Master's thesis

2023

Theophile Legrand

Master's thesis

**NTNU**
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Theophile Legrand

# On the convergence rate for piece-wise constant policy approximation of stochastic optimal control problem

.

June 2023

NTNU
Norwegian University of
Science and Technology

Norwegian University of
Science and Technology

# On the convergence rate for piece-wise constant policy approximation of stochastic optimal control problem

.

**Theophile Legrand**

MA3911
Submission date: June 2023
Supervisor: Espen Robstad Jakobsen
Co-supervisor: .

Norwegian University of Science and Technology
Department of Mathematical Sciences

**Abstract**

The goal of this thesis is to build on the method and result of [6] to improve the bound between the optimal value function and the piece-wise constant control one in stochastic control settings by exploring the properties of semi-concavity and semi-convexity for the data. A significant part of the work was to familiarise ourselves with the fundamental aspect of optimal control theory and controlled differential equation theory. Some very basic properties are demonstrated in this way, because the author felt that these classical considerations would be helpful in understanding the whole work.

In that regard, we show how to reach the optimal error bound in two different settings before tackling the main interest. These setting are the Stochastic smooth one and the deterministic lipschitz one. We then show that while semi-concavity or semi-convexity alone are not sufficient to improve the bound, the combination of the two yields a $1/3$ bound. This improves the $1/4$ bound found in [6] at the cost of several new stringent hypotheses. Later we present an analysis of the schemes developed in [10] and propose a new numerical experiments to show specifically the error bound induced by the time discretisation of the control. Finally some numerical experiments are conducted to show the effectiveness of the approach.

# Contents

# 1 Introduction

In this thesis, we derive error estimates of the value function in a stochastic or deterministic control problem for a different setting not found in the literature. This Introduction section will define the general problem, review the existing research, explain the goal of this work and finally state and discuss the result.

## 1.1 Setting

This section defines the stochastic control problem in the general case. Some parts of this definition will differ from between the deterministic and stochastic case but they will be highlighted. We are considering coefficients and cost functions that are independent of time.

Let $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mathbb{P})$ be a complete filtered space, $(W_t)_{t\geq 0}$ a $p$-dimensional $(\mathcal{F}_t)_{t\geq 0}$-Wiener process on $(\Omega, \mathcal{F}, \mathbb{P})$ and $\mathcal{A}$ the space of the controls with values in $\mathrm{A} \subseteq \mathbb{R}^m$ that are progressively measurable over $[0, T]$.
For $\alpha \in \mathcal{A}$ and $x \in \mathbb{R}^d$ define $X_\cdot = X_\cdot^{\alpha,x}$ the Ito diffusion satisfying:

$$X_t = x + \int_0^t b_{\alpha_r}(X_r)dr + \int_0^t \sigma_{\alpha_r}(X_r)dW_r \quad \text{for} \quad t \in [0, T]. \tag{1.1}$$

For a given terminal cost function $g$, running cost $f$ and a starting point $(t, x) \in [0, T] \times \mathbb{R}^d$, the optimal control problem consists of maximizing over $\alpha \in \mathcal{A}$ the total cost $J^\alpha$:

$$J^\alpha(t, x) = \mathbb{E}_x^\alpha \left[ \int_0^{T-t} f_{\alpha_r}(X_r)dr + g(X_{T-t}) \right]. \tag{1.2}$$

Hence we define the corresponding value function for $t \in [0, T]$ and $x \in \mathbb{R}^d$.
Define $v_h \colon [0, T] \times \mathbb{R}^d \longrightarrow \mathbb{R}$ the value function,

$$v(t, x) = \sup_{\alpha \in \mathcal{A}} J^\alpha(t, x). \tag{1.3}$$

We aim to estimate the error introduced by approximating the set of measurable controls $\mathcal{A}$ by piece-wise constant controls. Let $h > 0$ be the discretization parameter and $\mathcal{A}_h$ the subset of $\mathcal{A}$ of processes that are constant in the intervals $[nh, (n+1)h)$ for $n \in \mathbb{N}$. The value function associated with this restricted set of controls is defined by:

$$v_h(t, x) = \sup_{\alpha \in \mathcal{A}_h} J^\alpha(t, x). \tag{1.4}$$

## 1.2 State of the art

This approach involving approximation of the solution of the equation by considering only the piece-wise constant control policies is common in control problems analysis because of it's inherent link to numerical analysis. When one need to calculate the numerical solution of an equation, the calculation is done time step wise in a computer, hence the need to first study solutions that only consider piece wise constant data. This is similar to studying convergence and consistency in classical differential equations. It has been used time and again to numerically solve Hamilton Jacobi equation (see in [11], [4], [3]) because these control problems are rarely explicitly solvable. Since it is known that the value function of this stochastic control problem is also the solution of the second order parabolic Hamilton-Jacobi-Bellman equation (1.5), with $L_a v = b_a v_x + \frac{1}{2} tr(\sigma_a \sigma_a^T v_{xx})$ the infinitesimal generator of the Ito diffusion, we will refer to papers that have been written to extend the literatures of both domains.

$$\begin{cases} v_t + \sup\limits_{a \in \mathrm{A}} [L_a v + f_a] = 0 \\ v(x, T) = g(x) \qquad \forall x \in \mathbb{R}^d \end{cases} \tag{1.5}$$

In this context, the study of the convergence and convergence rate of this piece wise constant policies value function to the optimal one arises naturally and in the context of deterministic controlled differential equation and first order HJB equation, the matter has been studied for a long time [4, see]. For the controlled diffusion case, N.V. Krylov show in [8] that the order $\frac{1}{3}$ can be reached in the case of coefficients independant of time and space and $h^{\frac{1}{6}}$ in general. The settings ask for Hölder continuous coefficients. He further improves these results in [7] by showing that only a $\frac{1}{2}$-Hölder continuous in time hypothesis was necessary. To derive this result he uses a so-called 'shaking of coefficients' technique which allows to regularize the data and the value function while keeping some properties of the original value function and the data. We will explain and show the use of the shaking of coefficients here but the original results can be found in [9]. In [6] Espen R. Jakobsen, Athena Picarelli and Christoph Reisinger improve the bound to $\frac{1}{4}$ with the same setting. Besides the improvement, this result is important because it aligns the convergence rate with those calculated by Partial Differential Equation methods only in a similar setting ([2, see]).

Piece-wise constant policy approximation has been to develop numerical methods to solve second order HJB equation. Particularly to develop semi-Lagrangian schemes in [11] and semi-Lagrangian like schemes in [10]. The later work is interesting in that it reaches this bound via purely probabilistic methods, using a Markov chain approximation of the diffusion whereas in other work, differential equation methods were often prefered. Both of these schemes reached an order $\frac{1}{4}$ in time and $\frac{1}{5}$ in space for the general lipschitz coefficients and data case. In their setting, the piece-wise constancy of the policies come from the discretisation of the time interval. Both these schemes are also monotonic. This is an important feature because Barles and Souganidis [5] shows that it insures the convergence to the correct solution. In both work, the order calculated is not sharp in their application, indeed the numerical tests shows order higher than one in most cases, even in the less smooth ones. This is because the very general settings do not consider that most application have solutions at least piece-wise more smoother than asked.

## 1.3 Results and discussion

The main contributions in this thesis are the consideration of semi-concave and semi-convex data in the stochastic setting as well as proposing a new scheme aimed at showing numerically that the order of the calculated bound can be reached. In [3] I. Capuzzo Dolcetta and H. Ishii show that in a slightly different deterministic setting and with semi-concavity of the costs, an order one of the error can be derived. This hints at a possible improvement of the bound in the stochastic backward time setting. In the thesis we show that the order $\frac{1}{3}$ can be achieved with sensibly similar hypotheses:

$$0 \leq v(t,x) - v_h(t,x) \leq Ch^{\frac{1}{3}}, \quad \forall t \in [0,T],\ x \in \mathbb{R}^d.$$

However, in this stochastic setting with maximisation over the space of controls, and because of the backward time setting, the assumptions for obtaining an improvement in the boundary are much stricter. Indeed, as we will see it is required to have bounded second order finite difference approximation of both coefficients and the end cost (what will sometimes be referred to as both semi-concave and semi-convex) as well as semi-convex running cost. We also show the order $\frac{1}{2}$ in the general deterministic case and the order 1 in case of sufficiently smooth solution to the problem.

To test the result we will use the semi lagragian scheme developed in [10] however, as we stated before, the scheme has order $\frac{1}{4}$ in time and in all the research interested in solving HJB equations, the time during which a control is constant is the same one as the discretisation time. This is because in term of efficiency it would not make sense to differentiate both. However, with this coupling of $h$ and $\Delta t$ it is impossible to show the calculated order $\frac{1}{3}$ if we choose a scheme that has order $\frac{1}{4}$. In trying to solve this problem, we will propose a new scheme where $h$ the time over which the controls are constant is equal to several time interval in a row so that the error from the constancy of the control in differentiated from the error of the Euler approximation of the diffusion. The tests and explanations will take place in the numerical tests part 6.

This new scheme, although much less efficient speed and error rate wise, shows the rate of the calculated bound in term of piece-wise constant policy approximation only. This is a new result that is unseen elsewhere mainly because it did not appear in any real problem which context would led to optimal control problematisation.

## 1.4 Organisation

The organisation is as follows. In the next section 2, we state and show some basic results including well-posedness and the dynamic programming principle, and define some of the tools we will use throughout all of the sections (mollification, shaking of coefficients). In section 3, we derive the $h$ bound for the smooth case as an introduction to the general technique we will use in the following sections. In section 4 we derive the $h^{\frac{1}{2}}$ bound for the general case but in a general deterministic setting. Finally in section 5 we improve the result from [6] by introducing new hypothesis allowing semi concavity and semi convexity for the data. Section 6 demonstrates the improvement numerically using the method from [10] and section 7 concludes. The code can be found in the appendix A.

## 2 Preliminaries

### 2.1 Assumptions and well-posedness

We use the following notation $\varphi_a(.,.) = \varphi(a,.,.)$ for the control values $a \in \mathrm{A}$. Moreover for simplicity of notations, the norm $|.|$ denotes either the euclidean norm in $\mathbb{R}^d$ or the spectral norm in $\mathbb{R}^{d\times p}$ if we are dealing with matrix instead.

We will consider these assumptions:

**(H1)** A is a compact set;

**(H2)** $b\colon \mathbb{R}^d \times \mathrm{A} \longrightarrow \mathbb{R}^d$ and $\sigma\colon \mathbb{R}^d \times \mathrm{A} \longrightarrow \mathbb{R}^{d\times p}$ are continuous functions. There exists $C_0 > 0$ such that, for $\varphi \in \{b, \sigma\}$,

$$\forall x, y \in \mathbb{R}^d, \forall a \in \mathrm{A},\ \ |\varphi_a(x) - \varphi_a(y)| \leq C_0|x-y| \text{ and } , |\varphi_a(x)| \leq C_0; \tag{H2}$$

**(H3)** $g : \mathbb{R}^d \longrightarrow \mathbb{R}$ and $f : \mathrm{A} \times \mathbb{R}^d \longrightarrow \mathbb{R}$ are continuous functions. There exists $C_1 > 0$ such that,

$$\forall x, y \in \mathbb{R}^d, \forall a \in \mathrm{A},\ \ \begin{cases} |g(x) - g(y)| \leq C_1|x-y| \\ |f_a(x) - f_a(y)| \leq C_1|x-y| \\ |f_a(x)| \leq C_1; \end{cases} \tag{H3}$$

**(H4)** There exists $C_2 > 0$ such that, for $\varphi \in \{b, \sigma\}$,

$$\forall x, y \in \mathbb{R}^d, \forall a \in \mathrm{A},\ \ |\varphi_a(x+y) + \varphi_a(x-y) - 2\varphi_a(x)| \leq C_2|x-y|^2 \tag{H4}$$

**(H5)** There exists $C_3 > 0$ such that,

$$\forall x, y \in \mathbb{R}^d, \forall a \in \mathrm{A},\ \ \begin{cases} f(x+y) + f(x-y) - 2f(x) \geq -C_2|x-y|^2 \\ |g(x+y) + g(x-y) - 2g(x)| \leq C_2|x-y|^2 \end{cases} \tag{H5}$$

Under (H1) and (H2), for any $\alpha \in \mathrm{A}$ there exists a unique strong solution to 1.1. These two alongside (H3) will be considered valid throughout all the sections, they are necessary to have the value function also be lipschitz in space and $\frac{1}{2}$-Hölder in time. The last two hypothesis allow the value function to also be semi-convex in space but also to have better time regularity which are the main two arguments in Section 5 to improve the bound.

### 2.2 Mollification

This subsection introduces the $(t,x)$-mollifier which will be used throughout the different sections. The deterministic and the stochastic case do not require the same degree of mollification for t so we will define $p \in \mathbb{N}$. This is because of the propagation of regularities, indeed because of the Ito isometry the resulting regularities of the value function are not the same in deterministic and stochastic settings. In stochastic, the data even if they were lipschitz in time would only yield a 1/2-Hölder regularity in time while in deterministic in would be lipschitz in time. Hence, we consider different degree of mollification in time for Section 4 and 5.
Let $\varphi : [0,T] \times \mathbb{R}^d \longrightarrow \mathbb{R}$ a measurable function.

**Definition 2.1.** Define $\rho_\varepsilon\colon \mathbb{R}^{d+1} \longrightarrow \mathbb{R}$, the $x$ and $t$-mollifier:

$$\rho_\varepsilon(t,x) = \frac{1}{\varepsilon^{d+p}}\rho(\frac{t}{\varepsilon^p}, \frac{x}{\varepsilon}),$$

where $\rho\colon \mathbb{R}^{d+1} \longrightarrow \mathbb{R}$ is a standard mollifier:

$$\rho \in C^\infty(\mathbb{R}^{d+1}),\ \rho > 0,\ supp(\rho) = (0,1) \times \{|x| < 1\},\ \int_{supp(\rho)} \rho(e)de = 1.$$

**Definition 2.2.** Define $\varphi$'s mollified version, $\varphi^{(\varepsilon)}[0,T] \times \mathbb{R}^d \longrightarrow \mathbb{R}$:

$$\varphi^{(\varepsilon)}(t,x) = \int_{|y|<\varepsilon}\int_{0\leq s\leq \varepsilon^p} \varphi(t-s, x-y)\rho_\varepsilon(s,y)dsdy.$$

**Proposition 2.1.** *We get standard mollification results for* $\varphi \in C_b^{\frac{1}{p},1}([0,T]\times\mathbb{R}^d)$ *:*

$$\varphi^{(\varepsilon)} \in C^\infty([0,T]\times\mathbb{R}^d),\ \|\varphi-\varphi^{(\varepsilon)}\|_\infty \le C\varepsilon,\ \text{and}\ \|\partial_t^m D_x^k \varphi^{(\varepsilon)}\|_\infty \le C\varepsilon^{1-pm-k},\ \forall m+k\ge 1.$$

*Proof.* These are standard results, the first one is du to the convolution being as differentiable as the two convoluted function (and $\rho_\varepsilon \in C^\infty(\mathbb{R}^{d+1})$). The second one can be derived using Taylor expansion with integral reminder inside the convolution integration. The last one is due to $\varphi \in C_b^{1/2,1}([0,T]\times\mathbb{R}^d)$ (this means that it's $\frac{1}{2}$-Hölder in time and lipschitz in space so we can bound 'half' of a derivative in time and one derivative in space). One can look at the proof of Property 5.2 for an idea of the proof of this last property. One can also find in this in [6]. □

To mollify the data (that do not depend on time) one can consider them as function of time but constant in time and the results would still hold. Moreover, checking Definition 2.2, we would clearly have that the mollified version won't depend on time either.

