Boundary Mesh Refinement for Semi-Lagrangian Schemes

Athena Picarelli, Christoph Reisinger, Julen Rotaetxe Arto

Boundary Mesh Refinement for Semi-Lagrangian Schemes

Athena Picarelli*, Christoph Reisinger*, and Julen Rotaetxe Arto*

June 13, 2017

Abstract

We study semi-Lagrangian schemes for the Dirichlet problem for second-order degenerate elliptic PDEs. Like other wide stencil schemes, these schemes have to be truncated near the boundaries to avoid "over-stepping". The various modifications proposed in the literature lead to either reduced consistency orders for those points, or even a loss of consistency with the differential operator in the usual sense. We propose a local mesh refinement strategy near domain boundaries which achieves a uniform order of consistency up to the boundary in the first case, and in both cases reduces the width of the region where overstepping occurs, so that the practically observed convergence order is unaffected by overstepping. We demonstrate this numerically for a linear parabolic equation and a second order HJB equation.

1 Introduction

In this paper, we investigate the convergence behaviour of semi-Lagrangian schemes for second order degenerate elliptic equations on bounded domains.

For concreteness, we consider the Hamilton-Jacobi-Bellman (HJB) equation

$$u_t(t,x) + \sup_{\alpha\in\mathcal{A}} \{-L^\alpha[u](t,x) - c^\alpha(t,x)u(t,x) - f^\alpha(t,x)\} = 0, \quad (t,x) \in (0,T]\times\Omega, \tag{1}$$

$$u(0,x) = \psi(0,x), \quad x \in \bar{\Omega}, \tag{2}$$

$$u(t,x) = \psi(t,x), \quad (t,x) \in (0,T]\times\partial\Omega, \tag{3}$$

where $Q_T := (0,T]\times\bar{\Omega}$ with $\bar{\Omega} := \Omega\cup\partial\Omega \subseteq \mathbb{R}^d$, $\mathcal{A}$ is a compact set,

$$L^\alpha[u](t,x) = \mathrm{tr}[a^\alpha(t,x)D^2u(t,x)] + b^\alpha(t,x)Du(t,x) \tag{4}$$

is a second order differential operator, and the known function ψ contains the initial and spatial boundary values.

Linear parabolic equations are a special case where $|\mathcal{A}| = 1$, while fully-nonlinear equations of Isaacs-type are written as min max problems. The construction of the scheme in this paper is analogous in both of these cases.

*Mathematical Institute, University of Oxford, Andrew Wiles Building, Woodstock Rd, Oxford OX2 6GG, {athena.picarelli, christoph.reisinger, julen.rotaetxe}@maths.ox.ac.uk

The coefficients $a^\alpha = \frac{1}{2}\sigma^\alpha\sigma^{\alpha,T}$, b^α, c^α, f^α, and the data ψ in (1) take their values, respectively, in $\mathcal{S}^d$, the space of $d \times d$ symmetric matrices, $\mathbb{R}^d$, $\mathbb{R}$, $\mathbb{R}$, and $\mathbb{R}$, $\sigma^\alpha \in \mathbb{R}^{d\times P}$, such that a^α is positive semi-definite. We assume the usual well-posedness conditions, i.e. Lipschitz continuity of the coefficients in x uniformly in α, Hölder continuity with exponent $\frac{1}{2}$ in time and continuity in α for each $(t, x) \in Q_T$ [12]. This guarantees existence and uniqueness of the solution in the viscosity sense [7] as well as a comparison principle.

In this paper, we focus on semi-Lagrangian schemes for the approximation of (1)–(3) as introduced in [6, 14] and analysed more recently in [5, 8, 9, 11]. For convenience of the reader and to introduce the notation, we briefly describe the specific scheme used here.

