Noname manuscript No.
(will be inserted by the editor)

Some non monotone schemes for time dependent Hamilton-Jacobi-Bellman equations in stochastic control

Xavier Warin

the date of receipt and acceptance should be inserted later

Abstract We introduce some approximation schemes for linear and fully non-linear diffusion equations of Bellman type. Based on modified high order interpolators, the schemes proposed are not monotone but one can prove their convergence to the viscosity solution of the problem. Some of these schemes are related to a scheme previously proposed without proof of convergence. Effective implementation of these schemes in a parallel framework is discussed. They are extensively tested on some simple test case, and on some difficult ones where theoritical results of convergence are not available.

Keywords Hamilton-Jacobi-Bellman equations, stochastic control, numerical methods, semi-lagrangian

Mathematics Subject Classification (2000) MSC 49L20, MSC 65N12

1 Introduction

We are interested in a classical stochastic control problem whose value function is solution of the following Hamilton Jacobi equations:

$$\begin{aligned} \frac{\partial v}{\partial t}(t,x) - \inf_{a\in A}\Big(\frac{1}{2}tr(\sigma_a(t,x)\sigma_a(t,x)^T D^2 v(t,x)) + b_a(t,x)Dv(t,x) \\ +c_a(t,x)v(t,x) + f_a(t,x)\Big) = 0 \text{ in } Q \\ v(0,x) = g(x) \text{ in } \mathbf{R}^d \qquad (1) \end{aligned}$$

where $Q := (0,T] \times \mathbf{R}^d$, A is a compact metric space. $\sigma_a(t,x)$ is a $d \times q$ matrix and so $\sigma_a(t,x)\sigma_a(t,x)^T$ is a $d \times d$ symmetric matrix, the b_a and f_a coefficients

X. Warin
EDF R&D & FiME, Laboratoire de Finance des Marchés de l'Energie (www.fime-lab.org)
Tel: +33-1-47654184
E-mail: xavier.warin@edf.fr

are functions defined on Q with values respectively in $\mathbf{R}^d$ and R.
Let's introduce an $\mathbf{R}^d$-valued controled process $X_s^{x,t}$ defined on a filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ by

$$dX_s^{x,t} = b_{a_s}(t, X_s^{x,t})ds + \sigma_{a_s}(s, X_s^{x,t})dW_s$$
$$X_t^{x,t} = x$$

where $a = (a_s)$ is a progressive process with values in A. This kind of problem arise when you want to minimize a cost function

$$J(t, x, a) = \mathbb{E}[\int_t^T f_{a_s}(s, X_s^{x,t}) e^{\int_t^s c_{a_s}(u, X_u^{x,t})du} ds + e^{\int_t^T c_{a_s}(u, X_u^{x,t})du} g(X_T^{x,t})]$$

with respect to the control a. It is well known [1] that the optimal value $\hat{J}(t, x) = \inf_a J(T - t, x, a)$ is a viscosity solution of equation (1).
Several approaches exist to solve this problem:

- The first approach is the generalized finite differences method developed by Bonnans Zidani [2] where the derivatives are approximated taking some non directly adjacent points. Directions are chosen such that the operator is consistent and it is monotone. Barles Souganidis framework [3] can be used to prove that the scheme converges to the viscosity solution of the problem.
- The second is the semi-Lagrangian approach developed by Camilli Falcone for example [4], generalized by Munos Zidani [5] and studied in detail by Debrabant Jakobsen [6]. In this approach, the scheme is discretized in time with a step h, the brownian motion is discretized taking two values of the order of $\sqrt{h}$. The scheme still follows the Barles Souganidis framework.
- The third approach is based on Monte Carlo techniques and the resolution of a Second Order Backward Stochastic Differential Equation. Initially developed by Fahim, Touzi, Warin [7] for two particular schemes, it has been generalized for degenerated HJB equations by Tan [8]. The convergence of the scheme to the viscosity solution is still given by the Barles Souganidis framework.

We will look at the use of semi-Lagrangian schemes for solving the control problem (1). These schemes have been studied in detail recently by Debrabant and Jakobsen [6] but for a low degree interpolator (typically linear) that gives a monotone operator. Under an assumption of CFL type, they show that the schemes are converging to the viscosity solution of the problem and using the method of shaking coefficients [9], they provide an estimate of the rate of convergence. Finally, they develop a higher order scheme but without proof of convergence to the viscosity solution of the problem.
In this article, we will look at non monotone approximations of higher degree. When we locally modify the interpolator in some specific cases, we show the convergence of such schemes to the viscosity solution of the problem and give an estimate of the error based on the fineness of the mesh. As a result, in particular, we prove that Debrabant-Jakobsen higher order scheme is convergent.

This idea of using high interpolators is not new : [10] already have emphasized the fact that monotonicity could be relaxed and our approach is linked to almost-monotone schemes. Results on stability and convergence for high order schemes interpolators in Semi Lagrangian schemes are already given by the Italian School for first order Hamilton-Jacobi in [11]. Results for high order Semi Lagrangian schemes for advection equations can be found in [12,13].
The structure of the paper is as follows: In the first part, we give the notations and some classical results of existence and uniqueness of the solution of (1). After time discretization, we show for some general approximations the convergence of the discrete solution to the viscosity solution of the problem if we can solve the optimization problem obtained at each time step. We explain why the monotony is not necessary to obtain convergence toward the viscosity solution: a scheme converging under certain assumptions always converges to the viscosity solution.
In the second part we develop several Lagrange interpolators, spline interpolators, and approximations based on Bernstein polynomials converging.
The last part, the techniques to properly treat the boundary conditions and parallelization methods are developed so that problems of dimension greater than 2 can be tackled. On different test cases, we show that the developed schemes are effective even in cases where the theory gives no evidence of convergence (unbounded control).

2 Notation and regularity results

We denote by $\wedge$ the minimum and $\vee$ the maximum. We denote by $|\quad|$ the Euclidean norm of a vector. For a bounded function w, we set

$$|w|_0 = \sup_{(t,x)\in Q} |w(t,x)|, \qquad [w]_1 = \sup_{(s,x)\neq(t,y)} \frac{|w(s,x) - w(t,y)|}{|x-y| + |t-s|^{\frac{1}{2}}}$$

and $|w|_1 = |w|_0 + [w]_1$. $C_1(Q)$ will stand for the space of functions with a finite $|\quad|_1$ norm.
For t given, we denote

$$||w(t,.)||_\infty = \sup_{x\in\mathbf{R}^d} |w(t,x)|$$

We use the classical assumption on the data of (1) for a given $\hat{K}$:

$$\sup_a |g|_1 + |\sigma_a|_1 + |b_a|_1 + |f_a|_1 + |c_a|_1 \le \hat{K} \tag{2}$$

A classical result [14] gives us the existence and unicity of the solution in the space of bounded Lipschitz functions:

Proposition 1 *If the coefficients of the equation* (1) *satisfy* (2)*, there exists a unique viscosity solution of the equation* (1) *belonging to* $C_1(Q)$*. If* u_1 *and* u_2 *are respectively sub and supersolution of equation* (1) *satisfying* $u_1(0,.) \le u_2(0,.)$ *then* $u_1 \le u_2$*.*

A spatial discretization length of the problem Δx being given, thereafter $(i_1 \Delta x, .., i_d \Delta x)$ with $\bar{i} = (i_1, ..., i_d) \in \mathbf{Z}^d$ will correspond to the coordinates of a mesh $M_{\bar{i}}$ defining a hyper-cube in dimension d. For an interpolation grid $(\xi_i)_{i=0,..N} \in [-1,1]^N$, and for a mesh $\bar{i}$, the point $y_{\bar{i},\tilde{j}}$ with $\tilde{j} = (j_1, .., j_d) \in [0,N]^d$ will have the coordinate $(\Delta x(i_1 + 0.5(1+\xi_{j_1})), .., \Delta x(i_d + 0.5(1+\xi_{j_d}))$. We denote $(y_{\bar{i},\tilde{j}})_{\bar{i},\tilde{j}}$ the set of all the grids points on the whole domain.
We notice that for regular mesh with constant volume Δx^d, we have the following relation for all $x \in \mathbf{R}^d$:

$$\min_{\bar{i},\tilde{j}} |x - y_{\bar{i},\tilde{j}}| \leq \Delta x \tag{3}$$

Finally, from one line to the other constants C may be changed.