Finally, we will have to mollify $b$ and $\sigma$, which map $\mathbb{R}^d$ to $\mathbb{R}^d$. To do so, one can mollify each of it's coordinates and would get the same results.

## 2.3 Dynamic programming principle

The first but major tool we will need is the Dynamic Programming Principle (DPP) it is a useful way to compute the value function in a backward in time optimisation. It makes use of the constancy of the controls over $[0,h$ to show that the sub-optimal value function $v_h$ satisfies a Bellman equation. It can also be a step of the proof showing that $v$ the optimal value function satisfies the Hamilton Jacobi Bellman equation. The proof is inspired from [12] chapter 5, where some similar dynamic programming principle is shown but for a deterministic setting.

**Proposition 2.2** (Dynamic programming principle)**.** *Let* $s\in[0,T-h]$, $x\in\mathbb{R}^d$ *the point at which we want to calculate the value function.* $v_h$ *satisfies the following Dynamic Programming Principle:*

$$v_h(s,x) = \sup_{a\in \mathrm{A}} \mathbb{E}\left[\int_0^h f_a(X_r^{(x,a)})dr + v_h(s+h, X_h^{(x,a)})\right].$$

*Proof.* Let $s,x$ such that they satisfy the previous requirements and $a\in\mathrm{A}$. Let $X_\cdot = X_\cdot^{a,x}$ the solution to the classic problem for this constant control.
**a) First inequality:** Let $\delta>0$ and $\alpha^1\in\mathcal{A}_h$, a $\delta$-optimal control for $v_h(s+h,X_h)$,

$$v_h(s+h,X_h)-\delta \le \mathbb{E}\left[\int_0^{T-s-h} f_{\alpha_r^1}(X_r^1)dr + g(X_{T-s-h}^1)\right].$$

Where $X_\cdot^1 = X_\cdot^{\alpha^1,X_h}$, note that we take $X_h$ as the origin. This $\delta$-optimal control exists because $v_h$ is a sup.
Define $\bar\alpha\colon[0,T]\longrightarrow\mathbb{R}$ the following new truncated control:

$$\bar\alpha_t = \begin{cases} a & \text{for } t\le h\\ \alpha_{t-h}^1 & \text{for } t>h; \end{cases}$$

Now, defining $\bar X_\cdot = X_\cdot^{\bar\alpha,x}$, we have that for $t\in[0,h]$, $\bar X_t \overset{\text{a.s.}}{=} X_t$ because they satisfy the same equation: $Y_t = x+\int_0^t b_a(Y_r)dr + \int_0^t \sigma_a(Y_r)dW_r$, $\forall t\in[0,h]$ which has lipschitz data.
For $t>h$ via a change of variable $r=q-h$ and because $\bar X_h \overset{\text{a.s.}}{=} X_h$ and the data are bounded,

$$\begin{aligned} X_{t-h}^1 &= X_h + \int_0^{t-h} b_{\alpha_r^1}(X_r^1)dr + \int_0^{t-h}\sigma_{\alpha_r^1}(X_r^1)dW_r\\ &\overset{\text{a.s.}}{=} \bar X_h + \int_h^t b_{\alpha_{q-h}^1}(X_{q-h}^1)dq + \int_h^t \sigma_{\alpha_{q-h}^1}(X_{q-h}^1)dW_r\\ &\overset{\text{a.s.}}{=} \bar X_h + \int_h^t b_{\bar\alpha_q}(X_{q-h}^1)dq + \int_h^t \sigma_{\bar\alpha_q}(X_{q-h}^1)dW_r. \end{aligned}$$

Which is the same equation $\bar{X}$ satisfies on $t > h$:

$$\begin{aligned}\bar{X}_t &= x + \int_0^t b_{\bar{\alpha}_r}(\bar{X}_r)dr + \int_0^t \sigma_{\bar{\alpha}_r}(\bar{X}_r)dW_r \\ &= x + \int_0^h b_{\bar{\alpha}_r}(\bar{X}_r)dr + \int_0^h \sigma_{\bar{\alpha}_r}(\bar{X}_r)dW_r + \int_h^t b_{\bar{\alpha}_r}(\bar{X}_r)dr + \int_h^t \sigma_{\bar{\alpha}_r}(\bar{X}_r)dW_r \\ &= \bar{X}_h + \int_h^t b_{\bar{\alpha}_r}(\bar{X}_r)dr + \int_h^t \sigma_{\bar{\alpha}_r}(\bar{X}_r)dW_r.\end{aligned}$$

Hence, because they satisfy the exact same equation, with same constraints and lipschitz data:

$$\bar{X}_t \overset{\text{a.s.}}{=} X^1_{t-h} \ \forall t > h.$$

In conclusion, we have that:

$$\bar{X}_t \overset{\text{a.s.}}{=} \begin{cases} X_t & \text{for } t \in [0, h] \\ X^1_{t-h} & \text{for } t > h. \end{cases}$$

We have by definition of $v_h$, the almost sure equalities between the different diffusions and the $\delta$ optimal bound:

$$\begin{aligned} v_h(s,x) &\geq \mathbb{E}\left[\int_0^{T-s} f_{\bar{\alpha}_r}(\bar{X}_r)dr + g(\bar{X}_{T-s})\right] \\ &\geq \mathbb{E}\left[\int_0^h f_{\bar{\alpha}_r}(\bar{X}_r)dr + \int_h^{T-s} f_{\bar{\alpha}_r}(\bar{X}_r)dr + g(\bar{X}_{T-s})\right] \\ &\geq \mathbb{E}\left[\int_0^h f_a(X_r)dr + \int_h^{T-s} f_{\alpha^1_{r-h}}(X^1_{r-h})dr + g(X^1_{T-s-h})\right] \\ &\geq \mathbb{E}\left[\int_0^h f_a(X_r)dr + \int_0^{T-s-h} f_{\alpha^1_h}(X^1_h)dr + g(X^1_{T-s-h})\right] \\ &\geq \mathbb{E}\left[\int_0^h f_a(X_r)dr + v_h(s+h, X_h) - \delta\right]. \end{aligned}$$

Since this is true for all $a \in \mathrm{A}$ and for all $\delta > 0$, we have the first inequality:

$$v_h(s,x) \geq \sup_{a \in \mathrm{A}} \mathbb{E}\left[\left(\int_0^h f_a(X_r)dr + v_h(s+h, X_h)\right)\right].$$

**b) Second inequality:** For the second inequality, we similarly take $\alpha^2 \in \mathcal{A}_h$, another $\delta$-optimal control, but for $v_h(s,x)$. With $X^2_\cdot = X^{\alpha^2,x}_\cdot$,

$$\begin{aligned} v_h(s,x) - \delta &\leq \mathbb{E}\left[\int_0^{T-s} f_{\alpha^2_r}(X^2_r)dr + g(X^2_{T-s})\right] \\ &\leq \mathbb{E}\left[\int_0^h f_{\alpha^2_r}(X^2_r)dr + \int_h^{T-s} f_{\alpha^2_r}(X^2_r)dr + g(X^2_{T-s})\right]. \end{aligned}$$

For the second term, with a change of variable $q + h = r$, we have
$\int_h^{T-s} f_{\alpha^2_r}(X^2_r)dr = \int_0^{T-s-h} f_{\alpha^2_{q+h}}(X^2_{q+h})dq$. Define the translation $\bar{\alpha}_\cdot = \alpha^2_{\cdot+h}$ on $[0, T-h]$. Because they satisfy the same equation, we have $X^{\alpha^2,x}_{q+h} \overset{\text{a.s.}}{=} X^{\bar{\alpha},X^2_h}_q$ and,

$$\begin{aligned} \mathbb{E}\left[\int_0^{T-s-h} f_{\alpha^2_{q+h}}(X^{\alpha^2,x}_{q+h})dq + g(X^2_{T-s})\right] &= \mathbb{E}\left[\int_0^{T-s-h} f_{\bar{\alpha}_r}(X^{\bar{\alpha},X^2_h}_r)dr + g(X^{(\bar{\alpha},X^2_h)}_{T-s-h})\right] \\ &\leq \sup_{\alpha \in \mathcal{A}_h} \mathbb{E}\left[\int_0^{T-s-h} f_{\alpha_r}(X^{\alpha,X^2_h}_r)dr + g(X^{\alpha,X^2_h}_{T-s-h})\right] \\ &\leq \mathbb{E}\left[v_h(s+h, X^2_h)\right]. \end{aligned}$$

Since $\alpha^2 \in \mathcal{A}_h$, it is constant over $[0,h]$ and $X^2_r = X^{\alpha^2,x}_r = X^{\alpha^2_0,x}_r$ for $r \in [0,h]$.

$$\mathbb{E}\left[\int_0^h f_{\alpha^2_r}(X^2_r)dr + v_h(s+h, X^2_h)\right] = \mathbb{E}\left[\int_0^h f_{\alpha^2_0}(X^{\alpha^2_0,x}_r)dr + v_h(s+h, X^{\alpha^2_0,x}_h)\right]$$
$$\leq \sup_{a\in \mathrm{A}} \mathbb{E}\left[\int_0^h f_a(X^{a,x}_r)dr + v_h(s+h, X^{a,x}_h)\right].$$

Which proves the second inequality:

$$\begin{aligned} v_h(s,x) - \delta &\leq \mathbb{E}\left[\int_0^h f_{\alpha^2_r}(X^2_r)dr + \int_h^{T-s} f_{\alpha^2_r}(X^2_r)dr + g(X^2_{T-s})\right] \\ &\leq \mathbb{E}\left[\int_0^h f_{\alpha^2_r}(X^2_r)dr + v_h(s+h, X^2_h)\right] \\ &\leq \sup_{a\in \mathrm{A}} \mathbb{E}\left[\int_0^h f_a(X_r)dr + v_h(s+h, X_h)\right], \end{aligned}$$

and gives the dynamic programming principle.

□

In this proof we used only a very basic property of the sup, the uniqueness of the strong solution with the coefficients $b$ and $\sigma$ and the continuity of $f$ and $g$ hence it is easily extendable to similar control problems and particularly the following one which coefficient have been shaken.

## 2.4 Shaking of coefficients

To derive the piece-wise constant policies bound, we will need to use Ito's lemma on the value function, the data and coefficients but none of these are supposed to be two times differentiable. That is why need mollification however with plain mollification of the value function the Dynamic Programming Principle will not generally hold. To get around this problem, we introduce the following shaking of coefficients so that in the end, we get a mollified value function that still satisfies a one sided dynamic programming principle and that is close to the piece-wise constant policies one.

**Definition 2.3** (Solution to the mollified equation)**.** For $\alpha \in \mathcal{A}$, define $\tilde{X}_\cdot = \tilde{X}^{\alpha,x}_\cdot = \tilde{X}^x_\cdot$ as the solution to:

$$\tilde{X}_t = x + \int_0^t b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)dr + \int_0^t \sigma^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)dW_r,\ \forall t \in [0,T].$$

**Definition 2.4** (Value function)**.** Let $t \in [0,T]$ and $x \in \mathbb{R}^d$. Define $\tilde{J}^\alpha\colon [0,T]\times\mathbb{R}^d \longrightarrow \mathbb{R}$ the cost function for $\alpha \in \mathcal{A}$ and $\tilde{v}_h\colon [0,T]\times\mathbb{R}^d \longrightarrow \mathbb{R}$ the subsequent value function,

$$\tilde{J}^\alpha(t,x) = \mathbb{E}\left[\int_0^{T-t} f_{\alpha_r}(\tilde{X}_r)dr + g(\tilde{X}_{T-t})\right]$$

$$\tilde{v}(t,x) = \sup_{\alpha\in\mathcal{A}} J^\alpha(t,x).$$

Now we will define the shaken coefficients problem. First define $\mathcal{E}_h = \{\xi \in \mathcal{A}_h |\ \forall t \in \mathbb{R},\ \xi_t \in B_\varepsilon(0)\}$ where $B_\varepsilon(0)$ is the ball of radius $\varepsilon$ and centered at zero in $\mathbb{R}^d$.