Following loosely [8], we define a non-degenerate polyhedral coverage $\overline{\mathcal{C}} = \{C_j\}_{j\in\mathbb{J}}$ of $\bar{\Omega}$, where $\mathbb{J}$ is the index set of the cells. The elements of the set $\overline{\mathcal{C}}$ are chosen to satisfy

$$\begin{aligned} \operatorname{int}(C_j \cap C_i) &= \emptyset, \quad \forall\, i \neq j, \\ \bigcup_{j\in\mathbb{J}} C_j &\supseteq \bar{\Omega}, \\ \nu\Delta x \leq \sup_{j\in\mathbb{J}}\{\operatorname{diam}\mathcal{B}_{C_j}\} &\leq \sup_{j\in\mathbb{J}}\{\operatorname{diam} C_j\} \leq \Delta x, \end{aligned}$$

for some $\nu \in (0, 1)$, where 'int' and 'diam' are the interior and the diameter, and $\mathcal{B}_{C_j}$ is the greatest ball contained in C_j.

We define the subset of interior cells $\mathcal{C} = \{C \in \overline{\mathcal{C}} : C \subseteq \bar{\Omega}\}$ as the ones fully contained in the domain, and the nodes $\mathcal{N} := \{x_i\}_{i\in\mathbb{I}}$ as the set of all the vertices of elements in $\mathcal{C}$, where we denote by $\mathbb{I}$ the index set of the nodes with $N_x := |\mathcal{N}|$.

We assume non-negative basis functions $\{w_i(\cdot) : i \in \mathbb{I}\}$ associated with the mesh nodes, such that for any continuous function $\phi : \Omega \to \mathbb{R}$

$$[\mathcal{I}_{\Delta x}\phi](x) = \sum_{i\in\mathbb{I}} \phi(x_i)w_i(x), \tag{5}$$

for all $x \in \Omega$, $x_i \in \mathcal{N}$. For simplicity, we focus our attention occasionally on cuboid meshes and multilinear interpolants, defined by the standard piecewise multilinear basis. The interpolation error is then $\mathcal{O}\left(\Delta x^2\right)$ for sufficiently smooth functions, and this is the only property we will use for the consistency analysis. The non-negativity of the basis is required only for the monotonicity of the interpolation operation and subsequently for convergence of the scheme to the viscosity solution.

Writing $\sigma^\alpha = (\sigma_1^\alpha, \sigma_2^\alpha, \ldots, \sigma_P^\alpha) \in \mathbb{R}^{d\times P}$, where $\sigma_p^\alpha \in \mathbb{R}^d$ for $p \in \{1, 2, \ldots, P\}$ is the p-th column of σ^α, by the usual arguments,

$$\frac{1}{2}\operatorname{tr}\left[\sigma^\alpha\sigma^{\alpha,T}D^2\phi(x)\right] = \frac{1}{2}\sum_{p=1}^{P} \frac{\phi(x + k\sigma_p^\alpha) - 2\phi(x) + \phi(x - k\sigma_p^\alpha)}{k^2} + \mathcal{O}(k^2), \tag{6}$$

$$b^\alpha D\phi(x) = \frac{\phi(x + k^2b^\alpha) - \phi(x)}{k^2} + \mathcal{O}(k^2), \tag{7}$$

for $k > 0$ and any smooth function ϕ. For brevity we write $b^\alpha \equiv b^\alpha(t, x)$ and $\sigma^\alpha \equiv \sigma^\alpha(t, x)$. When these approximations are used for points in $\mathcal{N}$, the displaced points $x + k^2b^\alpha$ and $x \pm k\sigma_p^\alpha$ do not generally coincide with nodes in $\mathcal{N}$. Then, ϕ is approximated by its

interpolant $\mathcal{I}_{\Delta x}\phi$. In the case of linear basis functions, the resulting scheme is referred to as the Linear Interpolation Semi-Lagrangian (LISL) scheme.

We ignore for the time being the situation where $x + k^2 b^\alpha$ or $x \pm k\sigma_p^\alpha$ lies outside any of the mesh cells ("oversteps"), which of course will be the focus of the main body of the article.

The consistency error is easily seen to be (see [8])

$$\mathcal{O}\left(k^2 + \frac{\Delta x^2}{k^2}\right).$$

The first term is the consistency error for the finite difference approximation of the first and second order derivatives in (6) and (7), whereas the second term corresponds to the linear interpolation error when replacing ϕ by its interpolant in (6) and (7). The optimal choice $k = \sqrt{\Delta x}$ makes the consistency error proportional to Δx.