3 General discretization

The equation (1) is discretized in time by the scheme proposed by Camilli Falcone [4] for a time discretization h.

$$\begin{aligned} v_h(t+h,x) \;=\; & \inf_{a \in A} \left[\sum_{i=1}^{q} \frac{1}{2q} (v_h(t, \phi^+_{a,h,i}(t,x)) + v_h(t, \phi^-_{a,h,i}(t,x))) \right. \\ & \left. + f_a(t,x)h + c_a(t,x) h v_h(t,x) \right] \\ := \; & v_h(t,x) + \inf_{a \in A} L_{a,h}(v_h)(t,x) \end{aligned} \tag{4}$$

with

$$\begin{aligned} L_{a,h}(v_h)(t,x) &= \sum_{i=1}^{q} \frac{1}{2q} (v_h(t, \phi^+_{a,h,i}(t,x)) + v_h(t, \phi^-_{a,h,i}(t,x)) - 2 v_h(t,x)) \\ & \quad + h c_a(t,x) v_h(t,x) + h f_a(t,x) \\ \phi^+_{a,h,i}(t,x) &= x + b_a(t,x) h + (\sigma_a)_i(t,x) \sqrt{hq} \\ \phi^-_{a,h,i}(t,x) &= x + b_a(t,x) h - (\sigma_a)_i(t,x) \sqrt{hq} \end{aligned}$$

where $(\sigma_a)_i$ is the i-th column of σ_a. We note that it is also possible to choose other types of discretization in the same style as those defined in [5].
In order to define the solution at each date, a condition on the value chosen for v_h between 0 and h is required. We choose a time linear interpolation once the solution has been calculated at date h:

$$v_h(t,x) = (1 - \frac{t}{h}) g(x) + \frac{t}{h} v_h(h,x), \forall t \in [0,h]. \tag{5}$$

We first recall the following result :

Proposition 2 *Under the condition on the coefficients given by equation* (2)*, the solution* v_h *of equations* (4) *and* (5) *is uniquely defined and belongs to* $C_1(Q)$*. We check that if* $h \leq (16 \sup_a \left\{|\sigma_a|_1^2 + |b_a|_1^2 + 1\right\} \wedge 2 \sup_a |c_a|_0)^{-1}$*, there exists* C *such that*

$$|v - v_h|_0 \leq C h^{\frac{1}{4}} \tag{6}$$

Moreover, there exists C *independent of* h *such that*

$$|v_h|_0 \leq C \tag{7}$$

$$|v_h(t,x) - v_h(t,y)| \leq C|x-y|, \forall (x,y) \in Q^2 \tag{8}$$

Proof The existence of a solution in $C_1(Q)$ is an application of the proposition 8.4 in [6]. The error estimate corresponds to the theorem 7.2 in [6]. The existence of a Lipschitz constant uniform in x and the uniform bound are given by the corollary 8.3 in the same article.

It is assumed throughout this section for simplicity that N is fixed. For a function v from $\mathbf{R}^d$ to $\mathbf{R}$, we denote the set of all the values taken by v on the grids $(y_{\bar{i},\tilde{j}})_{\bar{i},\tilde{j}}$ by $(v^{\bar{i},\tilde{j}})_{\bar{i},\tilde{j}}$. We define the operator T_ρ ($\rho = (h, \Delta x)$) with values in $C(\mathbf{R}^d)$ such that $T_\rho((v^{\bar{i},\tilde{j}})_{\bar{i},\tilde{j}})$ is an approximation of v (not necessarily an interpolation). In the sequel we still note T_ρ the operator defined on the set of function v from $\mathbf{R}^d$ to $\mathbf{R}$ by $T_\rho v := T_\rho((v^{\bar{i},\tilde{j}})_{\bar{i},\tilde{j}})$.
We consider the HJB equation discretized at the grids points:

$$\begin{aligned} v_\rho^{\bar{i},\tilde{j}}(t+h) = {} & v_\rho^{\bar{i},\tilde{j}}(t) \\ & + \inf_{a \in A} \left[(L_{a,h} T_\rho((v_\rho^{\bar{k},\tilde{l}}(t))_{\bar{k},\tilde{l}}))(t, y_{\bar{i},\tilde{j}}) \right] \end{aligned} \tag{9}$$

with a linear interpolation between 0 and h following (5).
We denote $\tilde{v}_\rho(t) = T_\rho((v_\rho^{\bar{i},\tilde{j}}(t))_{(\bar{i},\tilde{j})})$ the reconstructed solution in $C(R^d)$.
We will now describe the approximation operator so that the scheme converges to the viscosity solution. We will extend the notion of weight developed by [6].

Assumption 1 *Suppose that* T_ρ *is an operator from* $C_0(Q)$ *to* $C_0(Q)$*, that there exists a function of* h $\tilde{K}_h \xrightarrow{h \longrightarrow 0} 0$ *such that for* $x \in M_{\bar{i}}$

$$(T_\rho f)(x) = \sum_{\tilde{j} \in [0,N]^d} (w^h_{\bar{i},\tilde{j}}(f))(x) f(y_{\bar{i},\tilde{j}}) \tag{10}$$

$$0 \leq (1 - \tilde{K}_h h) \leq \sum_{\tilde{j} \in [0,N]^d} (w^h_{\bar{i},\tilde{j}}(f))(x) \leq 1 + \tilde{K}_h h \tag{11}$$

and that the functions $w^h_{\bar{i},\tilde{j}}(f)$ *are positive weights functions depending on* f*,* h *and the support* $M_{\bar{i}}$*.*

Remark 1 The previous operator is more general than the one defined by [6]:

- A priori it depends on h. It allows us to accept some reconstructed solutions even if small oscillations are present.

- We don't impose $w^h_{\bar{i},\tilde{j}}(f)(y_{\bar{k},\tilde{l}}) = \delta_{\bar{i}\bar{k}}\delta_{\tilde{j},\tilde{l}}$ such that the approximation operator is not necessarily an interpolation operator.
- The weight depends on the fonction f which is used.

The following theorem shows that any approximation operator satisfying the above assumptions converges to the viscosity solution.

Theorem 1 *Suppose T_ρ satisfies the assumptions 1. We consider a sequence $\rho_p = (h_p, \Delta x_p) \longrightarrow (0,0)$ such that $\frac{\Delta x_p}{h_p} \longrightarrow 0$, $h_p \leq 1$. Let's build a solution $\tilde{v}_{\rho_p}$ of (9) for all p, then $\tilde{v}_{\rho_p}$ converges to the viscosity solution of (1). Moreover for h_p small enough there exists C independent on h_p, N, Δx_p such that*

$$|\tilde{v}_\rho - v|_0 \leq C(h_p^{\frac{1}{4}} + \frac{\Delta x_p}{h_p} + \tilde{K}_{h_p}) \tag{12}$$

Proof Choose $h \leq 1$ and satisfying the hypothesis of proposition (2). We directly estimate $\tilde{v}_\rho - v_h$. Introduce

$$e(t) = ||\tilde{v}_\rho(t,.) - v_h(t,.)||_\infty$$

By definition of T_ρ, for a given point x in $M_{\bar{i}}$

$$\begin{aligned}
|\tilde{v}_\rho(t,x) - v_h(t,x)| \leq & |\sum_{\tilde{j}\in[0,N]^d} w^h_{\bar{i},\tilde{j}}(\tilde{v}_\rho)(x)(v_\rho^{\bar{i},\tilde{j}}(t) - v_h(t,x))| \\
& +|\sum_{\tilde{j}\in[0,N]^d} w^h_{\bar{i},\tilde{j}}(\tilde{v}_\rho)(x) - 1| \quad |v_h(t,x)| \\
\leq & \sum_{\tilde{j}\in[0,N]^d} w^h_{\bar{i},\tilde{j}}(\tilde{v}_\rho)(x))|v_\rho^{\bar{i},\tilde{j}}(t) - v_h(t,x)| \\
& +\tilde{K}_h h|v_h|_0 \\
\leq & (1 + \tilde{K}_h h)|v_\rho^{\bar{i},\tilde{k}} - v_h(t,x)| \\
& +\tilde{K}_h h|v_h|_0
\end{aligned} \tag{13}$$

with $y_{\bar{i},\tilde{k}}$ such that $|v_\rho^{\bar{i},\tilde{k}}(t) - v_h(t,x)|$ maximizes $|v_\rho^{\bar{i},\tilde{j}}(t) - v_h(t,x)|$.
Moreover we denote

$$\begin{aligned}
(\hat{L}_{a,h}v)(t,x) = & \frac{1}{2q}\sum_{i=1}^{q}(v(t,\phi^+_{a,h,i}(t,x)) + v(t,\phi^-_{a,h,i}(t,x))) \\
& +hc_a(t,x)v(t,x) + hf_a(t,x)
\end{aligned} \tag{14}$$

such that $V := v_\rho^{\bar{i},\tilde{k}}(t) - v_h(t,x)$ satisfies

$$V = \inf_a[(\hat{L}_{a,h}\tilde{v}_\rho)(t-h, y_{\bar{i},\tilde{k}}) - \inf_a(\hat{L}_{a,h}v_h)(t-h,x)]$$