**Definition 2.5** (Solution to the shaken coefficients equation)**.** For $\alpha \in \mathcal{A}_h$ and $\xi \in \mathcal{E}_h$, define $\hat{X}_t = \hat{X}^{\alpha,\xi,x}_t = \hat{X}^x_t$ as the solution to:

$$\hat{X}_t = x + \int_0^t b^{(\varepsilon)}_{\alpha_r}(\hat{X}_r + \xi_r)dr + \int_0^t \sigma^{(\varepsilon)}_{\alpha_r}(\hat{X}_r + \xi_r)dW_r,\ \forall t \in [0,T].$$

**Definition 2.6** (Shaken coefficient value function)**.** Let $t \in [0,T]$ and $x \in \mathbb{R}^d$.
Define $u_h\colon [0,T]\times\mathbb{R}^d \longrightarrow \mathbb{R}$

$$u_h(t,x) = \sup_{\alpha\in\mathcal{A}_h, \xi\in\mathcal{E}_h} \mathbb{E}\left[\int_0^{T-t} f_{\alpha_r}(\hat{X}_r)\,dr + g(\hat{X}_{T-t})\right].$$

We can show the same dynamic programming principle for this shaken coefficient value function since it's also defined as the optimal one for the piece-wise constant policies and it uses functions that still holds the properties of the ones from $v_h$.
We can now define and state the result for the mollified version of the shaken coefficients value function $u_h^{(\varepsilon)}$:

**Proposition 2.3** ($u_h^{(\varepsilon)}$ super dynamic programming principle)**.** *The function $u_h^{(\varepsilon)}$ belongs to $C^\infty([0,T]\times\mathbb{R}^d)$. There exists a constant $C>0$ such that: Moreover, $u_h^{(\varepsilon)}$ satisfies the following super-dynamic programming principle,*

$$u_h^{(\varepsilon)}(t,x) \geq \mathbb{E}\left[\int_0^h f_a^{(\varepsilon)}(\tilde{X}_r)dr + u_h^{(\varepsilon)}(t+h,\tilde{X}_h)\right],$$

*for any $a \in \mathrm{A}$, $t\in[0,T-h]$ and $x\in\mathbb{R}^d$.*

*Proof.* First we will derive a relation between $\hat{X}$ and $X$. Then we will prove a kind of super dynamic programming principle on $u_h$ using the former DPP and then we will "mollify" it to find the inequality we are after.
Let $(a,e)\in\mathcal{A}\times B_\varepsilon(0)$, $t\in[0,T-h]$, $x\in\mathbb{R}^d$ and $|s|\leq\varepsilon$.
Noticing that with a constant control,

$$\hat{X}_t^{(a,e),x} = x + \int_0^t b_a^{(\varepsilon)}(\hat{X}_r+e)dr + \int_0^t \sigma_a^{(\varepsilon)}(\hat{X}_r+e)dW_r$$

and,

$$\begin{aligned}\tilde{X}_t^{a,x+e} - e &= x + \int_0^t b_a^{(\varepsilon)}(\tilde{X}_r^{a,x+e})dr + \int_0^t \sigma_a^{(\varepsilon)}(\tilde{X}_r^{a,x+e})dW_r\\ &= x + \int_0^t b_a^{(\varepsilon)}((\tilde{X}_r^{a,x+e}-e)+e)dr + \int_0^t \sigma_a^{(\varepsilon)}((\tilde{X}_r^{a,x+e}-e)+e)dW_r;\end{aligned}$$

we have,

$$\hat{X}_t^{(a,e),x} \overset{\text{a.s.}}{=} \tilde{X}_t^{a,x+e} - e,$$

for any $t<T-h$ because they satisfy the same equation over $[0,T-h]$ and that this equation has a unique strong solution:

$$Y_t = x + \int_0^t b_a^{(\varepsilon)}(Y_r+e)dr + \int_0^t \sigma_a^{(\varepsilon)}(Y_r+e)dW_r$$

By the dynamic programming principle that we can extand to that value function for the shaken coefficients,

$$\begin{aligned}u_h(t-s,x-e) &\geq \mathbb{E}\left[u_h(t-s+h,\hat{X}_h^{(a,e),x-e}) + \int_0^h f_a(\hat{X}_r^{(a,e),x-e})dr\right]\\ &\geq \mathbb{E}\left[u_h(t-s+h,\tilde{X}_h^{a,x}-e) + \int_0^h f_a(\tilde{X}_r^{a,x}-e)dr\right].\end{aligned}$$

Now by mollifiying (taking $e$ as the $y$ in 2.2) and inverting the integrals thanks to the dominated convergence theorem ($u_h+\int f_a$ is a bounded function and the integral domain is finished for the mollification):

$$u_h^{(\varepsilon)}(t,x) \geq \mathbb{E}\left[u_h^{(\varepsilon)}(t+h,\tilde{X}_h^{a,x}) + \int_{|e|<\varepsilon}\int_{0\leq s\leq\varepsilon}\int_0^h f_a(\tilde{X}_r^{a,x}-e)dr\rho_\varepsilon(s,e)dsde\right].$$

We can inverse the integral signs in the second term thanks the dominated convergence theorem again. We get (by considering $f$ as a function of time and space but constant in time)

$$u_h^{(\varepsilon)}(t,x) \geq \mathbb{E}\left[\int_0^h f_a^{(\varepsilon)}(\tilde{X}_r)dr + u_h^{(\varepsilon)}(t+h,\tilde{X}_h)\right],$$

Since this is true for any $a\in\mathrm{A}$, $t\in[0,T-h]$, $x\in\mathbb{R}^d$, this yields the super-dynamic programming principle. □

## 2.5 Bounds that are always used?

Finally, we will derive standard continuous dependence results for stochastic differential equations.

**Lemma 2.1.** *There exists $C > 0$ such that for any $t \in [0, T]$:*

$$\mathbb{E}\left|X_t - \tilde{X}_t\right| \leq C\varepsilon.$$

*Proof.* Let $t \in [0, T]$, define $Y_t = \mathbb{E}\left|X_t - \tilde{X}_t\right|^2$. We will show that it satisfies Grönwall lemma. Using triangular inequality, the quadratic inequality $(a+b)^2 \leq 2(a^2+b^2)$, Jensen's inequality and Ito isometry,

$$\begin{aligned}
Y_t &= \mathbb{E}\left|(\int_0^t b_{\alpha_r}(X_r) - b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)dr) + (\int_0^t \sigma_{\alpha_r}(X_r) - \sigma^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)dW_r)\right|^2 \\
&= 2\mathbb{E}\left[(\int_0^t \left|b_{\alpha_r}(X_r) - b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right| dr)^2 + (\int_0^t \left|\sigma_{\alpha_r}(X_r) - \sigma^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right| dW_r)^2\right] \\
&\leq 2\mathbb{E}\left[\int_0^t \left|b_{\alpha_r}(X_r) - b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right|^2 dr + \int_0^t \left|\sigma_{\alpha_r}(X_r) - \sigma^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right|^2 dr\right].
\end{aligned}$$

Now dealing with $b$, or sigma the process is the same:

$$\begin{aligned}
\mathbb{E}\left[\int_0^t \left|b_{\alpha_r}(X_r) - b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right|^2 dr\right] &\leq 2\mathbb{E}\left[\int_0^t \left|b_{\alpha_r}(X_r) - b_{\alpha_r}(\tilde{X}_r)\right|^2 + \left|b_{\alpha_r}(\tilde{X}_r) - b^{(\varepsilon)}_{\alpha_r}(\tilde{X}_r)\right|^2 dr\right] \\
&\leq 2\mathbb{E}\left[C_0^2 \int_0^t \left|X_r - \tilde{X}_r\right|^2 dr + \int_0^t C\varepsilon^2 dr\right] \\
&\leq C\int_0^t |Y_r|\, dr + Ct\varepsilon.
\end{aligned}$$

Thanks to Grönwall Lemma, since this is true for any $t \in [0, T]$, we have:

$$|Y_t| \leq C\varepsilon^2 t\exp(Ct) \leq C\varepsilon T\exp(CT) \leq C\varepsilon^2, \quad \forall t \in [0, T].$$

To conclude, we use Jensen's inequality again. □

**Proposition 2.4.** *There exists a constant $C > 0$ such that, for any $t, s \in [0, T]$ and $x, y \in \mathbb{R}^d$,*

$$|v(t, x) - \tilde{v}(t, x)| \leq C\varepsilon$$

*Proof.* The proof only relies on the lipschitz hypothesis of $f$ and $g$ and Lemma 2.1. □

**Lemma 2.2.** *There exists $C > 0$ such that for any $t \in [0, T]$:*

$$\left|X_t - \hat{X}_t\right| \leq C\varepsilon.$$

*Proof.* The proof is similare to the one of Lemma 2.1 except we also need $\xi < \varepsilon$. □

**Proposition 2.5.** *There exists a constant $C > 0$ such that, for any $t, s \in [0, T]$ and $x, y \in \mathbb{R}^d$,*

$$|v_h(t, x) - u_h(t, x)| \leq C\varepsilon.$$

*Proof.* Similarly, the proof follows from Lemma 2.2 and the regularity of $f$ and $g$. □

# 3 Optimal bound in the smooth setting

As an introduction to the tools and procedure of the proofs we will deal with the case of smooth value functions first.

## 3.1 Convergence rate

We will derive an optimal bound in case of sufficiently smooth data and value functions.

**Theorem 3.1.** *If $v_h$ is smooth enough, for any $t \in [0,T]$, $x \in \mathbb{R}^d$ and $h > 0$, we have:*

$$0 \leq v(t,x) - v_h(t,x) \leq Ch,$$

*where $C$ only depends on the constants from assumptions (H2) and (H3).*

First of all, we have $0 \leq v(t,x) - v_h(t,x)$ for any $t \in [0,T]$, $x \in \mathbb{R}^d$, because $\mathcal{A}_h \subset \mathcal{A}$.

## 3.2 Proof of the bound

Now for the proof of the actual optimal bound. The regularities needed will be highlighted and summarized at the end.

*Proof.* **1) Upper bound on $(\partial_t + L_a)v_h + f_a$.**
Let $s \in [0, T-h]$, $x \in \mathbb{R}^d$ and $a \in$ A. Define $L_a = b_a^T D_x + \frac{1}{2} tr\left[\sigma_a \sigma_a^T D_x D_x^T\right]$ the classic generator. For the value and cost functions, we will need to add $\partial_t$ to this generator when applying Ito's formula because they are time dependant.
By two applications of Dynkin's formula,

$$\begin{aligned}
\mathbb{E}\left[v_h(s+h, X_h)\right] &= v_h(s, X_0) + \mathbb{E}\left[\int_0^h (\partial_t + L_a)v_h(s+t, X_t)dt\right] \\
&= v_h(s,x) + \int_0^h (\partial_t + L_a)v_h(s, X_0) + \mathbb{E}\left[\int_0^t (\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)dr\right] dt \\
&= v_h(s,x) + h(\partial_t + L_a)v_h(s,x) + \mathbb{E}\left[\int_0^h \int_0^t (\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)drdt\right].
\end{aligned}$$

Here applying Dynkin's lemma twice requires $b$ and $\sigma$ to be $C^2$ and $v_h$ to be $C^4$ because we need $(\partial_t + L_a)v_h$ to be $C^2$ as well. In the deterministic setting, it would be $C^2$ for $v_h$ and $C^1$ for $b$. We can insert it in the dynamic programming principle from Proposition 2.2 (using only the first inequality) and suppress $v_h(s,x)$ from both sides,

$$0 \geq \mathbb{E}\left[\int_0^h f_a(X_t)dt\right] + h(\partial_t + L_a)v_h(s,x) + \mathbb{E}\left[\int_0^h \int_0^t (\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)drdt\right].$$

Developing $\mathbb{E}\left[\int_0^h f_a(X_t)\right]$ with Dynkin's formula (which requires $f \in C^2(\mathrm{R}^d)$ or $C^1$ for the deterministic setting) and moving all that is not fixed on the other side gives

$$h((\partial_t + L_a)v_h(s,x) + f_a(x)) \leq -\mathbb{E}\left[\int_0^h \int_0^t L_a f_a(X_r)drdt + \int_0^h \int_0^t (\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)drdt\right].$$

Since this minus sign appears, with only constant lower bounds on the generator of $f$, $v_h$ and $(\partial_t + L_a)v_h$, we can have:

$$(\partial_t + L_a)v_h(s,x) + f_a(x) \leq Ch, \text{ for any } s \in [0, T-h],\ x \in \mathbb{R}^d,\ a \in \mathrm{A}. \tag{3.1}$$

**2) Upper bound on $v - v_h$ for $s \in [0, T-h)$.**
Let $\alpha \in \mathcal{A}$, $s \in [0, T-h]$ and $x \in \mathbb{R}^d$. We'll calculate a bound on the cost function so first on $\mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t)dt\right]$ and then on $\mathbb{E}\left[g(X_{t-s})\right]$. For $f$, we use Dynkin's formula and the previously

calculated bound,

$$\begin{aligned}
\mathbb{E}\left[v_h(T-h, X_{T-h-s})\right] &= v_h(s, X_0) + \mathbb{E}\left[\int_0^{T-h-s} (\partial_t + L_{\alpha_t}) v_h(s+t, X_t) dt\right] \\
&\leq v_h(s,x) - \mathbb{E}\left[\int_0^{T-h-s} f_{\alpha_t}(X_t) dt\right] + (T-h-s)Ch \\
&\leq v_h(s,x) - \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt - \int_{T-h-s}^{T-s} f_{\alpha_t}(X_t) dt\right] + TCh \\
&\leq v_h(s,x) - \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt\right] + \int_{T-h-s}^{T-s} C_0 dt + Ch \\
&\leq v_h(s,x) - \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt\right] + C_0 h + Ch.
\end{aligned}$$

For $g$, we use the definition of $v_h$ at terminal time $T$ for the shifted starting point $X_{T-s}$ and Dynkin's formula on $v_h$ alongside the upper bound on it's generator:

$$\begin{aligned}
\sup_{a \in \mathrm{A}} \mathbb{E}\left[g(X_{T-s}^{a,x})\right] &= \mathbb{E}\left[v_h(T, X_{T-s})\right] \\
&= \mathbb{E}\left[v_h(T-h, X_{T-h-s}) + v_h(T, X_{T-s}) - v_h(T-h, X_{T-h-s})\right] \\
&\leq \mathbb{E}\left[v_h(T-h, X_{T-h-s})\right] + \mathbb{E}\left[\int_{T-h}^{T} (\partial_t + L_{\alpha_t}) v_h(t, X_{t-s}) dt\right] \\
&\leq \mathbb{E}\left[v_h(T-h, X_{T-h-s})\right] + Ch.
\end{aligned}$$

Hence getting the following bounds requires an upper bound on the generator of $v_h$:

$$\begin{cases} \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt\right] \leq v_h(s,x) - \mathbb{E}\left[v_h(T-h, X_{T-h-s})\right] + Ch \\ \mathbb{E}\left[g(X_{T-s})\right] \leq \mathbb{E}\left[v_h(T-h, X_{T-h-s})\right] + Ch. \end{cases} \tag{3.2}$$

Now for the cost function, summing both equations in (3.2):

$$J^\alpha(s,x) = \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt + g(X_{T-s})\right] \leq v_h(s,x) + Ch.$$

Thus taking the supremum over any $\alpha \in \mathcal{A}$:

$$v(s,x) - v_h(s,x) \leq Ch, \text{ for any } s \in [0, T-h], x \in \mathbb{R}^d. \tag{3.3}$$

**3) Upper bound on $v - v_h$ for $s \in [T-h, T]$.**
Let $\alpha \in \mathcal{A}$, $s \in [T-h, T]$ and $x \in \mathbb{R}^d$. Using Ito's lemma on $g$ and it's derivatives boundness alongside $f$ bound,

$$\begin{aligned}
J^\alpha(s,x) - g(x) &= \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t) dt + g(X_{T-s})\right] - g(x) \\
&\leq (T-s)C + \mathbb{E}\left[g(X_{T-s})\right] - g(x) \\
&\leq Ch + \mathbb{E}\left[\int_0^{T-s} L_{\alpha_t} g(X_t) dt\right] \\
&\leq Ch + (T-s) \sup_{a \in \mathrm{A}} \left(\|L_a g^+\|_\infty\right) \\
&\leq Ch.
\end{aligned}$$

This requires $g$ to be at least $C^2$ (and $C^1$ in deterministic) with an upper bound on it's generator. Thus taking the supremum over any $\alpha \in \mathcal{A}_h$:

$$|v_h(s,x) - g(x)| \leq Ch.$$

And taking the supremum over any $\alpha \in \mathcal{A}$:

$$|v(s,x) - g(x)| \leq Ch.$$

Hence:

$$|v(s,x) - v_h(s,x)| \leq Ch. \tag{3.4}$$

**4) Conclusion:**
Letting $s \in [0,T]$ and $x \in \mathbb{R}^d$ Using the bound on $|v(t,x) - v_h(t,x)|$ in part 2) (3.3) or 3) (3.4):

$$|v(s,x) - v_h(s,x)| \leq Ch. \tag{3.5}$$

□

In conclusion, 'sufficiently smooth' should at least mean the regularities in Table 3.2. In reality some of these regularities can be relaxed as it was shown in [3] in a deterministic setting, only semi convexity was necessary.

| | Stochastic | Deterministic |
|---|---|---|
| $b_a$ | $C^2(\mathbb{R}^d)$ with lower bound on it's generator | $C^1(\mathbb{R}^d)$ with lower bound on it's derivative |
| $\sigma_a$ | $C^2(\mathbb{R}^d)$ with lower bound on it's generator | / |
| $f_a$ | $C^2(\mathbb{R}^d)$ with lower bound on it's generator | $C^1(\mathbb{R}^d)$ with lower bound on it's derivative |
| $g$ | $C^2(\mathbb{R}^d)$ with upper bound on it's generator | $C^1(\mathbb{R}^d)$ with lower bound on it's derivative |
| $v_h$ | $C^4(\mathbb{R}^d)$ with lower bound on $(\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)$ and upper bound on $(\partial_t + L_a)v_h(s+r, X_r)$ | $C^2(\mathbb{R}^d)$ with lower bound on $(\partial_t + L_a)(\partial_t + L_a)v_h(s+r, X_r)$ and upper bound on $(\partial_t + L_a)v_h(s+r, X_r)$ |

We have given an optimal bound in case of sufficiently smooth data and value functions. However in the general case these won't have such regularities. All the requirement of differentiability and boundness won't generally be fulfilled. This is why we will need regularisation in the form of mollification and shaking of coefficients.