Following the notation in [8], the LISL approximations to (4) can be expressed as

$$L^\alpha_{\Delta x}[\mathcal{I}_{\Delta x}\phi](t,x) := \sum_{p=1}^{M} \frac{[\mathcal{I}_{\Delta x}\phi](t, x + y_p^{\alpha,+}(t,x)) - 2\phi(t,x) + [\mathcal{I}_{\Delta x}\phi](t, x + y_p^{\alpha,-}(t,x))}{2\Delta x}, \quad (8)$$

for $x \in \mathcal{N}$, and some $M \geq 1$. The functions $y_\cdot^{\alpha,\pm}(t,x)$ determine the stencil of the scheme at (t,x).

Different spatial schemes can be obtained depending on the values taken by M and $y_p^{\alpha,\pm}(t,x)$ in (8). In the following, we study specifically the approximation in [8] with

$$\begin{aligned} y_p^{\alpha,\pm} &= \pm\sqrt{\Delta x}\sigma_p^\alpha, && p \leq P, \\ y_{P+1}^{\alpha,\pm} &= \Delta x b^\alpha, && M = P+1. \end{aligned}$$

Other schemes are defined similarly. The scheme above has more flexibility in defining consistent boundary modifications, as explained in [16].

Finally, a fully discrete scheme is obtained by combining (8) with a time stepping scheme. We introduce a time mesh $\mathcal{T}_{\Delta t} = \{t_n : \ 0 \leq n \leq N_t\} \subseteq [0,T]$, for simplicity with $t_n = n\Delta t$, i.e. with uniform step size $\Delta t > 0$. We then define the standard θ-scheme by

$$\frac{u(t_n,x) - u(t_{n-1},x)}{\Delta t} - \inf_{\alpha\in\mathcal{A}} \{\theta L^\alpha_{\Delta x}[\mathcal{I}_{\Delta x}u](t_{n-1},x) + (1-\theta)L^\alpha_{\Delta x}[\mathcal{I}_{\Delta x}u](t_n,x)\} = 0, \quad (9)$$

where we have set $c = f = 0$ in (1) for simplicity. In the tests we will focus on $\theta = 0$ (explicit Euler) and $\theta = 1$ (implicit Euler). Although the scheme can in principle be defined in this way for all $x \in \Omega$, we restrict the scheme to $x \in \mathcal{N}$.

2 Domain overstepping and stencil truncation

We now focus on the boundary of the domain and the definition of the scheme there. We take $\Omega \subset \mathbb{R}^d$ for $d \geq 1$ a general domain with curved boundary, but for simplicity illustrate the method for $\overline{\mathcal{C}}$ defined by a Cartesian mesh on $\mathbb{R}^d$ with uniform mesh width Δx. Then we choose $\mathcal{C}$ and $\mathcal{N}$ as explained in Section 1. See Figure 1.

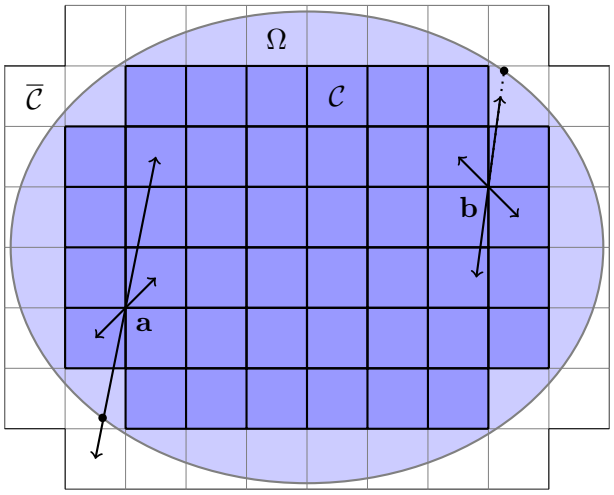

Figure 1: An elliptical domain and a mesh made of square cells. In situation **a**, $x + y_p^{\alpha,\pm}(t,x) \notin \bar{\Omega}$, while in situation **b**, $x + y_p^{\alpha,\pm}(t,x) \in \bar{\Omega}$, but the cell it is contained in has vertices outside $\bar{\Omega}$. The modified stencil in Sections 2.2 –2.4 uses values from the boundary.