So using $|inf. - inf.| \leq sup|. - .|$ we get

$$|V| \leq \frac{1}{2q}\sum_{i=1}^{q}\Bigg[\sup_a |\tilde{v}_\rho(t-h,\phi^+_{a,h,i}(t-h,y_{\bar{i},\tilde{k}})) \\ -v_h(t-h,\phi^+_{a,h,i}(t-h,x))| \\ +\sup_a |\tilde{v}_\rho(t-h,\phi^-_{a,h,i}(t-h,y_{\bar{i},\tilde{k}})) - v_h(t-h,\phi^-_{a,h,i}(t-h,x))|\Bigg] \\ +h\sup_a |c_a|_0|v_\rho^{\bar{i},\tilde{k}}(t-h) - v_h(t-h,y_{\bar{i},\tilde{k}})| + h|v_h|_0|c_a|_1\Delta x| \\ +h\sup_a |f_a|_1\Delta x$$

Using the fact that the data in (1) belong to $C_1(Q)$, such that

$$\begin{aligned} |\phi^-_{a,h,i}(t,y_{\bar{i},\tilde{k}}) - \phi^-_{a,h,i}(t,x)| &\leq \Delta x(1+\sup_a |b_a|_1 h + \sup_a |(\sigma_a)_i|_1\sqrt{hq}) \\ &\leq \Delta x(1+C(\sqrt{hq}+h)) \\ &\leq C\Delta x \end{aligned} \tag{15}$$

and using the fact that $|v_h|_1$ is bounded independently on h, one gets the estimate

$$|\tilde{v}_\rho(t-h,\phi^+_{a,h,i}(t-h,y_{\bar{i},\tilde{k}})) - v_h(t-h,\phi^+_{a,h,i}(t-h,x))| \leq \\ ||\tilde{v}_\rho(t-h,.) - v_h(t-h,.)||_\infty + \\ C|v_h|_1\Delta x$$

Using the fact that $|f_a|_1$, $|c_a|_1$ are bounded independently of a:

$$\begin{aligned} |V| &\leq ||\tilde{v}_\rho(t-h,.) - v_h(t-h,.)||_\infty + \\ &\quad +h\sup_a |c_a|_0|v_\rho^{\bar{i},\tilde{k}}(t-h) - v_h(t-h,y_{\bar{i},\tilde{k}})| \\ &\quad +C|v_h|_1\Delta x + h|c_a|_1|v_h|_0\Delta x + h\sup_a |f_a|_1\Delta x \\ &\leq ||\tilde{v}_\rho(t-h,.) - v_h(t-h,.)||_\infty(1+h\hat{K}) + \Delta x C \end{aligned}$$

where the constant C depends on $\tilde{K}$, $|v_h|_1$, $\hat{K}$.
So

$$|v_\rho^{\bar{i},\tilde{k}}(t) - v_h(t,x)| \leq e(t-h)(1+h\hat{K}) + C\Delta x \tag{16}$$

By combining the above equation with (13):

$$e(t) \leq (1+\tilde{K}_h h)(1+h\hat{K})e(t-h) + C(\Delta x + h\tilde{K}_h)$$

so there exists $\hat{C}$ such that

$$e(t) \leq (1+\hat{C}h)e(t-h) + C(\Delta x + h\tilde{K}_h)$$

Moreover applying the previous iteration at the first time step :

$$e(h) \leq (1 + \hat{C}h)|g|_0 + C(\Delta x + h\tilde{K}_h)$$

and by using the definition of v_ρ on $[0, h]$ given by (5)

$$e(t) \leq \frac{t}{h}\left[(1 + \tilde{K}_h h)|g|_0 + C(\Delta x + h\tilde{K}_h)\right] + (1 - \frac{t}{h})|g|_0, \forall t \leq h$$

Using the discrete Gronwall lemma

$$e(t) \leq C(\frac{\Delta x}{h} + \tilde{K}_h)e^{\hat{C}T} \forall t \leq T$$

Moreover by using $|\tilde{v}_{\rho_p} - v|_0 \leq |\tilde{v}_{\rho_p} - v_{h_p}|_0 + |v_{h_p} - v|_0$ and the proposition (2) we get the final result.

Remark 2 The suppositions on the weight function assure that the scheme is a nearly monotone one. The approximation of the function leads to a global scheme which is the perturbation of a monotone one and it is not surprising that ist is converging towards the viscosity solution (see remark 2.1 in [3])

Remark 3 Under assumption 1 with $\tilde{K}_h = 0$ we just impose that on each cell $M_{\hat{i}}$, given the values $(f(y_{\hat{i},\tilde{j}})_{\tilde{j}})$, all reconstructed values $(T_\rho f)(x)$ are in between the $\min_{\tilde{j}}(f(y_{\hat{i},\tilde{j}}))$ and $\max_{\tilde{j}}(f(y_{\hat{i},\tilde{j}}))$. Taking $\tilde{K}_h$ not null and decreasing to 0 permits to release the previous condition and to get reconstructed values slightly below $\min_{\tilde{j}}(f(y_{\hat{i},\tilde{j}}))$ or above $\max_{\tilde{j}}(f(y_{\hat{i},\tilde{j}}))$. An application of the result above permits to get convergence results for some new schemes in the section below and for some schemes in the litterature where authors could not get any convergence results.

Remark 4 Using a high order scheme on each mesh $M_{\tilde{i}}$ won't improve the theoretical rate of convergence, but improving the consistency at least locally we hope that the observe rate of convergence will be higher.

4 Some approximating operators

In this section we first develop some methods based on Lagrange interpolators and splines. We examine in particular the Lagrange interpolators using the Gauss Lobatto Legendre and Gauss Lobatto Chebyshev interpolators associated to some truncation. We will also consider the case of cubic splines and monotone cubic splines used by [6] which have the characteristic of not requiring truncation. In the last section, we will detail some polynomial approximation of Bernstein type that also do not require truncation and provide a monotone scheme.

4.1 Truncated Lagrange interpolators

For more information on the Lagrange interpolators and their properties, one can refer to Appendix (6.1). In this section, we suppose that a Lagrange interpolator with grid points $X = (\xi_i)_{i=0,N} \in [-1,1]^{N+1}$ is given. The space is discretized with meshes $M_{\bar{i}} = \prod_k [x_{i_k}, x_{i_k} + \Delta x]$ with $\bar{i} = (i_1, .., i_d)$ and on each mesh a Lagrange interpolator $I^X_{\Delta x,N}$ is defined by tensorization giving a multidimensional interpolator with $(N+1)^d$ points. We are particularly interested in Gauss Lobatto Chebyshev and Gauss Lobatto Legendre interpolators that have a low Lebesgue constant and thus avoid oscillations. On a mesh $M_{\bar{i}}$ and for a point x in this mesh, we note $\underline{v}_{\bar{i}} = \min_{\tilde{j}} v(y_{\bar{i},\tilde{j}})$, $\bar{v}_{\bar{i}} = \max_{\tilde{j}} v(y_{\bar{i},\tilde{j}})$. We introduce the following truncated operator:

$$\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(v) = (\underline{v}_{\bar{i}} - \tilde{K}_h h|\underline{v}_{\bar{i}}|) \vee I^X_{\Delta x,N}(v) \wedge (\bar{v}_{\bar{i}} + \tilde{K}_h h|\bar{v}_{\bar{i}}|)$$

where $\tilde{K}_h h < 1$ and $\tilde{K}_h \xrightarrow{h \longrightarrow 0} 0$.

Proposition 3 *The interpolator $\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}$ has the following properties:*

$$||\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x)||_\infty \leq (1 + \tilde{K}_h h)||f||_\infty$$

There exists $C_{N,d}$ such that for each Lipschitz bounded function f:

$$||\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) - f(x)||_\infty \leq (C_{N,d}\Delta x K + \tilde{K}_h h)|f|_1 \tag{17}$$

Proof The first assertion is obtained by definition of the truncation. The second assertion can be deduced from (22): if there is no truncation in x

$$|\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) - f(x)| \leq CK\Delta x \frac{(1 + \lambda_N(X))^d}{N+2}$$

where λ_N is the Lebesgue constant assiocated to interpolator. If there is truncation in x, for example a truncation to the maximum value, we note $y_{\bar{k},\tilde{l}}$ the point where $f(y_{\bar{i},\tilde{j}})$ is maximum and we suppose for instance that $f(y_{\bar{i},\tilde{j}}) \geq 0$. We have the relation:

$$\begin{aligned} |\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) - f(x)| &= |(1 + \tilde{K}_h h) f(y_{\bar{i},\tilde{j}}) - f(x)| \\ &\leq \tilde{K}_h h|f|_0 + |f(y_{\bar{i},\tilde{j}}) - f(x)| \\ &\leq (\tilde{K}_h h + K\Delta x)|f|_1 \end{aligned}$$

Of course the same result can be obtained with a minimum truncation.