# 4 Deterministic general case

In this section, we wanted to highlight the inherent worsening of the bound when in a stochastic setting due to Ito's lemma and the apparition of the second derivative in the 'stochastic' fundamental theorem of calculus, *eg* Dynkin's formula. To do so we will derive the optimal bound in the deterministic setting. Hence, for this section $\sigma \equiv 0$ thus we can drop the expectations and we will use the degree p=1 for the mollification.

## 4.1 Convergence rate

We will prove the following result throughout this section:

**Theorem 4.1.** *For any $t \in [0,T]$, $x \in \mathbb{R}^d$ and $h > 0$, we have:*

$$0 \leq v(t,x) - v_h(t,x) \leq Ch^{\frac{1}{2}},$$

*where $C$ only depends on the constants from assumptions (H2), (H3).*

As for the smooth case, we have $0 \leq v(t,x) - v_h(t,x)$ for any $t \in [0,T]$, $x \in \mathbb{R}^d$, because $\mathcal{A}_h \subset \mathcal{A}$.

## 4.2 Regularities of the derived problems

We will show that the value function that we presented in 2.4 also carry the space and time regularities from the coefficients and the data and state the implication for the mollified value function. One can easily see that the bounds calculated in 2.5 still hold.

**Lemma 4.1.** *There exists $C > 0$ such that for any $\alpha \in \mathcal{A}_h$, $\xi \in \mathcal{E}_h$, $t, s \in [0,T]$ and $x, y \in \mathbb{R}^d$:*

$$\left|\hat{X}_t^x - \hat{X}_s^y\right| \leq C(|x-y| + |t-s|)$$

*Proof.* Let $\alpha \in \mathcal{A}$, $t, s \in [0,T]$ and $x, y \in \mathbb{R}^d$. We will split the problem into two parts, one for time and one for space.

$$\left|\hat{X}_t^x - \hat{X}_s^y\right| = \left|\hat{X}_t^x - \hat{X}_t^y + \hat{X}_t^y - \hat{X}_s^y\right| \leq \left|\hat{X}_t^x - \hat{X}_t^y\right| + \left|\hat{X}_t^y - \hat{X}_s^y\right|.$$

For the space term we use (H2) and find a Grönwall inequality.

$$\begin{aligned}
\left|\hat{X}_t^x - \hat{X}_t^y\right| &= \left|x - y + \int_0^t b_{\alpha_r}(\hat{X}_r^x + \xi_r) - b_{\alpha_r}(\hat{X}_r^y + \xi_r)dr\right| \\
&\leq |x-y| + \int_0^t \left|b_{\alpha_r}(\hat{X}_r^x + \xi_r) - b_{\alpha_r}(\hat{X}_r^y + \xi_r)\right| dr \\
&\leq |x-y| + C_0 \int_0^t \left|\hat{X}_r^x - \hat{X}_r^y\right| dr
\end{aligned}$$

Hence with Grönwall Lemma, because it is true for any $t \leq T$,

$$\left|\hat{X}_t^x - \hat{X}_t^y\right| \leq |x-y| \exp(C_0 t) \leq C|x-y|.$$

For the time term,

$$\left|\hat{X}_t^y - \hat{X}_s^y\right| = \left|\int_s^t b_{\alpha_r}(\hat{X}_r^y)dr\right| \leq C_0|t-s|.$$

Summing the two, we get the wanted bound.

□

**Proposition 4.1.** *There exists a constant $C > 0$ such that, for any $t, s \in [0,T]$ and $x, y \in \mathbb{R}^d$,*

$$|u_h(t,x) - u_h(s,y)| \leq C(|x-y| + |t-s|).$$

*Proof.* This property is a direct consequence of the previous bound and of the lipschitz property of $f$ and $g$ (H3). □

Now, since we know that $u_h \in C_b^{1,1}([0,T] : \times \mathbb{R}^d)$, we have by phi mollified properties 2.1

**Proposition 4.2.** *There exists a constant $C > 0$ such that:*

$$\|\partial_t^m D_x^k u_h^{(\varepsilon)}\|_\infty \leq C\varepsilon^{1-m-k} \quad for \quad k+m \geq 1.$$

## 4.3 Proof of the bound

Now we will address the proof of the bound in Theorem 4.1. The proof will resemble the one for Theorem 3.1.

**1) upper bound on** $(\partial_t + L_a)u_h^{(\varepsilon)} + f_a^{(\varepsilon)}$**.**

Let $s \in [0, T-h]$, $x \in \mathbb{R}^d$ and $a \in \mathrm{A}$. Let $\tilde{L}_a = b_a^{(\varepsilon)T} D_x$.

We define $w : [s, T] \longrightarrow \mathbb{R}$ only for $u_h^{(\varepsilon)}$ now: $w(t) := u_h^{(\varepsilon)}(s + t, \tilde{X}_t^{a,x})$, and again with the fundamental theorem of calculus (instead of Dynkin's formula) applied twice on $w$ and $w'$, from 0 to $h$:

$$u_h^{(\varepsilon)}(s+h, \tilde{X}_h) = u_h^{(\varepsilon)}(s,x) + h(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + \int_0^h \int_0^t (\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s+r, \tilde{X}_r)drdt.$$

Inserting this into the SDPP for $u_h^{(\varepsilon)}$ in Proposition 4.2, we get:

$$u_h^{(\varepsilon)}(s,x) \geq \int_0^h f_a^{(\varepsilon)}(\tilde{X}_t)dt + u_h^{(\varepsilon)}(s,x) + h(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + \int_0^h \int_0^t (\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s+r, \tilde{X}_r)drdt.$$

Developing $f_a^{(\varepsilon)}(\tilde{X}_t) = f_a^{(\varepsilon)}(x) + \int_0^t \tilde{L}_a f_a^{(\varepsilon)}(\tilde{X}_r)dr$, and organising in the same way,

$$\begin{aligned} h((\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + f_a^{(\varepsilon)}(x)) \leq &- \int_0^h \int_0^t \tilde{L}_a f_a^{(\varepsilon)}(\tilde{X}_r)drdt \\ &- \int_0^h \int_0^t (\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s+r, \tilde{X}_r)drdt, \end{aligned}$$

Since $(\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a) = \partial_t\partial_t + \partial_t b_a^{(\varepsilon)T} D_x + b_a^{(\varepsilon)T} D_x \partial_t + b_a^{(\varepsilon)T} D_x b_a^{(\varepsilon)T} D_x = \sum_i \partial_t^{m_i} (b_a^{(\varepsilon)T} D_x)^{k_i}$ with $\max_i(m_i + k_i) \leq 2$, with the mollified properties from Proposition 2.1, we get:

$$(4.1) \qquad (\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + f_a^{(\varepsilon)}(x) \leq Ch\varepsilon^{-1}, \text{ for any } s \in [0, T-h],\ x \in \mathbb{R}^d,\ a \in \mathrm{A}.$$

This is the major improvement in the deterministic case, since the 'generator' is only one derivative, we can two degrees of order for that bound.

**2) Upper bound on** $\tilde{v} - v_h$ **for** $s \in [0, T-h)$**.**

Let $\alpha \in \mathcal{A}$, $s \in [0, T-h]$ and $x \in \mathbb{R}^d$. We'll calculate a bound on $\int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt$ and $g(\tilde{X}_{t-s})$:
With the fundamental theorem of calculus applied again on $w(t) := u_h^{(\varepsilon)}(s+t, \tilde{X}_t^{a,x})$, but this time between 0 and $T-h-s$.
With the above (4.1), mollification property 2.1 for $u_h^{(\varepsilon)}$, the bound of Proposition 4.1 on $\|v_h - u_h\|_\infty$ and the inherited boundedness of $f^{(\varepsilon)}$ from (H3) and following what we did before to get the bound in the smooth setting.

$$u_h^{(\varepsilon)}(T-h, \tilde{X}_{T-h-s}) \leq v_h(s,x) + 2C\varepsilon - \int_0^{T-s} f_{\alpha_t}^{(\varepsilon)}(\tilde{X}_t)dt + C_0 h + Ch\varepsilon^{-1}.$$

With the Proposition 2.1 but for $f_{\alpha_t}^{(\varepsilon)}$: $\int_0^{T-s} f_{\alpha_t}^{(\varepsilon)}(\tilde{X}_t)dt \geq \int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt - CT\varepsilon$. This gives the following bound instead of (3.2):

$$(4.2) \qquad \int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt \leq v_h(s,x) - u_h^{(\varepsilon)}(T-h, \tilde{X}_{T-h-s}) + C(\varepsilon + h\varepsilon^{-1} + h).$$

Using the Definition 2.6 at terminal time $T$ for the shifted starting point $x = X_{T-s}$, using a similar procedure as for (3.2) but using the space regularity in Proposition 4.1, part 2), we get:

$$\begin{aligned} \sup_{a \in \mathrm{A}, e \in B_\varepsilon(0)} (g(\tilde{X}_{T-s}^{a,x})) &= \sup_{a \in \mathrm{A}, e \in B_\varepsilon(0)} (g(\hat{X}_0^{\tilde{X}_{T-s}})) \\ &\leq u_h(T-h, X_{T-h-s}) + Ch. \end{aligned}$$

This gives the following bound

$$g(\tilde{X}_{T-s}) \leq u_h(T-h, \tilde{X}_{T-h-s}) + Ch. \tag{4.3}$$

Now for the cost function, with the bound 4.2 and 4.3 on $\int_0^{T-s} f_{\alpha_t}(X_t)dt$ and $g(X_{T-s})$:

$$\begin{aligned}
\tilde{J}^\alpha(s,x) &= \int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt + g(\tilde{X}_{T-s}) \\
&\leq v_h(s,x) - u_h^{(\varepsilon)}(T-h, \tilde{X}_{T-h-s}) + C(\varepsilon + h\varepsilon^{-1} + h) + u_h(T-h, \tilde{X}_{T-h-s}) + Ch \\
&\leq v_h(s,x) + C(\varepsilon + h\varepsilon^{-1} + h).
\end{aligned}$$

Thus taking the supremum over any $\alpha \in \mathcal{A}$:

$$\tilde{v}(s,x) - v_h(s,x) \leq C(\varepsilon + h\varepsilon^{-1} + h). \tag{4.4}$$

**3) Upper bound on $\tilde{v} - v_h$ for $s \in [T-h, T]$.**
Here the procedure is similar to the one in the optimal case (3.4) except now the end cost $g$ is not smooth anymore but it is still lipschitz:

$$|g(X_{T-s}) - g(x)| \leq C|X_{T-s} - X_0| \leq C(T-s) \leq Ch.$$

The result still holds for $\tilde{X}$. Hence we can bound $|J^\alpha(s,x) - g(x)|$ and $|\tilde{J}^\alpha(s,x) - g(x)|$ and we get:

$$|\tilde{v}(s,x) - v_h(s,x)| \leq Ch < C(\varepsilon + h + h\varepsilon^{-1}). \tag{4.5}$$

**4) Conclusion:**
Letting $s \in [0,T]$ and $x \in \mathbb{R}^d$ Using the bound 4.4 in part 2) or 4.5 in 3):

$$v(s,x) - v_h(s,x) \leq \tilde{v}(s,x) - v_h(s,x) + C\varepsilon \leq C(\varepsilon + h + h\varepsilon^{-1}).$$

Optimizing that bound, we find for $\varepsilon = h^{\frac{1}{2}}$:

$$v(s,x) - v_h(s,x) \leq Ch^{\frac{1}{2}}. \tag{4.6}$$

# 5 Stochastic case

In this section we consider the problem for a stochastic setting. Thus we take $\sigma \not\equiv 0$ which makes $X_\cdot$ a diffusion solution to the equation. The degree of mollification for $t$ is taken $p = 2$. The aim is to improve the result of [6] by having some regularities on the second derivative of the value function.

To improve this result we need to improve the bound in the first part of the proof 3.1 and to do so one can try to get a better upper bound on $-\partial_t u_h$ and on $-L_a u_h$ because they are the leading terms in $\varepsilon$. Keeping that in mind, the differential equation that $v$ is solving can give us an idea of the problem we will face: since because the problem is backward in time, we can equate $v_t$ and $-L_a v_h$ in term of differentiability and guess that bounding $-v_t$ from above will mean bounding $L_a v_h$ from above but because of the sup in the definition of the value functions, this will prove difficult.

Nonetheless, we will see that considering (H4) and (H5) we can reach a slight improvement in the bound.

## 5.1 Convergence rate

We will prove the following result throughout the rest of this section:

**Theorem 5.1.** *For any $t \in [0, T]$, $x \in \mathbb{R}^d$ and $h > 0$, we have:*

$$0 \le v(t, x) - v_h(t, x) \le Ch^{\frac{1}{3}},$$

*where $C$ only depends on the constants from the different assumptions.*

As for both previous cases, we have $0 \le v(t, x) - v_h(t, x)$ for any $t \in [0, T]$, $x \in \mathbb{R}^d$, because $\mathcal{A}_h \subset \mathcal{A}$.