2.1 Overstepping

Now consider a mesh node $x \in \mathcal{N}$ in the vicinity of the boundary. In the two situations sketched in Fig. 1 the interpolation at the point $x + y_p^{\alpha,\pm}(t,x)$ fails for given t, α and p. We say in these cases that the stencil "oversteps". We now discuss several possibilities to deal with these situations. They are all based on a local modification of the step $y_p^{\alpha,\pm}(t,x)$ and of k, combined in some cases with a modification of the interpolation weights.

Where the stencil oversteps, we define

$$\hat{y}_p^{\alpha,\pm}(t,x) = \mu_p^{\alpha,\pm}(t,x) y_p^{\alpha,\pm}(t,x),$$

where

$$\mu_p^{\alpha,\pm}(t,x) = \min\left\{\mu \geq 0 \,:\, x + \mu y_p^{\alpha,\pm}(t,x) \in \partial\Omega\right\}. \tag{10}$$

In case **a** in Fig. 1 this means $\mu < 1$, while in case **b** we have $\mu > 1$.

Remark 2.1. On rectangular domains, the elements of the Cartesian mesh cover exactly the domain and case **b** does not occur. Moreover, interior mesh points cannot be arbitrarily close to the boundary, but are always at least Δx away. This can be enforced in the general case by removing the outermost layer of cells in $\mathcal{C}$, such that again a distance of Δx between non-boundary mesh points and the domain boundary is ensured. This allows the derivation of CFL conditions for the explicit schemes as given below (Proposition 2.2 for the scheme in Section 2.4 and similar for other schemes).

2.2 Constant extrapolation

When the stencil oversteps, constant extrapolation simply uses instead the value at the boundary in the direction of the stencil step. This can be written as

$$\bar{L}^{\alpha}_{\Delta x}[\mathcal{I}_{\Delta x}\phi](t,x) := \sum_{p=1}^{M} \frac{[\mathcal{I}_{\Delta x}\phi](t, x+\hat{y}_p^{\alpha,+}(t,x)) - 2\phi(t,x) + [\mathcal{I}_{\Delta x}\phi](t, x+\hat{y}_p^{\alpha,-}(t,x))}{2\Delta x}. \tag{11}$$

Note that the scheme is generally not consistent up to the boundary. A proof of convergence is not available to the best of our knowledge.

2.3 Stencil cropping

The scheme in [10] shrinks the stencil so that it does not overstep. This corresponds to

$$\tilde{L}^{\alpha}_{\Delta x}[\mathcal{I}_{\Delta x}\phi](t,x) := \sum_{p=1}^{M} \frac{[\mathcal{I}_{\Delta x}\phi](t, x+\tilde{y}_p^{\alpha,+}(t,x)) - 2\phi(t,x) + [\mathcal{I}_{\Delta x}\phi](t, x+\tilde{y}_p^{\alpha,-}(t,x))}{2\tilde{k}^2}, \tag{12}$$

where

$$\tilde{y}_p^{\alpha,\pm} = \pm\tilde{k}\sigma_p^{\alpha} \quad \forall p = 1\dots P, \qquad \tilde{y}_{P+1}^{\alpha,\pm} = \tilde{k}^2 b^{\alpha}$$

and $0 < \tilde{k} \equiv \tilde{k}(t,x,\alpha) = \theta\Delta x$ for $\theta \in (0,1]$ is the biggest step such that $x + \tilde{y}_p^{\alpha,\pm} \in \bar{\Omega}, \forall p = 1\dots M$.

Generally this means the scheme is not consistent up to the boundary in the sense of [3], however, [10] can still prove a generalised consistency condition and convergence for viscosity solutions on convex domains.

