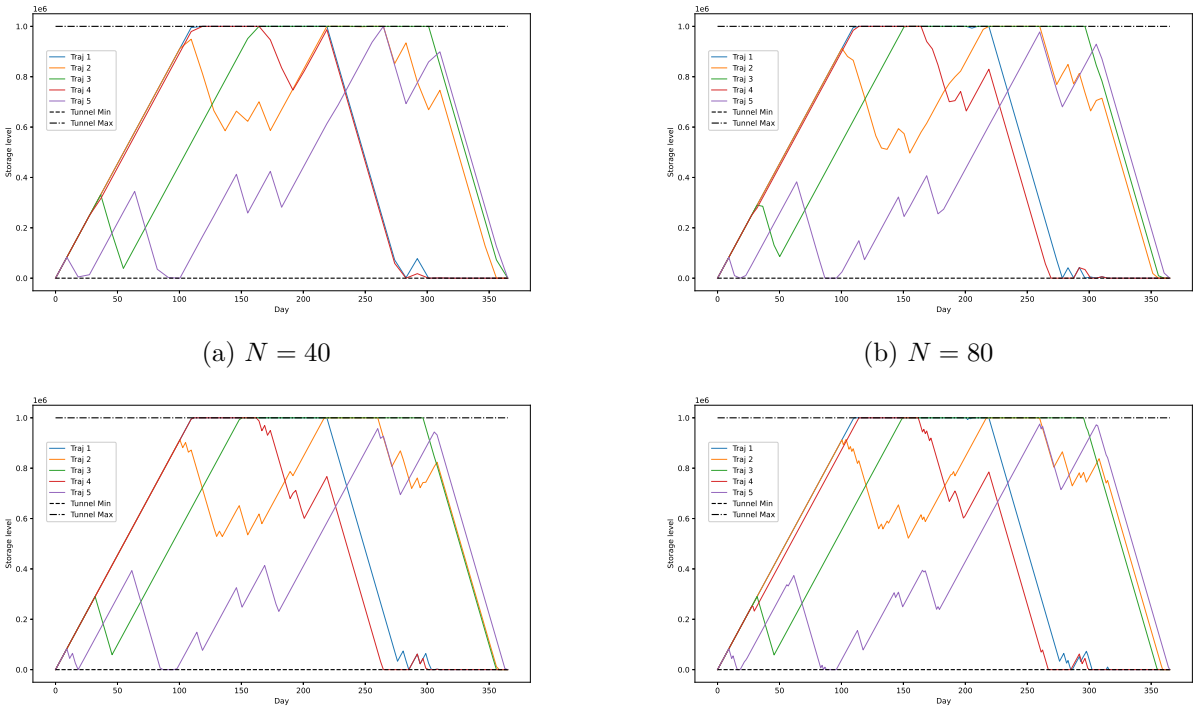
and $\sigma^L_{\text{year}} = 0.4968$. In order to estimate the parameters a_{year}, σ^S_{year}, and σ^L_{year}, a maximum-likelihood method is used to fit the returns of the different products available on the market, namely spot prices and monthly products. The spot product, more precisely the day-ahead product, together with end-of-week, end-of-month, and monthly products, are observed on the market. Since these products correspond to non-overlapping delivery periods, a piecewise constant-in-time representation of $F(0,\cdot)$ on a daily grid is obtained from these quotations.

In order to isolate the effect of the time discretization of the control policy, the initial futures curve is replaced by a piecewise-constant approximation on the coarsest grid used in the experiments, corresponding to $N = 5$. Since the storage horizon is $T = 365$ days, this gives a time step $\tau = 73$ days. More precisely, we set $F(0,t) = \sum_{i=0}^{4} \widehat{F}_i \mathbf{1}_{I_i}(t)$, with $I_i = [i\tau, (i+1)\tau)$ for $i = 0, \dots, 3$, and $I_4 = [4\tau, T]$. The constants $\widehat{F}_i$ are computed from the original market forward curve by averaging its values over the corresponding time window I_i. The same input curve is then used for all finer grids. This avoids introducing additional deterministic variations in the futures curve as the number of decision dates increases, so that the convergence study primarily reflects the refinement of the piecewise-constant policy. We use the simplified storage configuration, with no seasonal gate constraint (generally imposed by the storage owner when renting part of the capacity to third parties), and constant injection/withdrawal rates: $Q_{\max} = 10^6$, $Q_{\min} = 0$, $Q_0 = 0$, $C_I = \frac{Q_{\max}}{110}$, and $C_W = \frac{Q_{\max}}{59}$. Notice that, in this setting, the objective is affine with respect to the storage control, which leads to a bang–bang optimal feedback. For each value of N, we implement one feedforward neural network per time step. Each network takes as input the three-dimensional Markov state $(Q_n, \hat{W}^S_n, \hat{W}^L_n)$ and outputs a scalar in $[0,1]$ through a final `sigmoid` activation. The hidden layers use the `tanh` activation function. In all simulations reported here, each network (per time-step) has two hidden layers with 10 neurons each. The output of the network is then mapped into the admissible control interval according to (29).