Proposition 4 *The interpolator $\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}$ satisfies the assumptions* (1) *so that $\tilde{v}_{h,\Delta x,N}$ converges to the viscosity solution. Moreover*

$$||v - \tilde{v}_\rho||_\infty \leq O(h^{\frac{1}{4}}) + O(\frac{\Delta x}{h}) + O(\tilde{K}_h)$$

Proof Because of the truncation for each point x of a mesh $M_{\bar{i}}$, we have

$$(\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) = \underline{w}^h_{\bar{i}}(f)(x)\underline{f}_{\bar{i}} + \bar{w}^h_{\bar{i}}(f)(x)\bar{f}_{\bar{i}}$$

If $\underline{f}_{\bar{i}} \leq \hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) \leq \bar{f}_{\bar{i}}$ then

$$\begin{aligned}
\underline{w}^h_{\bar{i}}(f)(x) &= \frac{\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x) - \bar{f}_{\bar{i}}}{\underline{f}_{\bar{i}} - \bar{f}_{\bar{i}}} \\
\bar{w}^h_{\bar{i}}(f)(x) &= 1 - \underline{w}^h_{\bar{i}}(f)(x) \\
0 &\leq \underline{w}^h_{\bar{i}}(f)(x) \leq 1
\end{aligned}$$

If $\underline{f}_{\bar{i}} > \hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x)$

$$\begin{aligned}
\bar{w}^h_{\bar{i}}(f)(x) &= 0 \\
\underline{w}^h_{\bar{i}}(f)(x) &= \frac{\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x)}{\underline{f}_{\bar{i}}} \in [1 - \tilde{K}_h h, 1 + \tilde{K}_h h]
\end{aligned}$$

Otherwise

$$\begin{aligned}
\underline{w}^h_{\bar{i}}(f)(x) &= 0 \\
\bar{w}^h_{\bar{i}}(f)(x) &= \frac{\hat{I}^X_{h,\tilde{K}_h,\Delta x,N}(f)(x)}{\bar{f}_{\bar{i}}} \in [1 - \tilde{K}_h h, 1 + \tilde{K}_h h]
\end{aligned}$$

Then choose the weight functions above associated to the points with values associated to the extremal points $\underline{f}_{\bar{i}}$, $\bar{f}_{\bar{i}}$ and take a weight equal to 0 for other points. The final estimation is obtained by proposition (2) and theorem (1).

Remark 5 Due to this estimation the truncation should be such that $\tilde{K}_h = h^{\frac{1}{4}}$.

We also give the consistency error:

Proposition 5 *The consistency error is in* $O(h + \frac{\Delta x^2}{h})$ *in areas where the truncation is achieved and in* $O(h + \frac{\Delta x^{N+1}}{h})$ *otherwise.*

Proof Let u be the solution of (1) that we suppose regular. Defining

$$E(u) = \frac{1}{h}|\sup_a [u(t,x) - (L_{a,h}(\hat{I}^X_{h,\tilde{K}_h,\Delta x,N} u))(t-h,x)]| \tag{18}$$

we get

$$\begin{aligned}
E(u) \leq & \frac{1}{h} \sup_a |u(t,x) - (L_{a,h}u)(t-h,x)| \\
& + \sup_a |(L_{a,h}u)(t-h,x) - (L_{a,h}\hat{I}^X_{h,\tilde{K}_h,\Delta x,N} u)(t-h,x)| \\
\leq & \frac{1}{h} \sup_a |u(t,x) - (L_{a,h}u)(t-h,x)| \\
& + \frac{C}{h} |\hat{I}^X_{h,\tilde{K}_h,\Delta x,N} u - u|_0
\end{aligned}$$

using the assumption 2. Besides using consistency of the scheme with equation (1) and assumption 2:

$$\frac{1}{h}|u(t,x)-(L_{a,h}u)(t-h,x)| \leq Ch(|\frac{\partial^2 u}{\partial t^2}|_0 + |\frac{\partial^2 u}{\partial x^2}|_0 + |\frac{\partial^3 u}{\partial x^3}|_0 + |\frac{\partial^4 u}{\partial x^4}|_0)$$

The interpolation error is given by (21) when no truncation is achieved. When the truncation is effective for a point $x \in M_{\bar{i}}$, it means for example that the non truncated interpolator gives a value which is above all the values at the grid point of the mesh. So $I^X_{h,\tilde{K}_h,\Delta x,2}u(x) \leq \bar{u}_{\bar{i}} = \hat{I}^X_{h,\tilde{K}_h,\Delta x,N}u(x) \leq I^X_{h,\tilde{K}_h,\Delta x,N}u(x)$ and the approximation has an error in between $O(\Delta x^2)$ and $O(\Delta x^{N+1})$. Then the consistency error with this term is at least the one obtained by the linear interpolator.

4.2 Cubic spline interpolators

A cubic spline interpolation is used to interpolate a one-dimensional function. No discretization point inside the mesh is given. With $d = 1$, keeping the same notations as before, the grid points are $y_{\bar{i},\tilde{j}}$ with $\bar{i}$ the mesh number and $\tilde{j} = 0$ or 1 corresponding to the left or right part of the mesh. In particular $y_{\bar{i},1} = y_{\bar{i}+1,0}$.

4.3 Truncated cubic spline

Let $I^{X_c}_{\Delta x,1}$ be the cubic spline interpolator. We use the truncated interpolator:

$$\hat{I}^{X_c}_{h,\tilde{k}_h,\Delta x}(v) = (\underline{v}_{\bar{i}} - \tilde{K}_h h|\underline{v}_{\bar{i}}|) \vee I^{X_c}_{\Delta x,1}(v) \wedge (\bar{v}_{\bar{i}} + \tilde{K}_h h|\bar{v}_{\bar{i}}|)$$

It is clear that the interpolator satisfies the assumptions (1) and that we satisfy the assumptions of theorem (1).
As before, the consistency order depends on the fact the truncation has been performed or not.

Proposition 6 *The consistency error with the interpolator $\hat{I}^{X_c}_{h,\tilde{k}_h,\Delta x}$ is in $O(h+\frac{\Delta x^2}{h})$ in areas where the truncation is achieved and in $O(h+\frac{\Delta x^4}{h})$ otherwise.*

Proposition 7 *The solution $\tilde{v}_\rho$ obtained by the interpolator $\hat{I}^{X_c}_{h,\tilde{k}_h,\Delta x}$ converges to the viscosity solution v and the convergence rate is given by:*

$$||v - \tilde{v}_\rho||_\infty \leq O(h^{\frac{1}{4}}) + O(\frac{\Delta x}{h}) + O(\tilde{K}_h)$$

4.4 Monotone cubic spline ([6])

It is possible to modify the cubic spline algorithm to obtain a monotone interpolation by direction (but not globally monotone) so that the interpolated function is C_1. It is achieved by modifying the estimated derivatives used by the spline following the Eisenstat Jackson Lewis algorithm [16] (derived from the Fritsch-Carlson algorithm). This ensures the monotony of the interpolated function. Debrabant and Jakobsen have modified this algorithm by relaxing the continuity of the derivative so that the interpolation is reduced to a local problem on the mesh and adjacent cells. This interpolation is of order 4 in the mesh if the interpolated function is monotone. By tensorization, using Remark 5.1 in [6], the non-monotone interpolator operator in [6] in dimension d can be written:

$$\begin{aligned} I^S_{\Delta x}(f)(x) = & \sum_{\bar{i},\tilde{j}} w^h_{\bar{i},\tilde{j}}(f)(x) f(y_{\bar{i},\tilde{j}}) \\ \text{where} \quad & \text{the support of } w^h_{\bar{i},\tilde{j}}(f) \text{ is } M_{\bar{i}} \\ & w^h_{\bar{i},\tilde{j}}(f) \geq 0, \\ & \sum_{\bar{i},\tilde{j}} w^h_{\bar{i},\tilde{j}}(f)(x) = 1 \end{aligned} \tag{19}$$

It is clear that the interpolator satisfies assumption (1) and the assumptions of theorem (1). As shown in [6]

Proposition 8 *If the interpolated function $u(t-h,.)$ is monotone between grid points, the consistency error with the interpolator $I^S_{\Delta x}$ given by equation 18 is $O(h + \frac{\Delta x^4}{h})$*

As a direct result of theorem (1) we get the convergence of the scheme that was not given in [6] :

Proposition 9 *The solution $\tilde{v}_{h,\Delta x,1}$ obtained by interpolator $I^S_{\Delta x}$ converges to the viscosity solution of* (1) *and*

$$||v - \tilde{v}_\rho||_\infty \leq O(h^{\frac{1}{4}}) + O(\frac{\Delta x}{h})$$

Remark 6 In fact it is shown in [17] that when the data is non monotone, the Fritsch-Carlson type algorithm (which has been modified to get the Eisenstat Jackson Lewis algorithm) is only clipping the solution to the maximum of the interpolated points so is equivalent to a truncation with $\tilde{K}_h = 0$, so in that case $\hat{I}^S_{\Delta x} = I^S_{\Delta x}$. So the local consistency error is similar to the one obtained by the other scheme developed when truncation is achieved.