2.4 Stencil truncation

In contrast to the above two schemes, the one in [16] is designed to be consistent in the whole of the domain. Hence, the objective is to find truncated or extended stencil vectors $\hat{y}_p^{\alpha,\pm}(t,x)$ and corresponding finite difference weights $A_p^{\alpha} \equiv A_p^{\alpha}(t,x)$ and $B_p^{\alpha} \equiv B_p^{\alpha}(t,x)$, such that $x + \hat{y}_p^{\alpha,\pm}(t,x) \in \partial\Omega$ and the truncated scheme

$$\hat{L}^{\alpha}_{\Delta x}[\mathcal{I}_{\Delta x}\phi](t,x) := \sum_{p=1}^{M} \frac{A_p^{\alpha}[\mathcal{I}_{\Delta x}\phi](t, x+\hat{y}_p^{\alpha,+}(t,x)) - (A_p^{\alpha}+B_p^{\alpha})\phi(t,x) + B_p^{\alpha}[\mathcal{I}_{\Delta x}\phi](t, x+\hat{y}_p^{\alpha,-}(t,x))}{2\Delta x} \tag{13}$$

is a consistent approximation of (4) as $\Delta x \to 0$. If the stencil does not overstep, we have that $\hat{y}_p^{\alpha,\pm}(t,x) = y_p^{\alpha,\pm}(t,x)$ and $A_p^{\alpha} = B_p^{\alpha} = 1$.

For all $p \in [[1, P+1]]$, let $\mu_p^{\alpha,\pm} \in [0,1]$ be as in 10, then define

$$A_{P+1}^{\alpha} = B_{P+1}^{\alpha} = \frac{1}{\mu_{P+1}^{\alpha,+}} \left(= \frac{1}{\mu_{P+1}^{\alpha,-}} \right) \tag{14}$$

and, for $p \in [[1, P]]$,

$$A_p^\alpha = \frac{2}{(\mu_p^{\alpha,+})^2 + \mu_p^{\alpha,+}\mu_p^{\alpha,-}}, \qquad B_p^\alpha = \frac{2}{(\mu_p^{\alpha,-})^2 + \mu_p^{\alpha,-}\mu_p^{\alpha,+}}. \tag{15}$$

Proposition 2.1 (Consistency, see Corollary 2.3 in [16])**.** *For the truncated scheme, (13), (14) and (15), the following holds: The local consistency error for points with truncation and $p \neq P+1$ is $\mathcal{O}(\sqrt{\Delta x})$ if only one side of the stencil oversteps, and $\mathcal{O}(1)$ if both sides overstep.*

Remark 2.2. The scheme can be made consistent (of order $\mathcal{O}(\Delta x)$) for the case where both sides of the stencil overstep by using the exact boundary value in the truncated scheme (13) instead of the interpolant. We therefore assume in the following that this is done.

Proposition 2.2 (Monotonicity and stability, see Corollary 2.5 in [16])**.** *In the case of overstepping and $\theta < 1$ in (9), monotonicity requires that $\Delta t \leq C_1 \Delta x^{3/2}$ if only one side of the diffusion stencils oversteps, or $\Delta t \leq C_2 \Delta x^2$ if both sides overstep. However, if the stencil is not truncated, the positivity condition remains as in [8], that is $\Delta t \leq C_3 \Delta x$, where $C_1, C_2, C_3 > 0$ are sufficiently small constants depending on the coefficients σ^α, b^α and c^α, but independent of Δx and Δt.*

As the scheme is consistent up to the boundary in the classical sense, convergence follows directly from the framework in [3].

3 Local mesh refinement

We now consider a refinement of the mesh in boundary layers of width $\mathcal{O}(\sqrt{\Delta x})$ where the semi-Lagrangian scheme oversteps. The objective is to improve the local consistency error of the truncated stencil in Section 2.4 from $\mathcal{O}(\sqrt{\Delta x})$ to $\mathcal{O}(\Delta x)$, and for all schemes to reduce the width of the region where the stencil oversteps from $\mathcal{O}(\sqrt{\Delta x})$ to $\mathcal{O}(\Delta x)$.

For this purpose, we combine a local refinement of the mesh with appropriate changes to the stencil step k in (6) in the region near the boundary.

3.1 General mesh construction

Let us consider a mesh defined by cells $\mathcal{C}$ and N_x nodes $\mathcal{N} \subset \Omega$ with refinement parameter $\Delta x > 0$, as in Section 1. For simplicity, we assume that $\sigma^\alpha(t, x) = \sigma(x)$.

Since it is primarily the overstepping of the diffusion stencil that reduces the local truncation error, we split $\mathcal{N}$ and $\mathcal{C}$ into three subsets. We will subsequently define new step sizes k_i for nodes in $\mathcal{N}_i$, and a refinement with mesh size h_i for the cells in $\mathcal{C}_i$.