Since no closed-form solution is available for this constrained storage problem, we use the value obtained with $N = 320$ as a reference benchmark, $v_{\text{ref}} := v_{320}$. This grid corresponds to a time step of $\tau_{\text{ref}} = \frac{365}{320} \simeq 1.14$ days, hence approximately one policy update per day. We consider the time grids $N \in \{5, 10, 20, 40, 80, 160, 320\}$. For $N = 5, \dots, 160$, the networks are trained for 8×10^4 Adam iterations, using learning rate $\gamma = 10^{-3}$ and Monte Carlo batches of size 10^5. For the reference computation with $N = 320$, the training is increased to 16×10^4 Adam iterations. During training, the current policy is periodically evaluated using batches of size 10^6. After training, the reported value is recomputed with 10^8 independent Monte Carlo trajectories, processed in batches of size 5×10^6.

Figure 6 provides a qualitative illustration of the storage policies obtained for different time grids. The same colors correspond to the same underlying realization of the exogenous noise, generated in a nested way across the different grids. Hence, the observed differences between the panels are mainly due to the refinement of the decision grid and the corresponding re-optimization of the neural-network policies under the same piecewise-constant input futures curve. Several qualitative features can be observed. First, the inventory constraint is satisfied along all simulated trajectories, which confirms the effect of the constraint-preserving parametrization in (29). Second, the trajectories exhibit consistent storage-management behavior across different discretizations: the reservoir is filled and emptied in response to the evolution of the underlying price factors. As the number of time steps increases, the policies become more flexible in the timing of injection and withdrawal decisions. The refinement from $N = 40$ to $N = 320$ mainly affects the local timing of the operations, while the overall qualitative structure of the trajectories remains stable.

The convergence study compares the values obtained on coarser decision grids with the reference value $v_{\text{ref}} := v_{320}$. For $N \in \{5, 10, 20, 40, 80, 160\}$, we define the oriented relative error $e_N := \frac{v_{\text{ref}} - v_N}{v_{\text{ref}}}$. In the present experiment, the estimated values satisfy $v_N < v_{\text{ref}}$, so that e_N is



(a) $N = 40$ (b) $N = 80$

(c) $N = 160$ (d) $N = 320$ (reference benchmark)

Figure 6: (Gas storage problem) Simulated trajectories of the storage level for different time grids. Trajectories with the same color correspond to the same underlying realization of the driving Gaussian innovations. The innovations are generated once on the finest grid, $N = 320$, and the innovations used on coarser grids are obtained by aggregating consecutive fine-grid increments over the corresponding time intervals. Thus, the different panels compare the policies under the same exogenous noise scenario, refined as the time grid becomes finer.

positive. Let $\widehat{v}_N$ and $\widehat{v}_{\mathrm{ref}}$ denote the Monte Carlo estimators of v_N and v_{ref}, respectively. We estimate e_N by $\widehat{e}_N = 1 - \widehat{r}_N$, with $\widehat{r}_N := \frac{\widehat{v}_N}{\widehat{v}_{\mathrm{ref}}}$. To construct confidence intervals for the relative error, we apply the linear delta method to the ratio $\widehat{r}_N$ (cf. [14, Sec. 2.2]). Assuming that $\widehat{v}_N$ and $\widehat{v}_{\mathrm{ref}}$ are computed from independent Monte Carlo samples, we obtain

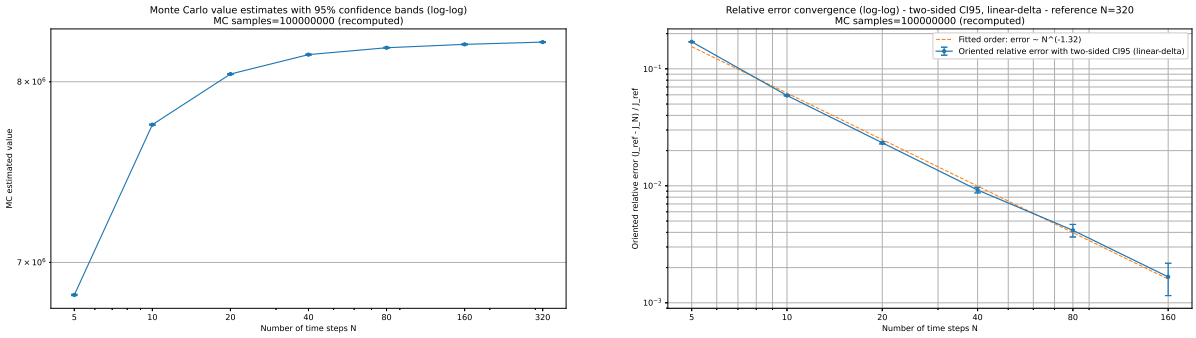
$$\widehat{\mathrm{SE}}(\widehat{e}_N) = \widehat{\mathrm{SE}}(\widehat{r}_N) = \left[\frac{\left(\widehat{\mathrm{SE}}(\widehat{v}_N)\right)^2}{\left(\widehat{v}_{\mathrm{ref}}\right)^2} + \frac{\left(\widehat{v}_N\right)^2}{\left(\widehat{v}_{\mathrm{ref}}\right)^4} \left(\widehat{\mathrm{SE}}(\widehat{v}_{\mathrm{ref}})\right)^2 \right]^{1/2},$$