4.5 Approximation with Bernstein polynomials

The weights associated to Bernstein polynomials are positive (Appendix (6.2)), independent on the function. Their sum is equal to one and we get nearly all the assumptions used by [6] except the fact that this is not an interpolator. By using the results given in appendix (6.2) we deduce that

Proposition 10 *The scheme with Bernstein approximation B_N of degree N (in each dimension) converges to the viscosity solution of* (1) *with*

$$||v - \tilde{v}_\rho||_\infty \leq O(h^{\frac{1}{4}}) + O(\frac{\Delta x}{h})$$

and the consistency error is of order $O(h + \frac{\Delta x^2}{Nh})$.

5 Some numerical results

In this section we focus on techniques for effective implementation of Semi Lagrangian algorithms. We are interested in any special treatment of the boundary conditions that can avoid problems with this kind of algorithm. The parallelization strategy is investigated and on numerical examples, we calculate the rate of convergence of the different methods on conventional tests from [6] and [5]. We eventually use numerical tests with unbounded controls to show that the methods work even outside the theoretical framework of convergence. We insist that in our tests any meshes are taken: in particular, the discretizations do not respect the monotony of functions and discretization parameters are not chosen so that the approximation points are inside the domain. If the value of a function must be estimated outside the domain, the scheme is amended as indicated in the following paragraph. If no change is possible we truncate the solution projected on the edge of the domain. The order of the estimate may be lower than the theoretical one or the one given by [6] but closer to a real use of the schemes.

5.1 Boundary conditions

The boundary conditions are often problematic for PDEs and their treatment by the Semi Lagrangian methods exacerbates the problem. Indeed, if for example we solve a problem with b and σ constant for simplicity and if x is a mesh point near the edge then $x + bh + \sqrt{h}\sigma$ can be out of the domain resolution. This problem occurs if a point is too close to the edge and if the volatility is too large or the time steps too small. A first possibility which is quite natural is to interpolate the solution outside the domain or to set it to a given value. The interpolation is to be avoided as much as possible because it causes oscillations that can explode during resolution. Following some ideas in [19], the first trick is to modify the schema. You can often avoid fetching points outside the area by changing the points sought by the interpolation.

In Figure (1) we show how a 1D scheme starting from a point x may need a point out of the domain. The points used can be modified (respecting 'mean'

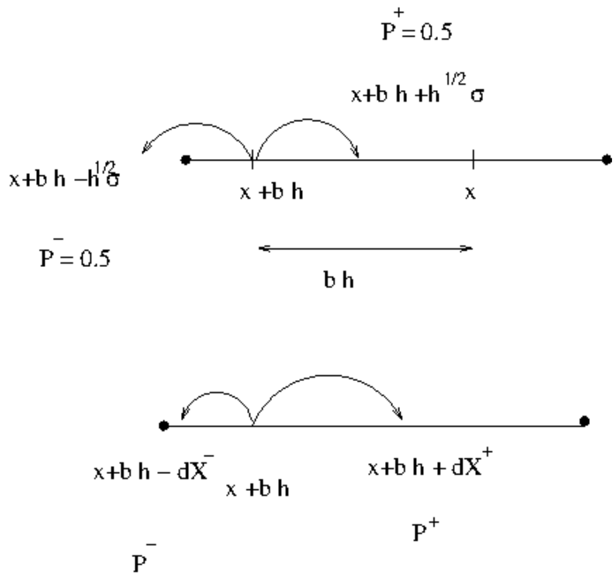

Fig. 1 Modification of the scheme for boundary conditions

and 'variance'). We denote dX^- the difference between $x + bh$ and the point reached by below and dX^+ the distance with the point reached by above. If the scheme is not modified, $dX^+ = dX^- = \sigma\sqrt{h}$ and the 'probability' to reach these points are $P^- = P^+ = \frac{1}{2}$. If a value has to be interpolated outside the domain, new weights and new interpolation points inside the domain are calculated respecting

$$\begin{aligned} dX^+ dX^- &= \sigma^2 h \\ P^+ &= \frac{\sigma^2 h}{(dX^+)^2 + \sigma^2 h} \\ P^- &= \frac{(dX^+)^2}{(dX^+)^2 + \sigma^2 h} \end{aligned}$$

In the general case this modification of the "probabilities" force us to modify the interpolation point in the other directions. In the corners of the domain the modification thus can be impossible and some kind of extrapolation has to be used.

Remark 7 The Bonnans Zidani method has the same flaw: when the scheme needs some points outside the domain, the consistency or the monotonicity has to be relaxed.

Remark 8 The use of this methods clearly doesn't satisfy the assumption in [6] and their results should be adapted to get convergence results similar to the one obtained in proposition 2. The consitency error due to time discretization (solving equation 4) cannot be better than $h^{\frac{1}{2}}$.

5.2 Parallelization technique

In order to solve a stochastic control problems in high dimension (3 or above) parallelization techniques are required. Of course thread parallelization can be easily added to these techniques. Suppose that we have 4 processors and that the grid of points is split between processor (figure (2)). At the initial date, each processor has its own data (the initial solution). At the first time step, each processor needs some data owned by other processors : some values needed by the interpolation. The control being bounded one can determine the envelop of the points needed by the processor. On figure (2), we give the data needed by processor 3 for its optimization. Then some MPI communications are realized

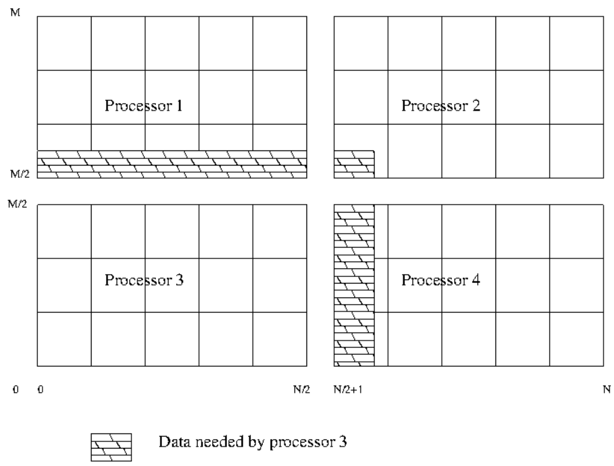

Fig. 2 Data to send to processor 3

once at each time step. This algorithm is effective because communications are just achieved only once at each time step and negligible in time spent. It has already been proved to be effective in dynamic programming problem in high dimension [20], [21]. Although it is specialy effective in high dimension permitting to tackle 4 dimensional problems, it remains attractive even on two dimensional problems. In figure below, we report acceleration obtained for test case (5.3.3) below on figure (3) in the special case where we have a number of time steps equal to 100, a number of mesh equal to 200 in each direction and a linear interpolator. It shows that even if the speed up is not linear, the efficiency of the parallelization is interesting.

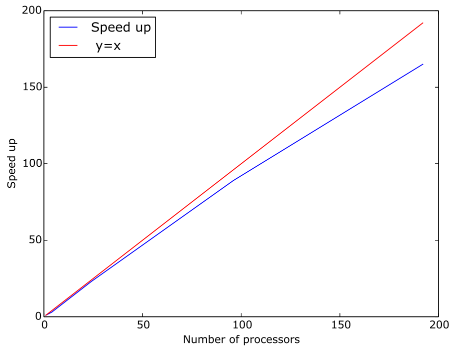

Fig. 3 Mpi acceleration for Semi-Lagragian Schemes for a 2D problem

5.3 Some test cases

In this section we give some results for the explicit scheme with polynomial approximation of Bernstein type (BERN i where i is the degree of the polynomial), with linear interpolation (LIN), with Chebyshev interpolation (TCHEB i), with Legendre interpolation (LEGEND i), with cubic splines (CUBIC), with monotone cubic splines (MPCSL following the name given by [6]). The first 3 are taken from the literature, but by extending the domain of resolution to get highly non-monotone solutions. The two last do not fit into the framework of the theory because the condition given by equation 2 is not verified. They are however interesting because the methods developed are effective. Unlike [6], we chose to set the same time step, and set the same control discretization for all the methods and all the space discretization in a given test case. By converging in space very thinly we should get some residual errors due to these fixed discretizations. All truncation are achieved with $K_h = 0$. In the tables, NbM correspond to the number of mesh per direction, Err the error with respect to the analytical or reference solution in infinite norm, Time correspond to the CPU times for the resolution, and Rate to the order of convergence numerically calculated: if $Err(n)$ is the error obtained with n meshes per direction, the order of convergence with $4n$ meshes is given by $\frac{log((Err(n)-Err(2n))/(Err(2n)-Err(4n))}{log2}$. As for the boundary treatment, extrapolation outside the domain is used at points when the methodology in section 5.1 is impossible to use. In each test case, the function g is obtained by taking the analytical solution with $t = 0$.