## 5.2 Improved regularities

With these new hypotheses, we will bound the derivatives of $u_h^{(\varepsilon)}$ in a better way which will give us an improvement. We will still

**Lemma 5.1.** *There exists $C > 0$ such that for any $x, y \in \mathbb{R}^d$ and $t, s \in [0, T]$:*

$$\mathbb{E}\left[|\tilde{X}_t^x - \tilde{X}_s^y|\right] \le C(|x - y| + |t - s|^{\frac{1}{2}}).$$

*Proof.* One can look at the proof for $\hat{X}$ in the deterministic setting, Proposition 4.1. One need only to deal with squared differences and the lesser regularity will come from the time part and Ito's isometry. □

**Lemma 5.2.** *There exists $C > 0$ such that for any $\alpha \in \mathcal{A}_h$, $\xi \in \mathcal{E}_h$, $t \in [0, T]$ and $x, y \in \mathbb{R}^d$:*

$$\mathbb{E}\left[|\hat{X}_t^x + \hat{X}_t^y - 2\hat{X}_t^{\frac{x+y}{2}}|\right] \le C|x - y|^2.$$

*Proof.* If we pick any other point $z \in \mathbb{R}^d$ and $a \in \mathcal{A}$ and take $\varphi \in \{b_a^{(\varepsilon)}, \sigma_a^{(\varepsilon)}\}$, since $b_a^{(\varepsilon)}$ and $\sigma_a^{(\varepsilon)}$ inherit (H4)we have,

$$\begin{aligned}
|\varphi(x) + \varphi(y) - 2\varphi(z)| &\le \left|\varphi(x) + \varphi(y) - 2\varphi(\frac{x+y}{2})\right| + \left|2\varphi(\frac{x+y}{2}) - 2\varphi(z)\right| \\
&\le \left|\varphi(x) + \varphi(y) - 2\varphi(\frac{x+y}{2})\right| + 2\left|\varphi(\frac{x+y}{2}) - \varphi(z)\right| \\
&\le C|y - x|^2 + C|x + y - 2z|.
\end{aligned}$$

Hence taking the square of that,

$$|\varphi(x) + \varphi(y) - 2\varphi(z)|^2 \le (C|y - x|^2 + C|x + y - 2z|)^2 \le 2C|y - x|^4 + 2C|x + y - 2z|^2.$$

With this inequality, one can derive from Gronwall lemma with Ito isometry and Jensen's lemma that this inquality holds:

$$\mathbb{E}\left[|\hat{X}_t^x + \hat{X}_t^y - 2\hat{X}_t^{\frac{x+y}{2}}|^2\right] \le C|x - y|^4$$

And with Jensen's inequality again we get the bound. □

Now we can show that the shaken coefficient value function will be semi-convex in space but that it will have a one sided lipschitz bound in time.

**Proposition 5.1** ($u_h$ additional properties)**.** *There exists a constant $C > 0$ such that, for any $t, s \in [0,T]$ and $x, y \in \mathbb{R}^d$,*

**1.** $|u_h(t,x) - u_h(s,y)| \leq C(|x-y| + |t-s|^{\frac{1}{2}})$;

**2.** $u_h(t,x+y) + u_h(s,x-y) - 2u_h(t,x) \geq -C|x-y|^2$;

**3.** $u_h(t,x) - u_h(s,x) \geq -C|t-s|$;

*Proof.* **1.** This is the usual result regarding regularity, see 4.1 but transpose it to the stochastic setting or see [7].

**2.** The fourth property is a consequence of f and g being semi convex functions as well as Lemma 5.2 . First let's show that there exists $C > 0$ such that the cost function is semi-convex in space. Let $\alpha \in \mathcal{A}_h$, $\xi \in \mathcal{E}_h$ we have:

$$\hat{J}^{(\alpha,\xi)}(t,x+y) + \hat{J}^{(\alpha,\xi)}(t,x-y) - 2\hat{J}^{(\alpha,\xi)}(t,x) = \mathbb{E}\left[\int_0^{T-t} f_{\alpha_r}(\hat{X}_r^{x+y}) + f_{\alpha_r}(\hat{X}_r^{x-y}) - 2f_{\alpha_r}(\hat{X}_r^{x})dr + g(\hat{X}_{T-t}^{x+y}) + g(\hat{X}_{T-t}^{x-y}) - 2g(\hat{X}_{T-t}^{x})\right].$$

Now dealing with the second part, with the help of Lemma 5.2,

$$\begin{aligned}\mathbb{E}\left[g(X_{T-t}^{x+y}) + g(X_{T-t}^{x-y}) - 2g(X_{T-t}^{x})\right] &= \mathbb{E}\Bigg[g(X_{T-t}^{x+y}) + g(X_{T-t}^{x-y}) - 2g(\frac{X_{T-t}^{x+y} + X_{T-t}^{x-y}}{2}) \\ &\quad +2g(\frac{X_{T-t}^{x+y} + X_{T-t}^{x-y}}{2}) - 2g(X_{T-t}^{x})\Bigg] \\ &\geq \mathbb{E}\left[-C\left|X_{T-t}^{x+y} - X_{T-t}^{x-y}\right|^2 - 2C|\frac{X_{T-t}^{x+y} + X_{T-t}^{x-y}}{2} - X_{T-t}^{x}|\right] \\ &\geq -C|x-y|^2 - C|x-y|^2\end{aligned}$$

One can apply a similar proof to the first part with $f$ and get the semi convexity bound on the cost function. We see here that to get the cost function to be semi-convex, we need a bound the diffusion to be semi-convex but also semi-concave, hence (H4).

To get the same on the value function, one can consider a $\frac{1}{2}\delta$-optimal control $(\alpha, \xi)$ for point $(x,t)$ as in the proof on the Dynamic programming principle. With that control we have:

$$\begin{cases} u_h(t,x+y) \geq J^{(\alpha,\xi)}(t,x+y) \\ u_h(t,x-y) \geq J^{(\alpha,\xi)}(t,x-y) \\ -2u_h(t,x) + \delta \geq -2J^{(\alpha,\xi)}(t,x) \end{cases}$$

and this gives,

$$u_h(t,x+y) + u_h(s,x-y) - 2u_h(t,x) + \delta \geq J^{(\alpha,\xi)}(t,x+y) + J^{(\alpha,\xi)}(s,x-y) - 2J^{(\alpha,\xi)}(t,x) \geq -C|x-y|^2$$

**3.** Finally the fifth property is a consequence of $g$ being semi concave. We have as a property of the sup:

$$u_h(t,x) - u_h(s,x) \geq -\sup_{\alpha\in\mathcal{A}_h, \xi\in\mathcal{E}_h} \mathbb{E}\left[\hat{J}^{(\alpha,\xi)}(s,x) - \hat{J}^{(\alpha,\xi)}(t,x)\right].$$

But we have problem that, $g$ is only almost surely $C_b^2$ so we cannot apply Dynkin's formula directly we need to go to it's mollified version, bound it's infinitesimal generator and send the $\delta$ to zero. Let $\delta > 0$, $t > s \in [0,T]$:

$$\begin{aligned}\mathbb{E}\left[g(\hat{X}_t) - g(\hat{X}_s)\right] &\leq \mathbb{E}\left[g^{(\delta)}(\hat{X}_t) - g^{(\delta)}(\hat{X}_s) + g(\hat{X}_t) - g(\hat{X}_s) - g^{(\delta)}(\hat{X}_t) - g^{(\delta)}(\hat{X}_s)\right] \\ &\leq \mathbb{E}\left[\int_s^t \tilde{L}_{\alpha_t} g^{(\delta)}(\hat{X}_t)dt\right] + 2C\delta \\ &\leq C(t-s) + 2C\delta.\end{aligned} \tag{5.1}$$

We could also say that $L_{\alpha_t} g(X_t)$ is defined and bounded almost everywhere and as such can be defined when integrated.

$$\mathbb{E}\left[\hat{J}^{(\alpha,\xi)}(s,x) - \hat{J}^{(\alpha,\xi)}(t,x)\right] = \mathbb{E}\left[\int_{T-t}^{T-s} f_{\alpha_r}(\hat{X}_r)dr + g(\hat{X}_{T-s}) - g(\hat{X}_{T-t})\right]$$
$$\leq C|t-s| + C|t-s|.$$

Applying the sup and the minus, we get the property. □

Now, this next property improves the property on the mollification of $u_h$ in order to get a better bound on $L_a u_h + f_a$ in the first part of the proof of the theorem.

**Proposition 5.2** ($u_h^{(\varepsilon)}$ additional properties)**.** *We have the usual mollification bounds:*

$$\|\partial_t^m D_x^k u_h^{(\varepsilon)}\|_\infty \leq C\varepsilon^{1-2m-k}, \ \forall m+k \geq 1$$

*But thanks to the new hypothesis, we can have better bounds on the derivative of this mollification.*

$$\inf_{t\in[0,T],x\in\mathbb{R}^d} \partial_t^m D_x^k u_h^{(\varepsilon)}(t,x) \geq -C\varepsilon^{2-2m-k} \ \textit{if } m \geq 1 \ \textit{or } k \geq 2$$

*Proof.* The first bound is direct since $u_h \in C_b^{\frac{1}{p},1}([0,T]\times\mathbb{R}^d)$.

For the second, morally we are applying one of the derivatives on $u_h$ inside the mollification. Effectively, we use the one sided lipschitz property that we derived for $u_h$. One can refer to this proof for the proof of the corresponding part of Property 2.1. If $m \geq 1$ we bound one time derivative and if $k \geq 2$ we bound two space derivatives.
Let $t \in [0,T]$ and $x \in \mathbb{R}^d$, and let $m \geq 1$,

$$\partial_t^m D_x^k u_h^{(\varepsilon)}(t,x) = \lim_{s\to 0} \frac{\partial_t^{m-1} D_x^k u_h^{(\varepsilon)}(t+s,x) - \partial_t^{m-1} D_x^k u_h^{(\varepsilon)}(t,x)}{s} = \lim_{s\to 0} \Delta^{m-1,k} u_h^{(\varepsilon)}(s).$$

Let $s > 0$, we have

$$\begin{aligned}
\Delta^{m-1,k} u_h^{(\varepsilon)}(s) &= \frac{(u_h * \partial_t^{m-1} D_x^k \rho_\varepsilon)(t+s,x) - (u_h * \partial_t^{m-1} D_x^k \rho_\varepsilon)(t,x)}{s} \\
&= \frac{1}{s}\int_{|y|<\varepsilon}\int_{0\leq s\leq\varepsilon^2} (u_h(t+s-s,x-y) - \\
&\qquad\qquad u_h(t-s,x-y))\partial_t^{m-1} D_x^k \rho_\varepsilon(s,y)dsdy \\
&\geq \frac{1}{s}\int_{|y|<\varepsilon}\int_{0\leq s\leq\varepsilon^2} (-Cs)\partial_t^{m-1} D_x^k \rho_\varepsilon(s,y)dsdy \\
&\geq -C\int_{|y|<\varepsilon}\int_{0\leq s\leq\varepsilon^2} \frac{1}{\varepsilon^{d+2}}\frac{1}{\varepsilon^{(m-1)2}}\frac{1}{\varepsilon^k}\partial_t^{m-1} D_x^k \rho(\frac{s}{\varepsilon^2},\frac{y}{\varepsilon})dsdy \\
&\geq -C\varepsilon^{2-2m-k}\int_{|y|<\varepsilon}\int_{0\leq s\leq\varepsilon^2} \frac{1}{\varepsilon^{d+2}}\partial_t^{m-1} D_x^k \rho(\frac{s}{\varepsilon^2},\frac{y}{\varepsilon})dsdy \\
&\geq -C\varepsilon^{2-2m-k}\int_{|z|<1}\int_{0\leq l\leq 1} \frac{1}{\varepsilon^{d+2}}\partial_t^{m-1} D_x^k \rho(l,z)\varepsilon^2 dl\varepsilon^d dz \\
&\geq -C\varepsilon^{2-2m-k}\int_{|z|<1}\int_{0\leq l\leq 1} \partial_t^{m-1} D_x^k \rho(l,z)dldz \\
&\geq -C\varepsilon^{2-2m-k}.
\end{aligned}$$

Now if $k \geq 2$, we have a similar method but for the second space derivative. Taking $z \in \mathbb{R}^d$ a vector of norm one:

$$\partial_t^m D_x^k u_h^{(\varepsilon)}(t,x) = \lim_{s\to 0} \frac{\partial_t^m D_x^{k-2} u_h^{(\varepsilon)}(t,x+sz) + \partial_t^m D_x^{k-2} u_h^{(\varepsilon)}(t,x-sz) - 2\partial_t^m D_x^{k-2} u_h^{(\varepsilon)}(t,x)}{s^2}.$$

With that one can do the same as we did for the time part only using the semiconvexity of $u_h$ instead of the time lipschitz property. □

### 5.3 Proof of the bound

We will address the proof of the bound in Theorem 5.1. The proof will resemble a combination of Theorem 3.1 and 4.1. We will go through it highlighting the important change.

**1) upper bound on** $(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)} + f_a^{(\varepsilon)}$**.**

Let $s \in [0, T-h]$, $x \in \mathbb{R}^d$ and $a \in A$.

First of all, let $\tilde{L}_a = b_a^{(\varepsilon)T} D_x + \frac{1}{2} tr\left[\sigma_a^{(\varepsilon)T} D_x^2 \sigma_a^{(\varepsilon)}\right]$ the generator but when the diffusion has mollified coefficients.

We use again Dynkin's formula twice but on $u_h^{(\varepsilon)}$ and get,

$$\mathbb{E}\left[u_h^{(\varepsilon)}(s+h, \tilde{X}_h)\right] = u_h^{(\varepsilon)}(s,x) + h(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + \mathbb{E}\left[\int_0^h \int_0^t (\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s+r, \tilde{X}_r)drdt\right],$$

with it's SDPP, developing $\mathbb{E}\left[f_a^{(\varepsilon)}(\tilde{X}_t)\right]$ with Dynkin's formula, we get:

$$\begin{aligned} h((\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + f_a^{(\varepsilon)}(x)) \le \mathbb{E}\Bigg[ &-\int_0^h \int_0^t \tilde{L}_a f_a^{(\varepsilon)}(\tilde{X}_r)drdt \\ &- \int_0^h \int_0^t (\partial_t + \tilde{L}_a)(\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s+r, \tilde{X}_r)drdt\Bigg]. \end{aligned}$$

Now, thanks to the new hypothesis and the improved properties of the mollified value function in Property 5.2, we will obtain a better bound than [6] (2.11).