Definition 3.1. Define

$$\begin{aligned}
\mathcal{C}_1 &= \{C \in \overline{\mathcal{C}} : \exists x \in C, \theta \in [0,1], 1 \leq p \leq P : x + \theta\sqrt{\Delta x}\sigma_p(x) \notin \overline{\Omega} \text{ or } x - \theta\sqrt{\Delta x}\sigma_p(x) \notin \overline{\Omega}\}^* \\
\mathcal{C}_2 &= \{C \in \overline{\mathcal{C}} \backslash \mathcal{C}_1 : \exists x \in \mathcal{C}_1, 1 \leq p \leq P : x + k_1\sigma_p(x) \in C \text{ or } x - k_1\sigma_p(x) \in C\} \\
\mathcal{C}_3 &= \overline{\mathcal{C}} \backslash (\mathcal{C}_1 \cup \mathcal{C}_2).
\end{aligned}$$

Then define $\mathcal{N}_1 = \mathcal{C}_1 \cap \mathcal{N}$ the nodes of elements in $\mathcal{C}_1$, $\mathcal{N}_2 = (\mathcal{C}_2 \cap \mathcal{N}) \backslash \mathcal{N}_1$, the nodes of elements in $\mathcal{C}_2$ which are not already in $\mathcal{N}_1$, and $\mathcal{N}_3 = \mathcal{N} \backslash (\mathcal{N}_1 \cup \mathcal{N}_2)$.

Therefore, $\overline{\mathcal{C}} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3$ with $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for $i \neq j$ and $i, j \in \{1, 2, 3\}$, and similarly for the nodes. This ensures that if $k_1, k_2 \leq k$, all nodes that overstep are contained in $\mathcal{N}_1$, and that no node in $\mathcal{C}_1$ steps into $\mathcal{C}_3$ after refinement.

We emphasise the need for the three region construction, with an overlapping layer $\mathcal{C}_2$ of a fine mesh (as in $\mathcal{C}_1$) and wide stencil (as in $\mathcal{C}_3$). This prevents the situation where a narrow stencil steps into the coarse mesh, which would spoil consistency.

Next, we refine the mesh elements in $\mathcal{C}_1$ and $\mathcal{C}_2$ with mesh-refinement parameter proportional to $h_1 = h_2 \sim \Delta x^\gamma$, where $\gamma > 1$ is a parameter to be determined so that the resulting local consistency error is at least $\mathcal{O}(\Delta x)$. After refinement, we remove the cells which are not fully in $\overline{\Omega}$ to create a new set of cells $\mathcal{C}'$ with vertices $\mathcal{N}'$. By refining $\overline{\mathcal{C}}$ and then pruning the refined cells outside $\overline{\Omega}$, the space between $\mathcal{C}$ and $\partial\Omega$ gets refined, which would not happen if we simply refined $\mathcal{C}$.

To determine γ, we procede in reverse order starting with $\mathcal{C}_3$. As the stencil may step into the finer regions $\mathcal{C}_1$ and $\mathcal{C}_2$, the consistency error there is

$$\mathcal{O}\left(k_3^2 + \frac{h_1^2}{k_3^2} + \frac{h_2^2}{k_3^2} + \frac{h_3^2}{k_3^2}\right) = \mathcal{O}\left(k_3^2 + \frac{\Delta x^2}{k_3^2} + \frac{\Delta x^{2\gamma}}{k_3^2}\right),$$

where the first term corresponds to the consistency error of the finite difference approximation of the second order derivative and the last terms to the interpolation error in the original and the refined regions respectively. As $\gamma \geq 1$, we do not modify the stencil step, i.e. $k_3 \sim \mathcal{O}(\sqrt{\Delta x})$. Similarly, the local consistency error for nodes $\mathcal{N}_2$ after refinement is

$$\mathcal{O}\left(k_2^2 + \frac{h_1^2}{k_2^2} + \frac{h_2^2}{k_2^2} + \frac{h_3^2}{k_2^2}\right) = \mathcal{O}\left(k_2^2 + \frac{\Delta x^2}{k_2^2} + \frac{\Delta x^{2\gamma}}{k_2^2}\right).$$

Choosing $k_2 \sim \mathcal{O}(\sqrt{\Delta x})$ the local consistency error is $\mathcal{O}(\Delta x)$, for all $\gamma \geq 1$.