Table 1 Test case 1

LINEAR				CUBIC				MPCSL				TCHEB 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
240	0.310		112	20	0.461		2	20	0.815		3	20	0.165		31
480	0.119		448	40	0.037		10	40	0.166		10	40	0.0086		133
960	0.040	1.26	1815	80	0.005	2.84	41	80	0.007	4.62	42	80	0.00108	4.37	552
1920	0.0075	1.29	7334	160	0.0005	4.52	165	160	0.0005	4.88	170	160	0.0003	3.40	2246
LEGEND 2				LEGEND 3				BERN 2				BERN 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
20	0.059		6	20	0.165		21	120	0.643		718	120	0.5528		2811
40	0.0069		25	40	0.0085		92	240	0.227		2878	240	0.1784		11292
80	0.0010	3.15	104	80	0.00107	4.38	380	480	0.077	1.47	11551	480	0.0557	1.60	45446
160	0.0003	3.00	420	160	0.0003	3.41	1547	960	0.021	1.42	46467	960	0.01532	1.60	181897

5.3.1 First test case without control [6]

Coefficients are:

$$\begin{aligned} f_a(t,x) &= \sin x_1 \sin x_2((1+2\beta^2)(2-t)-1) \\ &\quad -2(2-t)\cos x_1 \cos x_2 \sin(x_1+x_2)\cos(x_1+x_2) \\ c_a(t,x) &= 0, \quad b_a(t,x) = 0 \quad \sigma_a(t,x) = \sqrt{2}\begin{pmatrix} \sin(x_1+x_2) & \beta & 0 \\ \cos(x_1+x_2) & 0 & \beta \end{pmatrix} \end{aligned}$$

We take $\beta = 0.1$ and solve the problem on $Q = (0,1] \times [-2\pi, 2\pi]^2$. The analytical solution is $u(t,x) = (2-t)\sin x_1 \sin x_2$. The number of time steps is equal to 2000 so $h = 5e-4$. Considering the results in table (1), we can conclude that on a regular linear problem:

- The order of convergence of the linear approximation is below 2. Because the solution is smooth we could hope to get a rate of convergence equal to 2 the consistency error for the LINEAR scheme : in fact we only get a rate equal to 1.3 certain due to the boundary treatment. For LEGEND of degree 2, the rate of convergence observed is 3 : so it is equal to the consistency rate observed certainly indicating that the truncation is not achieved. With the discretization tested the boundary condition doesn't seem to perturb the solution. For CUBIC, MPCSL, TCHEB, LEGEND of degre 3 the rate of convergence is roughly 4 but with some oscillations indicating that some truncations are achieved.
- The cost of Chebyshev is twice the cost of the Legendre polynomials: a analysis shows that this is due to trigonometric functions that are costly in time.
- The use of monotone spline is not superior to classical spline approximation with truncation.
- Bernstein polynomial are not competive
- The three most effective schemes are the CUBIC, MPCSL and LEGEND with degree 2.

5.3.2 A second test case without control [6]

Its solution is not regular

$$u(t,x) = (1+t)\sin(\frac{x_2}{2})\begin{cases} \sin\frac{x_1}{2} & \text{for } -2\pi < x_1 < 0 \\ \sin\frac{x_1}{4} & \text{for } 0 < x_1 < 2\pi \end{cases}$$

Table 2 Test case 2

LINEAR				CUBIC				MPCSL				TCHEB 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
640	0.038		557	80	0.00875		17	80	0.00875		18	20	0.0136		12
1280	0.013		2240	160	0.00439		70	160	0.00439		72	40	0.00398		51
2560	0.0070	1.84	8662	320	0.00220	0.99	285	320	0.00220	0.99	288	80	0.00128	1.84	212
5120	0.0035	0.99	34820	640	0.00110	0.99	1177	640	0.00110	1.00	1221	160	0.0005	1.97	858
LEGEND 2				LEGEND 3				BERN 2				BERN 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
20	0.01422		2	20	0.0137		9	80	0.3774		112	80	0.3144		429
40	0.00411		11	40	0.0040		39	160	0.1889		448	160	0.135		1734
80	0.00132	1.85	46	80	0.00129	1.84	160	320	0.066	1.80	1794	320	0.0460	1.43	6937
160	0.0006	1.96	184	160	0.0006	1.96	649	640	0.0186	1.89	7197	640	0.0128	1.80	27892

Table 3 Test case 2 : error near and far away the singularity

LINEAR					CUBIC				
	D_1		D_2			D_1		D_2	
NbM	Err	Rate	Err	Rate	NbM	Err	Rate	Err	Rate
80	0.12		1.21		40	0.017		0.010	
160	0.085		0.36		80	0.0087		0.0033	
320	0.052	0.16	0.14	0.55	160	0.0043	0.96	0.00086	1.60
640	0.027	0.46	0.038	1.15	320	0.0022	0.99	0.0005	2.73
1280	0.013	0.85	0.010	1.83	640	0.0011	0.99	0.0005	

with

$$
\begin{aligned}
f_a(t,x) &= \sin\frac{x_2}{2}\begin{cases} \sin\frac{x_1}{2}(1+\frac{1+t}{4})(\sin^2 x_1 + \sin^2 x_2) & \text{for } -2\pi < x_1 < 0 \\ \sin\frac{x_1}{4}(1+\frac{1+t}{16})(\sin^2 x_1 + 4\sin^2 x_2) & \text{for } 0 < x_1 < 2\pi \end{cases} \\
&\qquad - \sin x_1 \sin x_2 \cos\frac{x_2}{2}\begin{cases} \frac{1+t}{2}\cos\frac{x_1}{2} & \text{for } -2\pi < x_1 < 0 \\ \frac{1+t}{4}\cos\frac{x_1}{4} & \text{for } 0 < x_1 < 2\pi \end{cases} \\
c_a(t,x) &= 0, \quad b_a(t,x) = 0 \quad \sigma_a(t,x) = \sqrt{2}\begin{pmatrix} \sin x_1 \\ \sin x_2 \end{pmatrix}
\end{aligned}
$$

On take $Q = (0,1] \times [-2\pi, 2\pi]^2$, the number of time step is equal to 2000 so $h = 5e{-}4$. Our previous results are confirmed and here Lagrange polynomial of degree two are the most effective. Note that CUBIC and MPCSL give the same results for these discretizations (it is not true for more coarse discretizations not given here). As expected the convergence rate dropped due to singularity to high order schemes. But even in this case the high order scheme remains far more effective.

In order to check that the singularity was slowing the convergence rate, the domain D has been split into two parts. First part D_1 (singularity area) is for $x_1 \in [-\frac{\pi}{8}, \frac{\pi}{8}]$, while the second is $D_2 = D \setminus D_1$. The error and the convergence rate have been calculated for the two domains for LINEAR and CUBIC approximations in table (3). As for the LINEAR scheme, the error remains mainly higher in the D_2 domain explaining why the global rate of convergence remains high. As for the CUBIC scheme, the error remains always lower in the D_2 domain and all the rate of convergence of the D_1 domain correspond to the rate of convergence of the global domain.

Table 4 Test case 3

LINEAR				CUBIC				MPCSL				TCHEB 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
80	0.59		237	10	0.312		19	10	0.688		30	8	0.0986		47
160	0.147		850	20	0.0499		30	20	0.050		29	16	0.0119		184
320	0.044	2.09	3334	40	0.0072	2.61	96	40	0.0064	3.86	98	32	0.0012	3.01	735
640	0.014	1.77	13259	80	0.001	2.78	384	80	0.001	3.03	387	64	0.0008	4.94	2944
LEGEND 2				LEGEND 3				BERN 2				BERN 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
8	0.0710		14	8	0.0988		31	20	0.7479		181	20	0.769		758
16	0.0094		49	16	0.0117		116	40	0.706		789	40	0.5898		2362
32	0.0023	3.11	149	32	0.0011	3.04	465	80	0.3210	0.62	2533	80	0.2334	0.98	9436
64	0.0009	2.36	590	64	0.0010	6.24	1854	160	0.0801	2.09	10111	160	0.0563	1.00	37750

5.3.3 Control problem with a regular solution [6], [5]

The regular solution is given by

$$u(t, x_1, x_2) = (\frac{3}{2} - t) \sin x_1 \sin x_2$$

Coefficients are given by

$$\begin{aligned} f_a(t, x) = {} & (\frac{1}{2} - t) \sin x_1 \sin x_2 + (\frac{3}{2} - t) \Big[\sqrt{\cos^2 x_1 \sin^2 x_2 + \sin^2 x_1 \cos^2 x_2} \\ & -2 \sin(x_1 + x_2) \cos(x_1 + x_2) \cos x_1 \cos x_2 \Big] \\ c_a(t, x) = {} & 0, \quad b_a(t, x) = a \quad \sigma_a(t, x) = \sqrt{2} \begin{pmatrix} \sin(x_1 + x_2) \\ \cos(x_1 + x_2) \end{pmatrix}, \\ A = {} & \{a \in \mathbf{R}^2 : a_1^2 + a_2^2 = 1\} \end{aligned}$$

$Q = (0, 1] \times [-\pi, \pi]^2$ and the number of time steps is equal to 1000 so $h = 1e{-}3$, the number of control equal to 4000. CPU times are given for a number of core equal to 192. Once again the quadratic approximation Legend 2 is the most effective.