where $\widehat{\mathrm{SE}}(\cdot)$ denotes the standard error, which refers only to the Monte Carlo error in the final post-training evaluation. More precisely, once the trained policy is fixed, $\widehat{v}_N$ is computed as the empirical average of the gains over $M = 10^8$ independent simulated trajectories, and $\widehat{\mathrm{SE}}(\widehat{v}_N)$ is the corresponding sample standard deviation divided by $\sqrt{M}$. The confidence intervals therefore quantify the statistical uncertainty of the Monte Carlo evaluation conditional on the trained policy. Therefore, the two-sided confidence interval with confidence level $1 - \alpha$ is given by

$$\mathrm{CI}_{1-\alpha}(\widehat{e}_N) = \left[\widehat{e}_N - z_{1-\alpha/2} \cdot \widehat{\mathrm{SE}}(\widehat{e}_N),\ \widehat{e}_N + z_{1-\alpha/2} \cdot \widehat{\mathrm{SE}}(\widehat{e}_N) \right],$$

where $z_{1-\alpha/2}$ denotes the $(1 - \alpha/2)$-quantile of the standard normal distribution. In the plots and tables below, we use $\alpha = 0.05$.

Figure 7 summarizes the quantitative convergence results. The left panel shows that the optimized gains increase monotonically with the number of time steps. This is consistent with



(a) Estimated optimized gains.

(b) Relative error with respect to $N = 320$.

Figure 7: (Gas storage problem) *Left:* Monte Carlo estimates of the optimized gain for different time grids, with 95% confidence intervals, in log–log scale. *Right:* Oriented relative error $(v_{320} - v_N)/v_{320}$ in log–log scale, with confidence intervals computed by the linear delta method and fitted algebraic decay.

the fact that finer grids provide a richer class of piecewise-constant policies and therefore allow for more flexible injection and withdrawal decisions. Although such monotonicity is not enforced by the neural-network parametrization and training procedure, it is clearly observed in the numerical results, suggesting that the approximation and optimization errors are sufficiently controlled in this experiment. The right panel reports the relative errors e_N in log–log scale. The error decreases regularly as the time grid is refined. The dashed line is obtained by a least-squares linear regression in log–log scale over the points $N = 5, \dots, 160$, yielding the empirical behavior $e_N \simeq CN^{-1.32}$. This rate should be interpreted as an empirical convergence rate with respect to the numerical reference value v_{320}, rather than as an exact theoretical order. Nevertheless, the regular decay confirms that refining the decision grid effectively reduces the loss induced by the piecewise-constant policy restriction. The confidence intervals are barely visible in the gain plot, since the final values are recomputed with 10^8 independent Monte Carlo trajectories; in the log–log error plot, the confidence intervals appear asymmetric because they are displayed on a logarithmic scale.

The corresponding numerical values are reported in Table 2, together with the Monte Carlo confidence intervals, the relative errors, and the observed local rates. The table quantitatively

Table 2: (Gas storage problem) Values $\widehat{v}_N$, errors $\widehat{e}_N$ and estimated order of convergence. The estimated values $\widehat{v}_N$ are recomputed using 10^8 independent Monte Carlo trajectories. The reference value is $\widehat{v}_{320}$. The reported confidence intervals are half-widths. For instance, the entry $\widehat{v}_N = 6.8338 \times 10^6$ with $\mathrm{CI}_{95\%}(\widehat{v}_N) = 3.356 \times 10^3$ corresponds to $\widehat{v}_N = (6.8338 \pm 3.356 \times 10^{-3}) \times 10^6$.

N	$\widehat{v}_N$	$\mathrm{CI}_{95\%}(\widehat{v}_N)$	$\widehat{e}_N$	$\mathrm{CI}_{95\%}(\widehat{e}_N)$	Rate
5	6.8338×10^6	3.356×10^3	1.7045×10^{-1}	5.067×10^{-4}	–
10	7.7503×10^6	3.088×10^3	5.9196×10^{-2}	5.073×10^{-4}	1.526
20	8.0457×10^6	3.022×10^3	2.3331×10^{-2}	5.104×10^{-4}	1.343
40	8.1621×10^6	3.005×10^3	9.2102×10^{-3}	5.125×10^{-4}	1.341
80	8.2036×10^6	2.998×10^3	4.1670×10^{-3}	5.131×10^{-4}	1.144
160	8.2242×10^6	2.993×10^3	1.6652×10^{-3}	5.134×10^{-4}	1.323
320	8.2379×10^6	2.993×10^3	–	–	–

confirms the behavior observed in Figure 7. The Monte Carlo confidence intervals are small compared with the differences between consecutive discretizations, especially for coarse and

intermediate grids. The local rates are relatively stable and mostly close to the fitted order observed in the log–log plot, with some expected variability due to the use of a numerical reference value and the presence of neural-network approximation and optimization errors. Since the exogenous stochastic factors are simulated exactly on the grid, the observed convergence directly reflects the discretization error induced by piecewise-constant policies. Overall, the results show a clear and consistent reduction of the error as the decision grid is refined.