Let's list the different bounds for each terms:

1. $\mathbb{E}\left[-\tilde{L}_a f_a^{(\varepsilon)}(\tilde{X}_r)\right] \le C$ because the first and second derivatives of $f_a^{(\varepsilon)}$ are bounded from below because of the semi-convexity of $f$;
2. $\mathbb{E}\left[-\partial_t \partial_t u_h^{(\varepsilon)}(s+r, \tilde{X}_r)\right] \le C\varepsilon^{-2}$;
3. $\mathbb{E}\left[-\partial_t \tilde{L}_a u_h^{(\varepsilon)}(s+r, \tilde{X}_r)\right] \le C\varepsilon^{-2}$ with this bound being the one from the leading term: $\mathbb{E}\left[-\partial_t[\frac{1}{2}tr(\sigma_a^{(\varepsilon)}(\tilde{X}_r)^T D_x^2 u_h(s+r, \tilde{X}_r)\sigma_a^{(\varepsilon)}(\tilde{X}_r))]\right]$;
4. $\mathbb{E}\left[-\tilde{L}_a \tilde{L}_a u_h^{(\varepsilon)}(s+r, \tilde{X}_r)\right] \le C\varepsilon^{-2}$ with this bound being the one from the leading term: $\mathbb{E}\left[-\frac{1}{2}tr(\sigma_a^{(\varepsilon)}(\tilde{X}_r)^T D_x^2[\frac{1}{2}tr(\sigma_a^{(\varepsilon)}(\tilde{X}_r)^T D_x^2 u_h(s+r, \tilde{X}_r)\sigma_a^{(\varepsilon)}(\tilde{X}_r))]\sigma_a^{(\varepsilon)}(\tilde{X}_r))\right]$;

$$(5.2) \qquad (\partial_t + \tilde{L}_a)u_h^{(\varepsilon)}(s,x) + f_a^{(\varepsilon)}(x) \le Ch\varepsilon^{-2}, \text{ for any } s \in [0, T-h],\ x \in \mathbb{R}^d,\ a \in A.$$

**2) Upper bound on** $v - v_h$ **for** $s \in [0, T-h)$**.**

Let $\alpha \in \mathcal{A}$, $s \in [0, T-h]$ and $x \in \mathbb{R}^d$. This time we'll calculate a bound on $\mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt\right]$ and $\mathbb{E}\left[g(\tilde{X}_{T-s})\right]$:

With Ito's formula, the above (5.2), mollification property 2.1 for $u_h^{(\varepsilon)}$, the bound of Proposition 2.5 on $\|v_h - u_h\|_\infty$ and the inherited boundedness of $f^{(\varepsilon)}$ from (H3).

$$\begin{aligned} \mathbb{E}\left[u_h^{(\varepsilon)}(T-h, \tilde{X}_{T-h-s})\right] &\le u_h^{(\varepsilon)}(s,x) + \int_0^{T-h-s} (\partial_t + \tilde{L}_{\alpha_t})u_h^{(\varepsilon)}(s+t, \tilde{X}_t)dt \\ &\le v_h(s,x) + 2C\varepsilon - \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}^{(\varepsilon)}(\tilde{X}_t)dt\right] + C_0 h + Ch\varepsilon^{-2} \end{aligned}$$

This gives the following bound instead of (3.2):

$$(5.3) \qquad \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt\right] \le v_h(s,x) - \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + C(\varepsilon + h\varepsilon^{-2} + h).$$

Note that we are switching back to the unmollified $f$ and $u_h$ because we already have a $C\varepsilon$ in the bound. But we need it in the first place to use $\tilde{X}$ and so that we deal with $\tilde{L}_a$ and not $L_a$. Using the regularities of $u_h$ in Proposition 5.1, its Definition 2.6 at terminal time $T$ for the shifted starting point $x = X_{T-s}$, and Lemma 5.1:

$$\begin{aligned}
\sup_{a\in \mathrm{A}} \mathbb{E}\left[g(\tilde{X}^{a,x}_{T-s})\right] &= \mathbb{E}\left[u_h(T, \tilde{X}_{T-s})\right] \\
&= \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s}) + u_h(T, \tilde{X}_{T-s}) - u_h(T-h, \tilde{X}_{T-h-s})\right] \\
&\leq \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + \mathbb{E}\left[|\tilde{X}_{T-s} - \tilde{X}_{T-h-s}|\right] + |T-s-T+h+s|^{\frac{1}{2}} \\
&\leq \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + Ch^{\frac{1}{2}}.
\end{aligned}$$

This gives the following bound

$$\mathbb{E}\left[g(\tilde{X}_{T-s})\right] \leq \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + Ch^{\frac{1}{2}}. \tag{5.4}$$

Now for the cost function, with the bound 5.3 and 5.4 on $\mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt\right]$ and $\mathbb{E}\left[g(\tilde{X}_{t-s})\right]$:

$$\begin{aligned}
\tilde{J}^{\alpha}(s,x) &= \mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(\tilde{X}_t)dt + g(\tilde{X}_{T-s})\right] \\
&\leq v_h(s,x) - \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + C(\varepsilon + h\varepsilon^{-2} + h) + \mathbb{E}\left[u_h(T-h, \tilde{X}_{T-h-s})\right] + Ch^{\frac{1}{2}} \\
&\leq v_h(s,x) + C(\varepsilon + h\varepsilon^{-2} + h^{\frac{1}{2}}).
\end{aligned}$$

Thus taking the supremum over any $\alpha \in \mathcal{A}$,

$$\tilde{v}(s,x) - v_h(s,x) \leq C(\varepsilon + h\varepsilon^{-2} + h^{\frac{1}{2}}),$$

and using the bound from Lemma 2.4,

$$v(s,x) - v_h(s,x) \leq C(\varepsilon + h\varepsilon^{-2} + h^{\frac{1}{2}}). \tag{5.5}$$

**3) Upper bound on $v - v_h$ for $s \in [T-h, T]$.**
Once again, we do this bound in the same way we did for (3.4) so g lipschitz property and f bound in (H3). Except this time, because the bound for $X$ in time is only $\frac{1}{2}$-Hölder continuous, we only get a $h^{\frac{1}{2}}$ bound. Let $\alpha \in \mathcal{A}$, $s \in [T-h, T]$ and $x \in \mathbb{R}^d$.

$$\begin{aligned}
|J^{\alpha}(s,x) - g(x)| &= \left|\mathbb{E}\left[\int_0^{T-s} f_{\alpha_t}(X_t)dt + g(X_{T-s})\right] - g(x)\right| \\
&\leq (T-s)C + \mathbb{E}\left[|X_{T-s} - x|\right] \\
&\leq Ch + C|T-s-0|^{\frac{1}{2}} \\
&\leq Ch^{\frac{1}{2}}
\end{aligned}$$

Thus taking the supremum over any $\alpha \in \mathcal{A}_h$:

$$|v_h(s,x) - g(x)| \leq Ch^{\frac{1}{2}}.$$

And taking the supremum over any $\alpha \in \mathcal{A}$:

$$|v(s,x) - g(x)| \leq Ch^{\frac{1}{2}}.$$

Hence:

$$|v(s,x) - v_h(s,x)| \leq Ch^{\frac{1}{2}} < C(h^{\frac{1}{2}} + \varepsilon + h\varepsilon^{-2}). \tag{5.6}$$

Note that we could improve this bound to $Ch$ thanks to $g$'s new regularities but it would not improve the overall bound because $\varepsilon + h\varepsilon^{-2}$ already cannot reach a regularity as precise as $h^{\frac{1}{2}}$.

**4) Conclusion:** Letting $s \in [0, T]$ and $x \in \mathbb{R}^d$ Using the bound 5.5 in part 2) or 4.5 in 3):

$$v(s,x) - v_h(s,x) \leq C(h^{\frac{1}{2}} + \varepsilon + h\varepsilon^{-2}).$$

Optimizing that bound, we find for $\varepsilon = h^{\frac{1}{3}}$:

$$v(s,x) - v_h(s,x) \leq Ch^{\frac{1}{3}}. \tag{5.7}$$

# 6 Numerical tests

For the testing of the bound we will use the scheme constructed in [10] because it is the closest to our base setting. The problem is that as it was explained in Subsection 1.3, this scheme was developed to test the impact of the discretisation of the time and space and it's convergence rate ($\frac{1}{4}$ in time and $\frac{1}{5}$ in space) prevents us from showing the bound. Indeed, in this paper and the others ([2], [1]) schemes are developed to calculate the optimal value function. The goal is to see the impact of the grid size on the rate of convergence to reach the best rate at the lowest computational cost whereas in this thesis, the main goal is to see the improvement in the error rate due only to the size of the segment where the policy is taken constant.

Usually, this time $h$ of constancy of the policies is taken equal to $\Delta t$ the mesh jump in time. This is because when trying to reach the best rate, one would never take policies that are constant on several $\Delta t$ in a row, it would be inefficient in that it complicates the computation of the Dynamic programming principle but does not reduce the number of time we have to use it. Indeed, one would still have to use it at each time steps and need to look at all the time steps over which the policy constant $h$ span over (instead of just one when $h = \Delta t$).

But the fact is that in those studies, the rate theoretically calculated are never reached because the solution value function is always at least piece wise smoother than assumed for the general theory part. To try and show our calculated rate, we will decouple $\Delta t$ and $h$, adapt the scheme in the right manner and show though it's use with different $h$ that indeed, the rate $Ch^{\frac{1}{3}}$ can be reached.

## 6.1 The discrete and decoupled scheme

The scheme is largely described, explained and analysed in [10], we will only explain it briefly here and show how we are modifying it. It revolves around these five steps:

1. Approximating the Ito diffusion by the classical Euler-Maruyama scheme: Let $\alpha \in \mathcal{A}_h$ we can represent it by it's values on each constant interval, $(a_0, ..., a_{N-1}) \in \mathrm{A}^N$ with $N \in \mathbb{N}$ the number on time steps we are taking:

$$X_{t_{n+1}} = X_{t_n} + hb_{a_n}(t_n, X_{t_n}) + \sigma_{a_n}(t_n, X_{t_n})\Delta W_n,$$

   for $n = 0, \dots, N-1$, with $t_0 = 0$, $t_N = T$ and $\Delta W_n = W_{t_{n+1}} - W_{t_n} \sim \sqrt{h}\mathcal{N}(0, I_p)$ the normal increments.

2. Defining the value function in a backward manner thanks to the Dynamic programming principle.

$$v_h(t_n, x) = \sup_{a \in \mathrm{A}} \mathbb{E}\left[hf_a(x) + v_h(t_{n+1}, X_{t_{n+1}})\right].$$

3. Defining the Gaussian quadrature which requires to further approximate the Ito diffusion by a Markov chain type approximation. Define $M \geq 2$ the order of the quadrature, $\xi_i$, $i = 1, \dots, M$ the $M+1$ quadrature points and $\lambda_i$ their respective weights. We get the following exact approximation for any P polynomial of degree lower or equal to 2M-1:

$$\int_{-\infty}^{-\infty} P(y)\frac{e^{-\frac{y^2}{2}}}{\sqrt{2\pi}}dy = \sum_{i=1}^{M} \lambda_i P(\xi_i).$$

   Hence we want to approximate the normal increment by a random variable for which the expectation would lead to the aforementioned sum. Taking $\{\zeta_n\}_{n=0,\dots,N-1}$, $N$ i.i.d. variable such that for $n$,

$$\mathbb{P}(\zeta_n = \xi_i) = \lambda_i, \text{ for } i = 1, \dots, M,$$

   will work. And this gives the following Markov chain approximation:

$$X_{t_{n+1}} = X_{t_n} + hb_{a_n}(t_n, X_{t_n}) + \sqrt{h}\sigma_{a_n}(t_n, X_{t_n})\zeta_n.$$

4. Approximating the expectation in the DPP via Gauss-Hermite quadrature method.

$$\begin{cases} v_h(t_n, x) = \sup_{a \in \mathrm{A}} \left[hf_a(x) + \sum_{i=1}^{M} \lambda_i v(t_{n+1}, x + hb_a(t_n, x) + \sqrt{h}\sigma_a(t_n, x)\xi_i)\right], \\ v(T, x) = g(x), \end{cases}$$

5. Calculating the value for this quadrature at each time via a multilinear interpolation of the values calculated for the previous time. This means that we only compute the values at certain points of the space, the grid $\{ x_m = m\Delta x, \forall m \in \mathbb{Z}^d\}$ (in $\mathbb{R}^d$, $\Delta x$ is a vector and the $m$ are vectors of increment).

$$\begin{cases} v(t_n, x_m) = \sup_{a\in A} \left[ hf_a(x_m) + \sum_{i=0}^{M} \lambda_i \mathcal{I}[v](t_{n+1}, x_m + hb_a(t_n, x_m) + \sqrt{h}\sigma_a(t_n, x_m)\xi_i) \right] \\ v(T, x_m) = g(x_m), \end{cases}$$

for $n = N-1, ..., 0$, $m = -M/2, \ldots, M/2$ with $M$ the number of points and $\mathcal{I}$ is the standard multilinear interpolation operator with respect to the space variable.

We are changing it by taking smaller Euler-Maruyama steps than h the size of the constancy of the policies: $h = \eta\Delta t$ with $\eta \in \mathbb{N}$. In this case, the DPP extends over multiple Euler steps. This means that we have to take multiple steps of the markov chain approximation and then apply the expectation which complexifies the calculation:

1. The Euler-Marayuma approximation does not change except that the coefficient do not depend on time, the controls are constant for $\eta$ time steps in a row and the increment are not scaled by $\sqrt{h}$ but by $\sqrt{\Delta t}$.