5.3.4 One dimensional optimization problem with unbounded control

The theory is developed for bounded controls. One may wonder if we are able to solve problems with unbounded control. We are interested in a stochastic target problem where we want to drive a portfolio towards the value 1 at T with a given probability x. The asset used for investment satisfies :

$$dS_t = \mu dt + \kappa dW_t$$

Supposing a null interest rate, the wealth process of an investor investing in bond and the asset follows

$$dX_t^\theta = \theta_t \mu X_t^\theta dt + \theta_t \kappa X_t^\theta dW_t$$

where θ_t is the investor strategy.
Using the methodology developped in [18], the minimal value u at date t of

the initial portfolio to reach a target 1 at date T with probability x satisfies by Ito lemma for a smooth u, $s \geq t$:

$$\begin{cases} du(s,p_s^{t,x,\alpha}) &= [\frac{\partial u}{\partial t} + \frac{\alpha_s^2}{2}\frac{\partial^2 u}{\partial x^2}](s,p_s^{t,x,\alpha})ds + \alpha_s \frac{\partial u}{\partial x}(s,p_s^{t,x,\alpha})dW_s, \\ dP_s^{t,x,\alpha} &= \alpha_s dW_s \\ P_t^{t,x,\alpha} &= x \end{cases}$$

The HJB equation is obtained by imposing that the variation of the wealth is equal to the variation of $u(s,p_s^{t,x,\alpha})$:

$$\inf_{\alpha \frac{\partial u}{\partial x} = \kappa\theta u} [\frac{\partial u}{\partial t}(t,x) + \frac{\alpha^2}{2}\frac{\partial^2 u}{\partial x^2}(t,x) - \mu\theta u(t,x)] = 0 \text{ for}(t,x) \in [0,T]\times[0,1],$$

which is equivalent to

$$\inf_{\alpha\in\mathbf{R}} [\frac{\partial u}{\partial t}(t,x) + \frac{\alpha^2}{2}\frac{\partial^2 u}{\partial x^2}(t,x) - \frac{\alpha\mu}{\kappa}\frac{\partial u}{\partial x}] = 0 \text{ for}(t,x) \in [0,T]\times[0,1]. \quad (20)$$

The final condition is obviously given by $u(T,x) = x$. So setting $T = 1$, $\mu = 0.1$, $\kappa = 0.1$, u is the solution of equation (1) with :

$$f_a(t,x) = 0, \quad c_a(t,x) = 0, \quad b_a(t,x) = -\frac{a\mu}{\kappa}, \quad \sigma_a(t,x) = a, \quad A = \mathbf{R}$$
$$u(0,x) = x, Q = (0,1]\times[0,1]$$

After having defined the Hamilton Jacobi Bellman to solve, we explain how to get an analytical solution to the problem. Using the first order condition, the function u satisfies

$$\frac{\partial u}{\partial t}(t,x) - \frac{\mu^2}{2\kappa^2}\frac{(\frac{\partial u}{\partial x})^2}{\frac{\partial^2 u}{\partial x^2}} = 0$$

The Fenchel transform of u , $v(t,q) = \sup_{x\in[0,1]}\{xq - u(t,x)\}$ satisfies:

$$\begin{cases} \frac{\partial v}{\partial t}(t,q) + \frac{\mu^2}{2\kappa^2}\frac{\partial^2 v}{\partial^2 q}(t,q) = 0 \text{ for}(t,x) \in [0,T]\times\mathbf{R} \\ v(T,q) = (q-1)^+ \end{cases}$$

Using Feyman Kac, v is the price of an European call where the asset follows the dynamic:

$$dQ_t = \frac{\mu}{\kappa}Q_t dW_t$$

Using Black Scholes formulae and taking the dual of v (so the bi-dual of u), we get the analytical solution for u that is used to estimate the error obtained by solving equation (20):

$$u(t,x) = N(N^{-1}(x) + \frac{\mu}{\kappa}\sqrt{T-t}),$$
$$N(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-\frac{u^2}{2}}du$$

Numerically a is bounded so that the diffusion coefficients don't explode.

Table 5 Test case 4

LINEAR				CUBIC				MPCSL				TCHEB 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
200	0.0445		45	80	0.023		7	80	0.0234		8	20	0.0503		14
400	0.0249		81	160	0.0143		13	160	0.0143		12	40	0.032		27
800	0.014	0.83	157	320	0.00879	0.66	29	320	0.0088	0.72	28	80	0.020	0.61	46
1600	0.0078	0.81	307	640	0.00529	0.66	55	640	0.0052	0.62	52	160	0.0125	0.68	95
3200	0.004	0.71	612	1280	0.00314	0.70	113	1280	0.0031	0.77	108	320	0.0075	0.59	180
LEGEND 2				LEGEND 3				BERN 2				BERN 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
20	0.054		3	20	0.0520		7	100	0.0575		34	100	0.049		61
40	0.034		7	40	0.033		10	200	0.032		63	200	0.0281		112
80	0.021	0.62	13	80	0.020	0.55	18	400	0.0187	0.93	117	400	0.016	0.79	217
160	0.013	0.70	23	160	0.0127	0.85	34	800	0.0106	0.71	235	800	0.009	0.79	435
320	0.007	0.41	47	320	0.0077	0.55	70	1600	0.006	0.81	464	1600	0.0052	0.88	867

Remark 9 This truncation is necessary for the scheme and introduce an error. By using the first order optimal condition in equation (20), we see that the optimal control goes to infinity as t goes to T.

The solutions obtained by solving the equation (20) for the various schemes are given in Table (5). The controls are bounded to 16, the number of controls tested is equal to 8000 and the number of time steps is taken equal to 1600 so $h = 6.25e-4$. CPU times are given for 48 cores used. All the methods have similar convergence rate but for very coarse meshes high order schemes are far more effective. On the finer meshes used, LEGEND 2, CUBIC and MPCSL still give the best results considering the error versus the computing time.

5.3.5 A 2D dimensional control problem

We use here the stochastic target problem from [18]. Coefficients are given by:

$$f_a(t,x) = 0, c_a(t,x) = 0, b_a(t,x) = \begin{pmatrix} -\frac{\kappa^2}{2} \\ -\frac{\mu}{\kappa} a \end{pmatrix},$$

$$\sigma_a(t,x) = \begin{pmatrix} \kappa \\ a \end{pmatrix}, A = \{a \in \mathbf{R}\}, Q = (0,1] \times [-3,3] \times [0,1]$$

The initial condition for a European call with strike K is given by:

$$g(x_1,x_2) = x_2(S_0 e^{x_1} - K)^+$$

We take $\kappa = 0.4$, $\mu = 1$, $S_0 = K = 1$. The value function u is convex in x_2. Its Legendre-Fenchel transform u^* can be estimated by Monte Carlo method and we numerically calculate our reference solution $u = u^{**}$ with

$$u(t,x_1,x_2) = \max_q \left[x_2 q - \mathbb{E}\left[(q e^{-\frac{\mu^2}{2\kappa^2} + \frac{\mu}{\kappa} g} - (x_1 e^{-\frac{\kappa^2}{2} + \kappa g})^+)^+ \right] \right]$$

and $g \sim \mathbf{N}(0,1)$. Numerically we have to truncate the domain in x_1. The maximum control is truncated to 10 and discretized with 2000 values. The number of time steps is equal to 1600 So $h = 6.25e-4$. We give the error on a sub domain of the domain of resolution $[-1.6, 1.6] \times [0,1]$. CPU times are given for 192 cores. Similarly to the previous case, the higher order scheme are not superior to the LINEAR scheme in term of rate of convergence but for coarse meshes the higher order schemes are clearly superior.