References

[1] L. A. Abbas-Turki, J.-F. Chassagneux, J.-P. Lemor, G. Loeper, and S. Sananes. Stochastic Policy Gradient Methods in the Uncertain Volatility Model. *arXiv preprint arXiv:2605.06670*, 2026.

[2] C. Anil, J. Lucas, and R. Grosse. Sorting out Lipschitz function approximation. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 291–301. PMLR, 2019.

[3] M. Assellaou, O. Bokanowski, and H. Zidani. Error estimates for second order Hamilton-Jacobi-Bellman equations. Approximation of probabilistic reachable sets. *Discrete Contin. Dyn. Syst.*, 35(9):3933–3964, 2015.

[4] A. Bachouch, C. Huré, N. Langrené, and H. Pham. Deep neural networks algorithms for stochastic control problems on finite horizon: Numerical applications. *Methodology and Computing in Applied Probability*, 24(1):143–178, 2022.

[5] C. Barrera-Esteve, F. Bergeret, C. Dossal, E. Gobet, A. Meziou, R. Munos, and D. Reboul-Salze. Numerical methods for the pricing of swing options: a stochastic control approach. *Methodology and Computing in Applied Probability*, 8(4):517–540, 2006.

[6] C. Beck, W. E, and A. Jentzen. Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *J. Nonlinear Sci.*, 29:1563–1619, 2019.

[7] D. P. Bertsekas and S. E. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, 1996.

[8] O. Bokanowski, J.-F. Chassagneux, X. Li, and C. Reisinger. Numerical approximation for path-dependent McKean–Vlasov control with non-asymptotic error estimates. *arXiv preprint arXiv:2606.27181*, 2026.

[9] O. Bokanowski, A. Prost, and X. Warin. Neural networks for first order HJB equations and application to front propagation with obstacle terms. *Partial Differ. Equ. Appl.*, 4(5):Paper No. 45, 36, 2023.

[10] O. Bokanowski and X. Warin. Representation results and error estimates for differential games with applications using neural networks. *Dynamic Games and Applications*, 15(2):417–453, 2025.

[11] R. Carmona and M. Ludkovski. Valuation of energy storage: An optimal switching approach. *Quantitative Finance*, 10(4):359–374, 2010.

[12] J.-F. Chassagneux and A. Richou. Numerical simulation of quadratic BSDEs. *The Annals of Applied Probability*, 26(1):262–304, 2016.

[13] S. N. Cohen, J. Hebner, D. Jiang, and J. Sirignano. Neural actor-critic methods for Hamilton–Jacobi–Bellman PDEs: Asymptotic analysis and numerical studies. *arXiv preprint arXiv:2507.06428*, 2025.

[14] C. Cox. Fieller's theorem, the likelihood and the delta method. *Biometrics*, 46(3):709–718, 1990.

[15] W. E, J. Han, and A. Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4):349–380, 2017.

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

[17] W. H. Fleming and H. M. Soner. *Controlled Markov Processes and Viscosity Solutions*, volume 25 of *Stochastic Modelling and Applied Probability*. Springer, New York, 2nd edition, 2006.

[18] G. B. Folland. *Real Analysis: Modern Techniques and Their Applications*. Pure and Applied Mathematics. John Wiley & Sons, New York, 2nd edition, 1999.

[19] N. Frikha, M. Germain, M. Laurière, H. Pham, and X. Song. Actor-critic learning for mean-field control in continuous time. *Journal of Machine Learning Research*, 26(127):1–42, 2025.

[20] M. Germain, H. Pham, and X. Warin. Neural networks-based algorithms for stochastic control and PDEs in finance. *arXiv preprint arXiv:2101.08068*, 2021.

[21] X. Guo, A. Hu, and Y. Zhang. Reinforcement learning for linear-convex models with jumps via stability analysis of feedback controls. *SIAM Journal on Control and Optimization*, 61(2):755–787, 2023.

[22] J. Han and W. E. Deep learning approximation for stochastic control problems. In *Advances in Neural Information Processing Systems (NeurIPS), Deep Reinforcement Learning Workshop*, 2016.

[23] J. Han, A. Jentzen, and W. E. Solving high-dimensional partial differential equations using deep learning. *Proc. Natl. Acad. Sci. USA*, 115(34):8505–8510, 2018.

[24] J. Han and J. Long. Convergence of the deep BSDE method for coupled FBSDEs. *Probability, Uncertainty and Quantitative Risk*, 5(1):1–33, 2020.

[25] R. Hu and M. Laurière. Recent developments in machine learning methods for stochastic control and games. *Numerical Algebra, Control and Optimization*, 14(3):435–525, 2024.