2. Let us define $\{Y_i^{(x,a)}\}_{j=0,\ldots,\eta}$ the solution to:

$$\begin{cases} Y_{j+1} = Y_j + \Delta t b_a(Y_j) + \sigma_a(Y_j)\Delta W_j, \\ Y_0 = x \end{cases}$$

The dynamic programming principle changes in that it's no longer the expectation for one increment but for $\eta$ ones and we are not evaluating the value function at the next time but at $\eta$ times later indeed, $t_n + h = t_n + \eta\Delta t = t_{n+\eta}$.

$$v_h(t_n, x) = \sup_{a\in A} \mathbb{E} \left[ \sum_{j=0}^{\eta-1} \Delta t f_a(Y_j) + v_h(t_{n+\eta}, Y_\eta) \right].$$

3. We now need to perform this Markov chain approximation but for each of the $\eta+1$ $Y_i$. Since the normal increment are i.i.d. we can still use the $M$ points $\xi_i$ and their weights $\lambda_i$:

$$\begin{cases} Y_{j+1} = Y_j + \Delta t b_a(Y_j) + \sqrt{\Delta t}\sigma_a(Y_j)\zeta_i, \\ Y_0 = x \end{cases}$$

4. Approximating the expectation in the DPP via Gauss-Hermite quadrature method. Now, instead of defining the $M$ evaluation points implicitly by $x + hb_a(t_n, x) + \sqrt{h}\sigma_a(t_n, x)\xi_i$ and because we now do the expectation over $\eta$ increments, we need to consider the following $\frac{M^{\eta+1}-1}{M-1}$ evaluation points $\{\{x_i\}_{i\in\{1,\ldots,M\}^j}\}_{j=0,\ldots,\eta}$ defined recursively for $j = 1, \ldots, \eta - 1$:

$$\begin{cases} x_{(i,l)} = x_i + hb_a(t_n, x_i) + \sqrt{h}\sigma_a(t_n, x_i)\xi_l, \ \forall l \in \{1, \ldots, M\}, \ \forall i \in \{1, \ldots, M\}^j \\ x_0 = x, \end{cases}$$

and where $(i, l)$ is the concatenation of this multi index $i$ and the next index $l$. Defining the new weights $\Lambda_i = \prod_{j\in i} \lambda_i$ for any multiindex $i$, it gives the following updated quadrature for $n = 0, \ldots, N - \eta$ and $q = N - \eta + 1, \ldots, N$:

$$\begin{cases} v_h(t_n, x) = \sup_{a\in A} \left[ \sum_{j=0}^{\eta-1} \Delta t \sum_{i\in\{1,\ldots,M\}^j} \Lambda_i f_a(x_i) + \sum_{i\in\{1,\ldots,M\}^\eta} \Lambda_i v(t_{n+\eta}, x_i) \right], \\ v(t_q, x) = \sup_{a\in A} \left[ \sum_{j=0}^{N-q-1} \Delta t \sum_{i\in\{1,\ldots,M\}^j} \Lambda_i f_a(x_i) + \sum_{i\in\{1,\ldots,M\}^{N-q}} \Lambda_i g(x_i) \right]. \end{cases}$$

We are initializing the value function with it's Definition 1.3.

5. Using a multilinear interpolation in the same way as for the original scheme for the value function at the next time inside the DPP:

$$\begin{cases} v_h(t_n, x_m) = \sup_{a\in A} \left[ \sum_{j=0}^{\eta-1} \Delta t \sum_{i\in\{1,\ldots,M\}^j} \Lambda_i f_a(x_i) + \sum_{i\in\{1,\ldots,M\}^\eta} \Lambda_i \mathcal{I}[v](t_{n+\eta}, x_i) \right], \\ v(t_q, x_m) = \sup_{a\in A} \left[ \sum_{j=0}^{N-q-1} \Delta t \sum_{i\in\{1,\ldots,M\}^j} \Lambda_i f_a(x_i) + \sum_{i\in\{1,\ldots,M\}^{N-q}} \Lambda_i g(x_i) \right]. \end{cases}$$

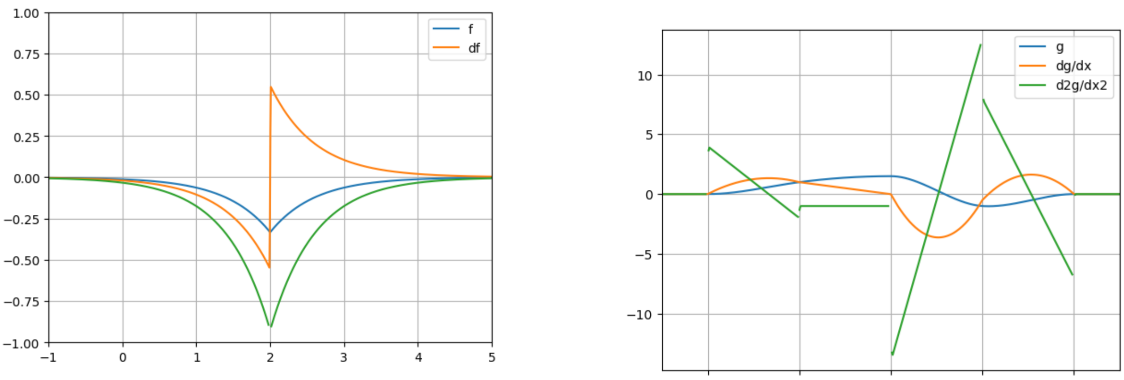
## 6.2 Results

In the following we apply the original scheme and the modified one to a problem satisfying the hypotheses we chose. We take $\Delta t \sim \Delta x^2$ for all cases but we will refine the grids to observe the rate of convergence. As the error measure, we use the $L^\infty$ norm between a value function at $t = 0$ and the one calculated for the finest grid at $t = 0$. All of our calculation take place between the terminal time $T = 1$ and 0 and the order of the error at rank $i$ is calculated in comparison to the next finer grid through $\frac{ln(\|e_{i-1}\|)-ln(\|e_i\|)}{ln(2)}$. We are refining the grids by doublig their number of time points so this theoretically gives the rate if it is sharp. For the second scheme, we do the same calculations but for $h = 2\Delta t$ and $h = 4\Delta t$, until we see the right order $\frac{1}{3}$ appear or until the calculation are too complicated for the author's computer. This will not give the right order and in a final attempt to reach said order we try to fit a curve $Ch^q$ with C and q the unknown while taking even less smooth data and coefficients. Throughout the tests, we kept the same inverse CFL condition that was in [10] and so the same number of points, for k we took $8 \times 2^k$ time points and $8 \times 2^{2k}$ space points. Finally, $\sigma = 0.4$. All calculation are done in python and one can find the code in the appendix A.

In the tests settings, we try to solely satisfy the hypothesis (H4) and (H5) but no more. The functions are the following

$$\begin{cases} b_a(x) = 0; \\ \sigma_a(x) = \sigma; \\ f_a(x) = -\frac{1}{3}e^{|x-(2-a)|\frac{5}{3}} \end{cases} \tag{6.1}$$

For the end cost $g$ we fitted a Hermite cubic spline. We choose those because they come with continuity and continuity of their derivative which force (H5) valid. The running cost and the end cost as well as their derivatives can be found in the Figure 1. They were choosen to fill in the regularities required for instance the end cost has bounded but not continuous second derivatives and lipschitz first derivatives. Moreover, they were taken to be zero outside of the original stencil $[-1, 5]$ so that we would not have to increase the size of the stencil for each time steps. Otherwise, for the points close enough to the border, for which $x_m + hb_a(t_n, x_m) + \sqrt{h}\sigma_a(t_n, x_m)\xi_i$ reaches outside of the stencil, we would have to have calculated the value function outside the stencil but since the costs rapidly go to 0 of those border, it is unnecessary.



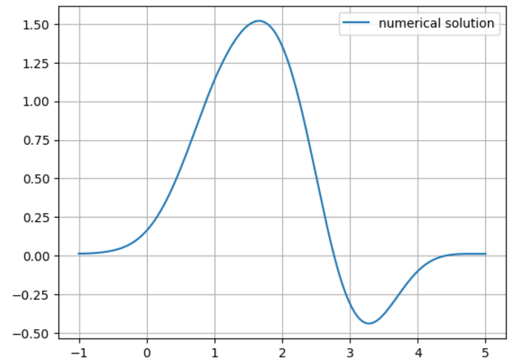

(a) Running cost and their derivatives for zero control.

(b) End cost and their derivatives.

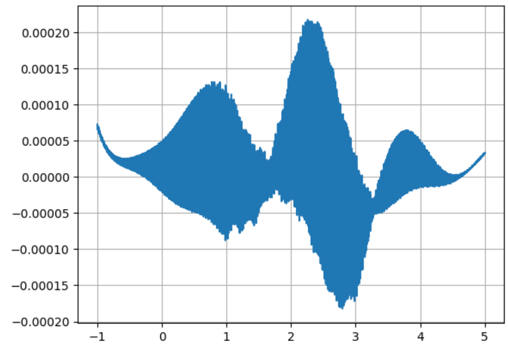
Figure 1: Costs graphs alongside their derivatives.

In Figure (2a), one can find the calculated numerical solution. Since the problem does not have a theoretical solution, it is the value function that was calculated for the finest grid. Next to it the difference between the two finest value functions show (as it was also pointed by [10]) that the error order is not the same in all part of the space. We can see that the error is highest on the three part of the space where the numerical solution has the highest derivative.

In Table 1 we report the error, the rates and the times to calculate the solutions for both schemes. The rates, are higher than 1 for the first scheme. This is because the solution seems to be smoother than anticipated and the fact that no computable solutions exist allows for error since we are already not comparing to an exact solution. We also implemented the second scheme. For the second scheme however, as it is displayed in Table 1 column 2 and 3, we were also not successful in showing this particular $\frac{1}{3}$ rate with this method of mesh refinement while keeping $h$




(a) Numerical solution: computed for $N = 8\times 2^8$ and $\frac{N^2}{8}$ points in the stencil.



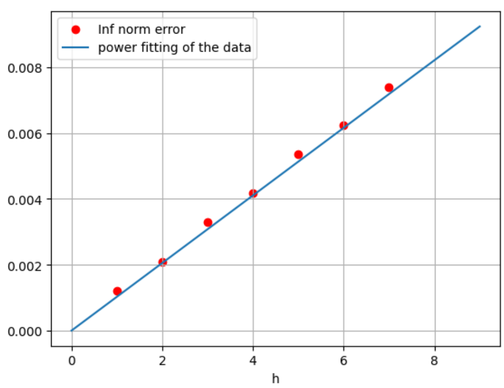
(b) Error between the value functions for the two finest grids.

constant. It seems once again that the optimal solution being so smooth prevent the rate from dropping to the lower bound.

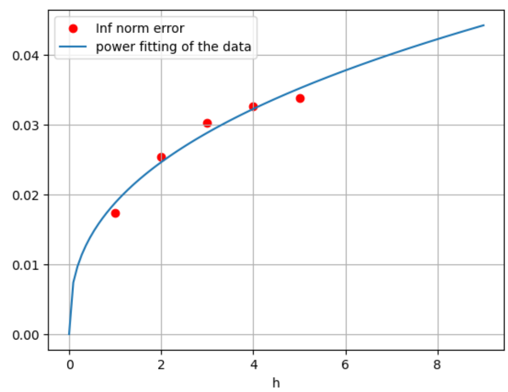
| | $\eta = 1$ | | | $\eta = 2$ | | | $\eta = 4$ | | |
|---|---|---|---|---|---|---|---|---|---|
| k | error | order | time(s) | error | order | time(s) | error | order | time(s) |
| 1 | 2.80e-1 | - | 0.01 | 2.16e-1 | - | 0.01 | 2.30e-1 | - | 0.01 |
| 2 | 6.51e-2 | 2.18 | 0.01 | 6.61e-2 | 1.70 | 0.01 | 7.65e-2 | 1.59 | 0.01 |
| 3 | 1.49e-2 | 2.26 | 0.02 | 2.21e-2 | 1.58 | 0.04 | 2.77e-2 | 1.46 | 0.18 |
| 4 | 3.72e-3 | 2.06 | 0.18 | 8.67-3 | 1.35 | 0.36 | 1.07e-2 | 1.37 | 1.66 |
| 5 | 8.41e-4 | 1.63 | 1.55 | 3.06e-3 | 1.50 | 3.86 | 3.23e-3 | 1.72 | 14.2 |
| 6 | 2.13e-4 | 1.98 | 19 | 1.17e-3 | 1.38 | 27.4 | - | - | - |
| 7 | 8.65e-5 | 1.30 | 107 | - | - | - | - | - | - |

Table 1: Error table computed with the first scheme for $\eta = 1$ and the second for 2 and 4.

For the last two tests, we try to fit a power curve to compare the results for similar grids but different $\eta$. In Table 2 one can find the result for the first setting and a new setting explained after. These result corroborate our guess as per the smoothness of the optimal solution, we can see the linearity in the error rate and in it's plotting and fitting in Figure 3.




(a) Almost smooth problem.



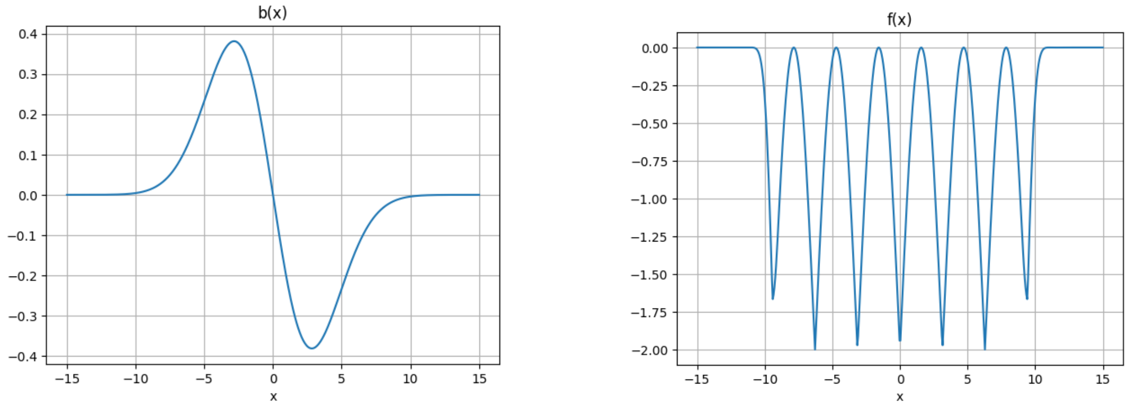

(b) Non smooth problem.

Figure 3: Fitting the errors.

Finally for the last test, we change the setting:

$$\text{(6.2)} \qquad \begin{cases} b_a(x) = \frac{-2}{3}x^{1/3}e^{(-(\frac{x}{4})^2)} + \frac{a}{4}; \\ \sigma_a(x) = (a-\sigma)^2; \\ f_a(x) = 2(|\sin(x)| - 1)e^{(-(|x|-9)_+^2)} - |a| \end{cases}$$

We kept the same end cost. One can find the graphs of the drift and running cost in Figure 4. We kept the same idea of sending the data to 0 ouside of the stencil but allowed for adjustment in the position of the interpolation points through the drift and diffusion now depending on the controls. However this is balanced by the running cost having a discount term $-|a|$ to prevent too much freedom in the choosing of the control. In Figure 5 we plotted the optimal solution. We can spot the kinks in the optimal solution that makes it semi-convex, for instance, close to 0 we can spot non smooth edges.




(a) Drift.

(b) Running cost.

Figure 4: Second setting.

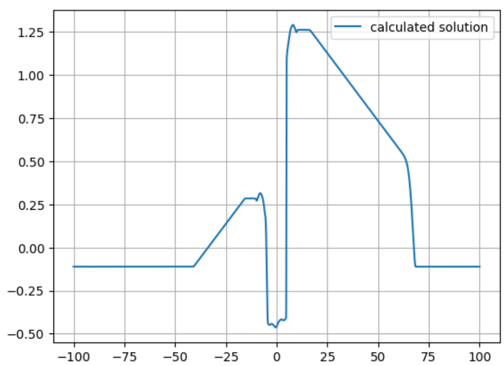



Figure 5: Optimal solution.