Table 6 Test case 5

LINEAR				CUBIC				MPCSL				TCHEB 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
40	1.21		21	10	0.85		7	10	0.341		6	10	0.231		41
80	1.008		86	20	0.086		10	20	0.0746		10	20	0.077		127
160	0.619		285	40	0.045	4.23	33	40	0.118		33	40	0.224		394
320	0.319	0.36	1176	80	0.041	3.33	131	80	0.041		129	80	0.175		1577
640	0.192	1.24	4647	160	0.038	0.55	455	160	0.039	5.48	450	160	0.093	-0.72	6312
LEGEND 2				LEGEND 3				BERN 2				BERN 3			
NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time	NbM	Err	Rate	Time
10	0.032		6	10	0.035		30	40	0.417		266	40	0.395		916
20	0.017		21	20	0.019		87	80	0.2950		929	80	0.2647		3657
40	0.013	1.96	83	40	0.0155	2.12	270	160	0.1785	0.07	3711	160	0.1336	0.01	14623
80	0.014		293	80	0.0132	0.66	1077	320	0.069	0.08	14530				

6 Appendix

6.1 Some results on Lagrange interpolators

For a d dimensional grid $X = X_N^d$, the interpolation operator is the composition of interpolators $I_N^X(f)(x) = I_N^{X_N,1} \times I_N^{X_N,2} ... \times I_N^{X_N,d}(f)(x)$ where $I_N^{X_N,i}$ is the one dimensional interpolator in direction i.
If we divide the domain $I = [a_1, b_1] \times .. \times [a_d, b_d]$ in meshes $\Delta x = (\Delta x_1, \Delta x_2 .., \Delta x_d)$ such that

$$M_{i_1,..i_d} = [a_1 + i_1 \Delta x_1, a_1 + (i_1+1)\Delta x_1] \times .. \times [a_d + i_d \Delta x_d, a_d + (i_d+1)\Delta x_d]$$

and if a Lagrange interpolation is used on each mesh $M_{i_1,...,i_d}$ for the function $g(x) = f(a_1 + i_1 \Delta x_1 + (x_1+1)\frac{\Delta x_1}{2}, .., a_d + i_d \Delta x_d + (x_d+1)\frac{\Delta x_d}{2})$ then (see for example [22] page 270) for $f \in C^{k+1}(I)$,

$$||f - I_{N,\Delta x}^X f||_\infty \leq C(N) \sum_{i=1}^{d} \Delta x_i^{k+1} \sup_{x \in [-1,1]^d} |\frac{\partial^{k+1} f}{\partial x_i^{k+1}}| \tag{21}$$

When f is only K Lipschitz, using Jackson's theorem we get :

$$||I_N^X(f) - f||_\infty \leq CK \sup_i \Delta x_i \sqrt{d} \frac{(1+\lambda_N(X))^d}{N+2} \tag{22}$$

6.2 Some results on Bernstein polynomials

The approximation $B_N(f)$ of a function $f : [0,1] \longrightarrow \mathbf{R}$ is the polynomial

$$B_N(f)(x) = \sum_{i=0}^{N} f(\frac{i}{N}) P_{N,i}(x) \text{ where } P_{N,i}(x) = \binom{N}{i} x^i (1-x)^{N-1}.$$

It is important to notice that it is not an interpolation. Only points 0 and 1 are interpolated. By tensorization [23]

$$B_{N_1,..,N_d}(f)(x_1,..,x_d) = \sum_{i_1=0}^{N_1} ... \sum_{i_d=0}^{N_d} \left[\prod_{j=1}^{d} P_{N_j,i_j}(x_j) \right] f(\frac{i_1}{N_1}, ..., \frac{i_d}{N_d})$$

By introducing the modulus of continuity

$$w_1(f, \delta_1, .., \delta_d) = \sup\{|f(x_1, .., x_d) - f(y_1, .., y_d)|; |x_i - y_i| \leq \delta_i, i = 1, ..d\}$$

we have the following estimation [23]

$$|f(x_1, .., x_d) - B_{N_1,..,N_d}(f)(x_1, .., x_d)| \leq C w_1(f, \frac{1}{\sqrt{N_1}}, ..., \frac{1}{\sqrt{N_d}})$$

For a regular function the convergence rate is low

$$|f(x_1, .., x_d) - B_{N,..,N}(f)(x_1, .., x_d)| \leq \frac{C}{N} \sum_i^d |\frac{\partial^2 f}{\partial x_i^2}(x_1, .., x_d)|.$$

The weights associated to this approximation are positive and independent on the function so this operator is monotone. It is known that it preserves the convexity. Many other approximations with similar properties can be developed [24].

Acknowledgements Special thank to Nadia Oudjane and Stéphane Villeneuvre for their careful reading. Special thanks to Romuald Elie for providing an analytical solution to test case (5.3.4).

References

1. W.H. Fleming and H.M. Soner, *Controlled Markov Processes and Viscosity Solutions*, Springer, (2005)
2. J.F. Bonnans and H. Zidani, *A fast algorithm for the two dimensional HJB equation of stochastic control*, ESAIM:M2AN, 38-4 , (2004), pp. 723–735
3. G. Barles and P. E. Souganidis, *Convergence of Approximation Schemes for Fully Non-linear Second Order Equation* , Asymptotic Anal., 4, (1991), pp. 271–283
4. F. Camilli and M. Falcone , *An approximation scheme for the optimal control of diffusion processes*, Modélisation Mathématique et Analyse Numérique 29.1,(1995), pp. 97–122
5. R. Munos and H. Zidani, *Consistency of a simple multidimensional scheme for Hamilton-Jacobi-Bellman equations*, C. R. Acad. Sci. Paris, Ser. I Math, (2005)
6. K. Debrabant and E. R. Jakobsen., *Semi-Lagrangian schemes for linear and fully non-linear diffusion equations*, Math. Comp, no. 283 (2013), pp. 1433–1462
7. A. Fahim, N. Touzi. and X. Warin, *A Probabilistic Numerical Scheme for Fully Nonlinear PDEs*, Annals of Applied Probability 21, 4, (2011), pp. 1322–1364.
8. X. Tan , *A splitting method for fully nonlinear degenerate parabolic PDEs*, Electron. J. Probab. 18(15), (2013), pp. 1–24
9. N. V. Krylov, *On The Rate Of Convergence Of Finite-Difference Approximations For Bellman's Equations With Variable Coefficients*, Probab. Theory Relat. fields,117,(2000), pp. 1-16
10. G. Barles and P. E. Souganidis, *Convergence of Approximation Schemes for Fully Non-linear Second Order Equation* , Asymptotic Anal., 4, (1991), pp. 271-283
11. M. Falcone, R. Ferretti,*Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations*, Society for Industrial and Applied Mathematics, (2014)
12. M. Falcone, R. Ferreti, *Convergence Analysis for a Class of High-Order Semi-Lagrangian Advection Schemes*, SIAM J. Numer. Anal., 35(3), (1998), pp. 909-940
13. R. Ferretti, *A technique for high-order treatment of diffusion terms in semi-Lagrangian schemes*, Communications, 8(2), Global Science Press, (2010)

14. H. Ishii, P.L. Lions, *Viscosity solutions of fully nonlinear second-order elliptic partial differential equations*, Journal of differential equations,83(1,(1990), pp. 26-78
15. P. Forsyth and G. Labahn,*Numerical methods for controlled Hamilton-Jacobi-Bellman PDEs in finance*, Journal of Computational Finance, 11(2), (2007), pp. 1-44
16. S. C. Eisenstat, K. R. Jackson and J. W. Lewis, *The order of monotone piecewise cubic interpolation*, SIAM J. Numer. Anal., 22, (1985), pp. 1220–1237
17. L. M. Koci and G. V. Milovanovi, *Shape preserving approximations by polynomials and splines*, Comput. Math. Appl., 33, (1997), pp. 59–97
18. B. Bouchard, R. Elie and N. Touzi, *Stochastic Target Problems with Controlled Loss*, SIAM Journal on Control and Optimization, 48(5), (2009), pp. 3123–3150
19. K. Kunisch, L. Xie, *POD-based feedback control of the Burgers equation by solving the evolutionary HJB equation*, Computers and Mathematics with Appl., 49, (2005), pp. 1113-1126
20. C. Makassikis, S. Vialle and X. Warin, *Large Scale Distribution of Stochastic Control Algorithms for Financial Applications*, PDCoF08, pages 1-8, (2008-04)
21. C. Makassikis, P. Mercier, S. Vialle, and X. Warin, *Stochastic control optimization & simulation applied to energy management: From 1-D to N-D problem distributions, on clusters, supercomputers and Grids*, Grid@Mons conference, (2008)
22. A. Quarteroni, R. Sacco and F. Saleri, *Méthodes numriques*, Springer, (2007)
23. S. G. Gal, *Shape preserving by real and complex polynomials*, Birkhauser Verlag AG, (2008)
24. G. A. Anastassiou and S. G. Gal,*Approximation theory : Moduli of continuity and global smoothness preservation*, Birkhuser, (2000)