[26] M. Hua, M. Laurière, and E. Vanden-Eijnden. A simulation-free deep learning approach to stochastic optimal control. *arXiv preprint arXiv:2410.05163*, 2024.

[27] Z. Huang, B. Negyesi, and C. W. Oosterlee. Convergence of the deep BSDE method for stochastic control problems formulated through the stochastic maximum principle. *Mathematics and Computers in Simulation*, 227:553–568, 2025.

[28] C. Huré, H. Pham, A. Bachouch, and N. Langrené. Deep neural networks algorithms for stochastic control problems on finite horizon: convergence analysis. *SIAM Journal on Numerical Analysis*, 59(1):525–557, 2021.

[29] C. Huré, H. Pham, and X. Warin. Deep backward schemes for high-dimensional nonlinear PDEs. *Mathematics of Computation*, 89(324):1547–1579, 2020.

[30] E. R. Jakobsen, A. Picarelli, and C. Reisinger. Improved order 1/4 convergence for piecewise constant policy approximation of stochastic control problems. *Electronic Communications in Probability*, 24:Paper No. 59, 10, 2019.

[31] P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, Berlin, Heidelberg, 1992.

[32] N. V. Krylov. Approximating value functions for controlled degenerate diffusion processes by using piece-wise constant policies. *Electronic Journal of Probability*, 4(2):1–19, 1999.

[33] W. Lefebvre, G. Loeper, and H. Pham. Differential learning methods for solving fully nonlinear PDEs. *Digital Finance*, 5(1):183–229, 2023.

[34] G. N. Milstein and M. V. Tretyakov. *Stochastic Numerics for Mathematical Physics*. Scientific Computation. Springer, Berlin, Heidelberg, 2004.

[35] S. Peng. A general stochastic maximum principle for optimal control problems. *SIAM Journal on Control and Optimization*, 28(4):966–979, 1990.

[36] H. Pham. *Continuous-Time Stochastic Control and Optimization with Financial Applications*, volume 61 of *Stochastic Modelling and Applied Probability*. Springer, Berlin, 2009.

[37] H. Pham and X. Warin. Actor-critic learning algorithms for mean-field control with moment neural networks. *Methodology and Computing in Applied Probability*, 27:13, 2025.

[38] H. Pham, X. Warin, and M. Germain. Neural networks-based backward scheme for fully nonlinear PDEs. *SN Partial Differential Equations and Applications*, 2(1):16, 2021.

[39] A. Picarelli and C. Reisinger. Probabilistic error analysis for some approximation schemes to optimal control problems. *Systems & Control Letters*, 137:104619, 2020.

[40] A. Picarelli, M. Scaratti, and J. Tam. Extended mean field control: A global numerical solution via finite-dimensional approximation. *arXiv preprint arXiv:2503.20510*, 2025.

[41] PyTorch Contributors. ReduceLROnPlateau, 2026. PyTorch 2.12 documentation.

[42] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[43] J. Sirignano and K. Spiliopoulos. DGM: A deep learning algorithm for solving partial differential equations. *Journal of Computational Physics*, 375:1339–1364, 2018.

[44] H. M. Soner, J. Teichmann, and Q. Yan. Learning algorithms for mean field optimal control. *arXiv preprint arXiv:2503.17869*, 2025.

[45] W. Stannat and A. Vogler. Approximation of optimal feedback controls for stochastic reaction–diffusion equations. *ESAIM: Control, Optimisation and Calculus of Variations*, 31:6, 2025.

[46] W. Stannat, A. Vogler, and L. Wessels. Neural network approximation of optimal controls for stochastic reaction–diffusion equations. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 33(9):093118, 2023.

[47] D. Talay and L. Tubaro. Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic Analysis and Applications*, 8(4):483–509, 1990.

[48] U. Tanielian and G. Biau. Approximating Lipschitz continuous functions with GroupSort neural networks. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 442–450. PMLR, 2021.

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

[50] X. Warin. Gas storage hedging. In *Numerical Methods in Finance: Bordeaux, June 2010*, pages 421–445. Springer, 2012.

[51] X. Warin. Reservoir optimization and machine learning methods. *EURO Journal on Computational Optimization*, 11:100068, 2023.

[52] J. Yong and X. Y. Zhou. *Stochastic Controls: Hamiltonian Systems and HJB Equations*, volume 43 of *Applications of Mathematics*. Springer, New York, 1999.

[53] M. Zhou, J. Han, and J. Lu. Actor-critic method for high dimensional static Hamilton–Jacobi–Bellman partial differential equations based on neural networks. *SIAM Journal on Scientific Computing*, 43(6):A4043–A4066, 2021.