In Table 2 and Figure 3, one can find the error for $\eta = 1, \dots 4$ and the fitting of this data points. The result are $C = 1.87e-2$ and $q = 0.39$ and the variances are satisfying: $1.11e-3$ and $4.80e-2$. This shows the $\frac{1}{3}$ rate in this particular setting.

| | almost smooth | | non smooth | |
|---|---|---|---|---|
| $h$ | Error | Time (s) | Error | Time (s) |
| 1 | 1.21e-1 | 2.08 | 1.74e-1 | 2.56 |
| 2 | 2.08e-1 | 3.08 | 2.54e-1 | 5.51 |
| 3 | 3.28e-1 | 6.64 | 3.02e-1 | 12.4 |
| 4 | 4.18e-1 | 14.9 | 3.25e-1 | 26.6 |
| 5 | 5.36e-1 | 30.2 | 3.38e-1 | 51.0 |
| 6 | 6.23e-1 | 60.9 | - | - |
| 7 | 7.38e-1 | 122 | - | - |

Table 2: Error table computed with $h = \eta$ for both problems.

# 7 Conclusion

This thesis explores the hypothesis of semi-concavity and semi-convexity for the coefficients and data in calculating the error introduced by having piece-wise constant policies in a stochastic optimal control problem. Using the proof developed by Espen R. Jakobsen, Athena Picarelli and Christoph Reisinger in [6] we showed a $Ch^{\frac{1}{3}}$ bound for the error when the controls are constant on $h$ long time intervals. Taking the reader to the complex proof of this result, we have also shown the order 1 bound for the smooth case and the order $\frac{1}{2}$ bound for the deterministic and lipschitz cases. The improved $\frac{1}{3}$ error rate, although sharper is much harder to achieve because of the stricter hypotheses it requires. However the technique and the proof might prove useful in another setting where the time would not be inverted.

The thesis also proposed a new scheme based on Semi Lagrangian procedure and decoupling of the time over which the controls are constant $h$ of the time stencil $\Delta t$. Aiming at seeing the order of the error introduced by the constancy of the controls instead of the one introduced by the mesh size in the numerical tests, we developed several tests. We showed that in the case of a non smooth but semi convex optimal value function, we could reach the rate $\frac{1}{3}$ thanks to the differentiation of $h$ and $\Delta t$. On the second scheme that let us perform the test, while the thesis provides conjectures and assumptions as to how it should specifically show the error introduced by the constant controls, it didn't conduce any theoretical analysis for it's proposed scheme hence an area that would warrant further investigation could be the study of a similar decoupled scheme.

# A Code

```
# -*- coding: utf-8 -*-
"""MT_test_full.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1QbzzicOKKH_vsvERfWONAz4BDCuOUSJe
"""

import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import CubicHermiteSpline
import time

# problem definition:
sigma = 0.4

def b(x,a):
  return(0)

def s(x,a):
  return sigma*np.ones(x.shape + (1, ))

def f(x,a):
  return -np.exp(-abs(x - (2 - a))/0.6)/3

# Create cubic spline object
spline = CubicHermiteSpline(np.array([-1, 0, 1, 2, 3, 4, 5]), np.array([0, 0, 1, 3/2, -1, 0, 0]),

def g(x):
  return spline(x)*np.exp(-(x - 2)**2)

# parameter creation:
problem = (b, s, f, g)

interpolation = (np.array([[1, -1]]), np.array([1/2, 1/2]))

Nt = [(2**3)*(2**(k+1)) for k in range(8)]
dt = [1/n for n in Nt]
times = [np.linspace(0, 1, n) for n in Nt]

points = [np.linspace(-1, 5, (n**2)//8) for n in Nt]

A = np.array(np.linspace(-1, 1, 10))

# this function serves to compare two solution that have been calculated on unequal grids.
def bring_to_same_number_of_points(coarse_V, fine_V):
  ratio = len(fine_V)//len(coarse_V)
  coarse_V_finer = np.zeros(fine_V.shape)
  for i in range(len(coarse_V)):
    start_idx = i * ratio
    end_idx = (i + 1) * ratio
    coarse_V_finer[start_idx:end_idx] = coarse_V[i]
  return(coarse_V_finer)

def test_display(k, v, times, points, n=None):
  ListeV = []
  ListeTime = []
  for i in range(k):
```

```
    T = times[i]
    X = points[i]
    t = time.time()
    if(n is None):
      ListeV.append(v(problem, interpolation, T, X, A))
    else:
      ListeV.append(v(problem, interpolation, T, X, A, n))
    ListeTime.append(time.time()-t)

  DiffWithMostPrecise = []
  ListeError = []
  ListeOrder = []
  finest_V = ListeV[-1]

  for i in range(k):
    finer = bring_to_same_number_of_points(ListeV[i], finest_V)
    diff = finest_V - finer
    DiffWithMostPrecise.append(diff)
    ListeError.append(np.linalg.norm(diff, np.inf))

  for i in range(k - 1):
    order = (np.log(ListeError[i]) - np.log(ListeError[i+1]))/np.log(2)
    ListeOrder.append(order)

  print(ListeError)
  print(ListeOrder)
  print(ListeTime)
  return(ListeV, ListeTime, DiffWithMostPrecise, ListeError, ListeOrder)

"""# Problem when £h=\Delta t£ : """

def value_function(problem, interpolation, T, X, A):
  drift, diffusion, running_cost, end_cost = problem

  Nt = len(T)
  dt = T[1]-T[0]
  Xi, Lambda = interpolation

  # We took costs that are 0 outside of X so we don't need to expand the stencil.
  previous_V = end_cost(X)
  x, a = np.meshgrid(X, A)

  # Since the stencil is not changing, we can calulate the interpolation points once and for all
  x_plus_drift = np.expand_dims(x + drift(x, a), -1)
  sigma_M = np.sqrt(dt)*diffusion(x, a).dot(Xi)
  interpolation_points = x_plus_drift + sigma_M

  for i in range(1, Nt):

    J_alpha_M = np.interp(interpolation_points, X, previous_V)
    J_alpha = J_alpha_M.dot(Lambda)
    J_alpha = dt*running_cost(x, a) + J_alpha

    previous_V = J_alpha.max(axis = 0)
  return previous_V

"""# Problem when £h=\eta \Delta t£ : """

def value_function_ndt(problem, interpolation, T, X, A, n):
  drift, diffusion, running_cost, end_cost = problem
  Nt = len(T)
```

```
  dt = T[1]-T[0]
  Xi, Lambda = interpolation

  # We took costs that are 0 outside of X so we don't need to expand the stencil.
  # But now we need to keep track of the n last calculated values of V.
  previous_V = []
  x_alpha_mg = np.meshgrid(X, A)
  liste_interpoaltion_points = [x_alpha_mg[0]]
  liste_alpha = [x_alpha_mg[1]]

  # Constructing the different stencil that will be needed.
  for q in range(n):
    interpolation_points = liste_interpoaltion_points[-1]
    alpha = liste_alpha[-1]
    J_A_M = end_cost(interpolation_points)

    for k in range(q):
      J_A_M = J_A_M.dot(Lambda)
      J_A_M += dt*running_cost(liste_interpoaltion_points[-k-2], liste_alpha[-k-2])

    previous_V.append(J_A_M.max(axis=0))

    x_plus_drift = np.expand_dims(interpolation_points + drift(interpolation_points, alpha), -1)
    sigma_M = np.sqrt(dt)*diffusion(interpolation_points, alpha).dot(Xi)
    interpolation_points_M = x_plus_drift + sigma_M

    # We keep track of the interpolation points.
    # We also need to have the controls with augmented dimensions so that they are summed in the r
    liste_interpoaltion_points.append(interpolation_points_M)
    liste_alpha.append(np.expand_dims(liste_alpha[-1], axis=-1))

  for i in range(1, Nt-n):
    V_n = previous_V.pop(0)
    J_A_M = np.interp(liste_interpoaltion_points[-1], X, V_n)

    # This part serves to sum the n-1 running costs and to sum the quadrature points each time.
    for q in range(n):
      J_A_M = J_A_M.dot(Lambda)
      J_A_M += dt*running_cost(liste_interpoaltion_points[-q-2], liste_alpha[-q-2])

    previous_V.append(J_A_M.max(axis=0))
  return previous_V[-1]

"""# Fitting £Ch^q£ to the data with h varying"""

from scipy.optimize import curve_fit
def power(x,C,q):
  return C*x**q

# value function computation
ListeV_ndt = []
ListeTime_ndt = []
for i in range(8):
  T = times[4]
  X = points[4]
  t = time.time()
  ListeV_ndt.append(value_function_ndt(problem, interpolation, T, X, A, i+1))
  ListeTime_ndt.append(time.time()-t)

DiffWithMostPrecise_ndt = []
ListeError_ndt = []
```

```
ListeOrder_ndt = []

fine_V = ListeV_ndt[0] #finest_V
for i in range(len(ListeV_ndt)-1):
  finer = bring_to_same_number_of_points(ListeV_ndt[i+1], fine_V)
  diff = fine_V - finer
  DiffWithMostPrecise_ndt.append(diff)
  ListeError_ndt.append(np.linalg.norm(diff, np.inf))

for i in range(len(ListeError_ndt)-1):
  order = (np.log(ListeError_ndt[i]) - np.log(ListeError_ndt[i+1]))/np.log(2)
  ListeOrder_ndt.append(order)

h = np.array([i+1 for i in range(len(ListeError_ndt))])
params, _ = curve_fit(power, h, np.array(ListeError_ndt))

"""# Looking for a less smooth problem"""

# problem definition:
sigma = 0.4

def b(x,a):
  return -x**1/3*np.exp(-(x/4)**2)*2/3 + a/4

def s(x,a):
  return np.expand_dims((a-sigma)**2, -1)

def f(x,a):
  return np.exp(-np.maximum(abs(x)-9, 0)**2)*(abs(np.sin(x))-1)*2 - abs(a)

# parameter creation:
problem = (b, s, f, g)

interpolation = (np.array([[1, -1]]), np.array([1/2, 1/2]))

Nt = [(2**3)*(2**(k+1)) for k in range(8)]
dt = [1/n for n in Nt]
times = [np.linspace(0, 1, n) for n in Nt]

points = [np.linspace(-100, 100, (n**2)//8) for n in Nt]

A = np.array(np.linspace(-1, 1, 10))

# value function computation
ListeV_ndt = []
ListeTime_ndt = []
for i in range(6):
  T = times[4]
  X = points[4]
  t = time.time()
  ListeV_ndt.append(value_function_ndt(problem, interpolation, T, X, A, i+1))
  ListeTime_ndt.append(time.time()-t)

DiffWithMostPrecise_ndt = []
ListeError_ndt = []
ListeOrder_ndt = []

fine_V = ListeV_ndt[0] #finest_V
for i in range(len(ListeV_ndt)-1):
  finer = bring_to_same_number_of_points(ListeV_ndt[i+1], fine_V)
  diff = fine_V - finer
```

```
  DiffWithMostPrecise_ndt.append(diff)
  ListeError_ndt.append(np.linalg.norm(diff, np.inf))

for i in range(len(ListeError_ndt)-1):
  order = (np.log(ListeError_ndt[i]) - np.log(ListeError_ndt[i+1]))/np.log(2)
  ListeOrder_ndt.append(order)

h = np.array([i+1 for i in range(len(ListeError_ndt))])
params, pcov = curve_fit(power, h, np.array(ListeError_ndt), p0=(1,1/3))
```

# Bibliography


[1] Fabio Camilli and Maurizio Falcone. "An approximation scheme for the optimal control of diffusion processes". en. In: *M2AN - Modélisation mathématique et analyse numérique* 29.1 (1995), pp. 97–122. URL: http://www.numdam.org/item/M2AN_1995__29_1_97_0/.

[2] Kristian Debrabant and Espen R. Jakobsen. "Semi-Lagrangian schemes for linear and fully non-linear diffusion equations". In: *Mathematics of Computation* 82.283 (Dec. 2012), pp. 1433–1462. DOI: 10.1090/s0025-5718-2012-02632-9. URL: https://doi.org/10.1090%5C%2Fs0025-5718-2012-02632-9.

[3] Ishii Dolcetta. "Approximate solutions of the bellman equation of deterministic control theory". In: *Applied Mathematics and Optimization* 11 (Oct. 1983), pp. 161–181. DOI: 10.1007/BF01442176.

[4] Maurizio Falcone. "A numerical approach to the infinite horizon problem of deterministic control theory". In: *Applied Mathematics and Optimization* 15 (Feb. 1987), pp. 1–13. DOI: 10.1007/BF01442644.

[5] P.E. Souganidis G. Barles. "Convergence of approximation schemes for fully nonlinear second order equations". In: *Asymptotic Analysis* 4 (June 1990), pp. 271–283.

[6] Espen R. Jakobsen, Athena Picarelli, and Christoph Reisinger. "Improved order 1/4 convergence for piecewise constant policy approximation of stochastic control problems". In: *Electronic Communications in Probability* 24.none (2019), pp. 1–10. DOI: 10.1214/19-ECP256. URL: https://doi.org/10.1214/19-ECP256.

[7] N. Krylov. "Approximating Value Functions for Controlled Degenerate Diffusion Processes by Using Piece-Wise Constant Policies". In: *Electron. J. Probab.* 4 (1999), no. 2, 1–19. ISSN: 1083-6489. DOI: 10.1214/EJP.v4-39. URL: http://ejp.ejpecp.org/article/view/39.

[8] N. V. Krylov. "Mean value theorems for stochastic integrals". In: *The Annals of Probability* 29.1 (2001), pp. 385–410. DOI: 10.1214/aop/1008956335. URL: https://doi.org/10.1214/aop/1008956335.

[9] N.V. Krylov. "On the rate of convergence of finite-difference approximations for Bellmans equations with variable coefficients". In: *Probab Theory Relat Fields* 117 (May 2000), pp. 1–16. DOI: 10.1007/s004400050264.

[10] Athena Picarelli and Christoph Reisinger. *Probabilistic error analysis for some approximation schemes to optimal control problems*. 2020. arXiv: 1810.04691 [math.OC].

[11] Christoph Reisinger and Peter Forsyth. *Piecewise Constant Policy Approximations to Hamilton-Jacobi-Bellman Equations*. 2016. arXiv: 1503.05864 [math.NA].

[12] Piermarco Cannarsa Carlo Sinestrari. *Semiconcave Functions, Hamilton–Jacobi Equations, and Optimal Control*. Birkhäuser Boston, 1957. ISBN: 0-8176-4084-3.