A Technical results

A.1 Proof of Lemma 4.4

Proof. We show the proof for $p = 2$; the general case $p \geq 2$ follows from the same argument, using the discrete-time Burkholder-Davis-Gundy inequality in L^p. This is a standard moment estimate (see, for instance, [31, Theorem 4.5.4]), and we report an adapted proof for the discrete-time setting, to make explicit the constant C that appears in the estimate. By considering the recursion for the process $x_k := \widetilde{X}^{x,\alpha}_{t_k}$, we get

$$\mathbb{E}\Big[\max_{0\leq k\leq n}|x_k - x|^2\Big] \leq 2\mathbb{E}\Big[\max_{0\leq k\leq n}\Big|\sum_{i=0}^{k-1} b_i(x_i,\alpha_i(x_i))\tau\Big|^2\Big] + 2\mathbb{E}\Big[\max_{0\leq k\leq n}\Big|\sum_{i=0}^{k-1}\sigma_i(x_i,\alpha_i(x_i))\sqrt{\tau}Z_{i+1}\Big|^2\Big].$$

For the first term on the right-hand side, we get

$$\begin{aligned}\mathbb{E}\Big[\max_{0\leq k\leq n}\Big|\sum_{i=0}^{k-1} b_i(x_i,\alpha_i(x_i))\tau\Big|^2\Big] &\leq \tau n\sum_{i=0}^{n-1}\mathbb{E}[|b_i(x_i,\alpha_i(x_i))|^2]\tau \\ &\leq \tau n\sum_{i=0}^{n-1}\left(2[b]_3^2 + 2[b]_1^2\mathbb{E}[|x_i|^2]\right)\tau,\end{aligned}$$

where $[b]_3$ is a positive constant such that $\sup_{t\in[0,T]}|b(t,0,a)| \leq [b]_3$ for all $a \in A$; such a constant exists by Assumption 2.1. For the second term, instead, we take into account the Burkholder-Davis-Gundy inequality in the discrete time setting, and similarly to before, we obtain

$$\begin{aligned}\mathbb{E}\Big[\max_{0\leq k\leq n}\Big|\sum_{i=0}^{k-1}\sigma_i(x_i,\alpha_i(x_i))\sqrt{\tau}Z_{i+1}\Big|^2\Big] &\leq 4\sum_{i=0}^{n-1}\mathbb{E}[|\sigma_i(x_i,\alpha_i(x_i))|^2]\tau \\ &\leq 4\sum_{i=0}^{n-1}\left(2[\sigma]_3^2 + 2[\sigma]_1^2\mathbb{E}[|x_i|^2]\right)\tau,\end{aligned}$$

where $[\sigma]_3 > 0$ is such that $|\sigma(0,a)| \leq [\sigma]_3$, for all $a \in A$.

Define now $\Gamma_n := \mathbb{E}[\max_{0\le k\le n}|x_k - x|^2]$, and observe that $\mathbb{E}[|x_i|^2] \le 2\Gamma_i + 2|x|^2$. Then we have obtained the following

$$\Gamma_n \le 4[b]_3^2(\tau n)^2 + 16[\sigma]_3^2(\tau n) + 8([b]_1^2\tau n + 4[\sigma]_1^2)\sum_{i=0}^{n-1}\Gamma_i\tau + 8([b]_1^2\tau n + 4[\sigma]_1^2)|x|^2\tau n$$
$$\le ((4[b]_3^2T + 16[\sigma]_3^2) + (8[b]_1^2 + 4[\sigma]_1^2)|x|^2)\tau n + 8([b]_1^2T + 4[\sigma]_1^2)\sum_{i=0}^{n-1}\Gamma_i\tau.$$

Finally, by a discrete Gronwall's lemma, we get

$$\Gamma_n \le (C_1 + C_2|x|^2)\tau n \cdot e^{C_3\tau n},$$

where $C_1 := 4[b]_3^2T + 16[\sigma]_3^2$, $C_2 := 8[b]_1^2 + 4[\sigma]_1^2$ and $C_3 := 8([b]_1^2T + 4[\sigma]_1^2)$. We conclude, taking $C := \max\{C_1, C_2, C_3\}$ that

$$\mathbb{E}\Big[\max_{0\le k\le n}|x_k - x|^2\Big] \le C(1+|x|^2)Te^{CT}.$$

□

A.2 Density argument with respect to Radon measures

Lemma A.1. *Let μ be a finite Radon measure on $\mathbb{R}^d$ and let $A \subset \mathbb{R}^k$ be non-empty, compact, and convex. Then, for every Borel measurable function $\phi : \mathbb{R}^d \to A$ and every $\delta > 0$, there exists a Lipschitz continuous function $\phi^\delta : \mathbb{R}^d \to A$ such that*

$$\int_{\mathbb{R}^d} |\phi^\delta(x) - \phi(x)|^2 \,\mu(\mathrm{d}x) < \delta.$$

Proof. Since A is compact, ϕ is bounded and therefore belongs to $L^2(\mu;\mathbb{R}^k)$, where

$$L^2(\mu;\mathbb{R}^k) := \Big\{\varphi : \mathbb{R}^d \to \mathbb{R}^k \text{ measurable } \Big|\ \|\varphi\|_{L^2(\mu)}^2 := \int_{\mathbb{R}^d}|\varphi(x)|^2\mu(\mathrm{d}x) < \infty\Big\}.$$

By the density of continuous compactly supported functions in $L^2(\mu;\mathbb{R}^k)$ (cf. [18, Prop. 7.9]), there exists $h \in C_c(\mathbb{R}^d;\mathbb{R}^k)$ such that

$$\int_{\mathbb{R}^d} |h(x) - \phi(x)|^2 \,\mu(\mathrm{d}x) < \frac{\delta}{4}.$$

In addition, by a standard mollification argument, we can choose $h^\delta \in C_c^\infty(\mathbb{R}^d;\mathbb{R}^k)$, hence globally Lipschitz, such that

$$\int_{\mathbb{R}^d} |h^\delta(x) - h(x)|^2 \,\mu(dx) < \frac{\delta}{4}.$$

To conclude the proof, it is enough to observe that, A being closed and convex, the projection $\Pi_A : \mathbb{R}^k \to A$ is well-defined and 1-Lipschitz. Therefore, $\phi^\delta := \Pi_A \circ h^\delta$ is Lipschitz continuous and takes values in A. Moreover, since $\phi(x) \in A$ for every $x \in \mathbb{R}^d$, we have

$$|\phi^\delta(x) - \phi(x)| = |\Pi_A(h^\delta(x)) - \Pi_A(\phi(x))| \le |h^\delta(x) - \phi(x)|.$$

Consequently,

$$\int_{\mathbb{R}^d} |\phi^\delta(x) - \phi(x)|^2 \,\mu(\mathrm{d}x) \le \int_{\mathbb{R}^d} |h^\delta(x) - \phi(x)|^2 \,\mu(\mathrm{d}x)$$
$$\le 2\Big(\int_{\mathbb{R}^d} |h^\delta(x) - h(x)|^2 \,\mu(\mathrm{d}x) + \int_{\mathbb{R}^d} |h(x) - \phi(x)|^2 \,\mu(\mathrm{d}x)\Big) < \delta.$$

□

B Explicit computations for Example 6.1

B.1 Radial reduction and exact solution.

Define the radius $R_s := |X_s|$. By Itô's formula, we get a finite variation evolution for the square of the radius,

$$\mathrm{d}R_s^2 = 2\langle X_s,\, \alpha_s\rangle \mathrm{d}s + R_s^2 \mathrm{d}s, \quad R_t^2 = |x|^2. \tag{30}$$

Thus, let $r := |x|$ and define $v(t,r)$ as the radial reduction of the HJB. For $r = |x| > 0$, the radial derivatives satisfy

$$\nabla V(t,x) = \frac{\partial v}{\partial r}\nabla r(t,x) = v_r(t,|x|)\frac{x}{|x|} = v_r(t,|x|)\frac{x}{r},$$
$$\nabla^2 V(t,x) = v_{rr}(t,|x|)\frac{xx^\top}{r^2} + \frac{v_r}{r}(t,|x|)\left(I_2 - \frac{xx^\top}{r^2}\right),$$

where v_r and v_{rr} denote the partial derivatives of v with respect to the radial variable. Also, $\mathrm{Tr}[\sigma\sigma^\top(t,x)\nabla^2 V(t,x)] = r v_r(t,|x|)$ and we derive the HJB equation for v, in a viscosity sense,

$$\begin{cases} \partial_t v(t,r) + \frac{1}{2} r v_r(t,r) - M|v_r(t,r)| = 0, & (t,r) \in [0,T)\times\mathbb{R}_+ \\ v(T,r) = (r - r_0)^2. \end{cases}$$

Whenever $x \neq 0$ and $v_r(t,|x|) \neq 0$, the optimal feedback selection of (21) is then given by $\alpha^*(t,x) = -M\frac{x}{|x|}\mathrm{sgn}(v_r(t,|x|))$. Inside the zero-level set, v is locally flat, and the optimal controls are generally non-unique.

B.2 Backward reachable set.

We derive an explicit characterization of the boundary curves of the backward reachable set and the associated value function. According to (30), $R_s := |X_s| = \sqrt{R_s^2}$ satisfies the finite-variation dynamics

$$\mathrm{d}R_s = \left(\frac{R_s}{2} + u_s\right)\mathrm{d}s, \qquad u_s := \left\langle \alpha_s,\, \frac{X_s}{|X_s|}\right\rangle,$$

where u_s represents the radial component of the control. All expressions involving $X_s/|X_s|$ are meant for $X_s \neq 0$, and the case $X_s = 0$ is treated by continuity in the radial variable.

The boundary curves of the backward reachable set are obtained by imposing the extremal admissible radial controls. The lower boundary corresponds to the maximal outward radial control $u \equiv M$, while the upper boundary corresponds to the maximal inward radial control $u \equiv -M$. The two curves are then expressed as solutions of the following linear ODEs:

$$\begin{cases} \dot\rho_-(t) = \frac{1}{2}\rho_-(t) + M, & \rho_-(T) = r_0, \\ \dot\rho_+(t) = \frac{1}{2}\rho_+(t) - M, & \rho_+(T) = r_0, \end{cases} \qquad \begin{array}{ll} \text{thus} & \rho_-(t) = -2M + (r_0 + 2M)e^{(t-T)/2}; \\ \text{thus} & \rho_+(t) = 2M + (r_0 - 2M)e^{(t-T)/2}. \end{array}$$

It follows that the zero-level set boundaries are explicitly characterized through $\rho_-(t)$ and $\rho_+(t)$,

$$\mathcal{Z}_t = \{x \in \mathbb{R}^2 \,:\, \max\{0, \rho_-(t)\} \le |x| \le \rho_+(t)\},$$

and the explicit formula for the value function is

$$V(t,x) = \begin{cases} e^{T-t}(\rho_-(t) - |x|)^2, & \rho_-(t) > 0, \quad 0 \le |x| < \rho_-(t), \\ 0, & x \in \mathcal{Z}_t, \\ e^{T-t}(|x| - \rho_+(t))^2, & |x| > \rho_+(t). \end{cases}$$

Moreover, outside the zero-level set, the sign of v_r is determined by the relative position of r with respect to the reachable interval. More precisely, $v_r(t,r) < 0$ if $r < \rho_-(t)$, and $v_r(t,r) > 0$ if $r > \rho_+(t)$. Hence, whenever $x \notin \mathcal{Z}_t$ and $x \neq 0$, an optimal feedback is given by

$$\alpha^*(t,x) = \begin{cases} +M\frac{x}{|x|}, & |x| < \max\{0, \rho_-(t)\}, \\ -M\frac{x}{|x|}, & |x| > \rho_+(t), \end{cases}$$

i.e., if $|x| > \rho_+(t)$, the control acts inward, decreasing the radius as much as possible; if $|x| < \rho_-(t)$, it acts outward, increasing the radius as much as possible.

C Explicit weak order 2.0 scheme

Let $(W_t)_{t\geq 0}$ an m-dimensional standard Brownian motion, we consider a d-dimensional controlled stochastic differential equation of the form

$$\mathrm{d}X_t = b(t, X_t, \alpha_t)\mathrm{d}t + \sigma(t, X_t, \alpha_t)\mathrm{d}W_t, \quad X_0 = \xi, \quad t \in [0,T],$$

for which we assume the control $\alpha_t = \alpha(t, X_t)$ in feedback-form. For notational simplicity, we denote the discretized version, corresponding to our discrete-time setting in Section 3.1, by $(Y_n)_{n=0}^N$, where the size of the time step is $\Delta t := T/N$ and $\Delta W_n := \sqrt{\Delta t} Z_n$, with Z_n i.i.d. $\sim \mathcal{N}(0_m, \mathrm{I}_m)$, representing the Brownian increment.

We consider explicit scheme proposed by Kloeden and Platen and which was originally formulated for autonomous SDEs ([31, eq. (15.1.3)]). Since the optimal control problems under consideration in this work are inherently non-autonomous, the coefficients may be time-dependent, and the feedback control policy $\alpha_t = u(t, X_t)$ introduces an explicit dependence on the temporal variable t. Hence we treat time as an additional state variable by defining an augmented state vector $\bar{X}_s = (s, X_s)^\top$. This leads to the following expressions, for Example 1 (Section 6.1):

$$\begin{aligned} Y_{n+1} \;=\; & Y_n + \frac{1}{2}\Big(b(t_n, Y_n, \alpha_n) + b(t_{n+1}, \bar{Y}_{n+1}, \bar{\alpha}_{n+1})\Big)\Delta t \\ & + \frac{1}{4}\Big(\sigma(Y^+_{n+1}) + \sigma(Y^-_{n+1}) + 2\sigma(Y_n)\Big)\Delta W_{n+1} \\ & + \frac{1}{4}\Big(\sigma(Y^+_{n+1}) - \sigma(Y^-_{n+1})\Big)\big((\Delta W_{n+1})^2 - \Delta t\big)\Delta t^{-1/2} \end{aligned} \tag{31}$$

with

$$\begin{aligned} \bar{Y}_{n+1} &:= Y_n + b(t_n, Y_n, \alpha_n)\Delta t + \sigma(Y_n)\Delta W_{n+1}, \\ Y^\pm_{n+1} &:= Y_n + b(t_n, Y_n, \alpha_n)\Delta t \pm \sigma(Y_n)\sqrt{\Delta t}, \\ \alpha_n &:= \alpha(t_n, Y_n), \qquad \bar{\alpha}_{n+1} := \alpha(t_{n+1}, \bar{Y}_{n+1}) \end{aligned}$$

and, for Example 2, with σ constant (Section 6.2):

$$\begin{aligned} Y_{n+1} &= Y_n + \frac{1}{2}\Big(b(\alpha(t_{n+1}, \bar{Y}_{n+1})) + b(\alpha(t_n, Y_n))\Big)\Delta t + \sigma\Delta W_{n+1}, \\ \bar{Y}_{n+1} &= Y_n + b(\alpha(t_n, Y_n))\Delta t + \sigma\Delta W_{n+1}. \end{aligned} \tag{32}$$