Finally, the local consistency error for points in $\mathcal{C}_1$ after refinement is

$$\mathcal{O}\left(k_1 + \frac{h_1^2}{k_1^2} + \frac{h_2^2}{k_1^2}\right) = \mathcal{O}\left(k_1 + \frac{\Delta x^{2\gamma}}{k_1^2}\right),$$

where we have assumed that the exact boundary value is used in the case of stencil truncation (see Remark 2.2). Choosing $\gamma = \frac{3}{2}$ and $k_1 \sim \mathcal{O}(\Delta x)$ the truncation error is $\mathcal{O}(\Delta x)$.

Figure 2 shows a locally refined mesh and describes the effects of the refinement.

Figure 2 gives a distorted view for illustration purposes, where in reality $\mathcal{C}_2$ is substantially smaller than $\mathcal{C}_1$, and $\mathcal{C}_1$ is substantially smaller than $\mathcal{C}_3$.

Remark 3.1. Figure 2 shows that the local refinement leaves 'hanging nodes' at the interface between $\mathcal{C}_2$ and $\mathcal{C}_3$. These nodes do not pose a problem for semi-Lagrangian discretizations. For the interpolation, the 'hanging nodes' are not used for stencil points with any neighbours belonging to $\mathcal{C}_3$.

*Extrapolation into $\overline{\Omega} \backslash \mathcal{C}$ is not problematic as the distance between $\mathcal{C}$ and $\partial\Omega$ is $\mathcal{O}(\Delta x)$.

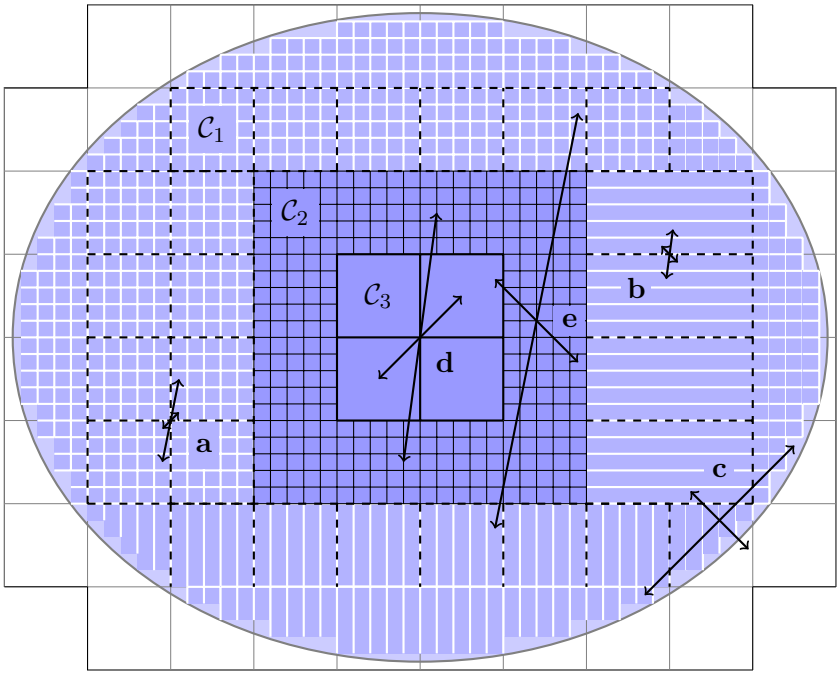

Figure 2: Locally refined mesh. The regions in Definition 3.1 are shown in different styles: coarse black mesh for $\mathcal{C}_1$, fine black mesh for $\mathcal{C}_2$ and fine white mesh for $\mathcal{C}_3$. Shown dashed is also the original coarse mesh. The stencil is shrunk for the nodes **a** and **b** from Figure 1 earlier, which both lie in $\mathcal{C}_3$, and does not overstep anymore. The stencil for the new node **c** oversteps in both ways that **a** and **b** did before refinement, but by a lesser distance. The stencil is not shrunk for nodes in $\mathcal{C}_1$ and $\mathcal{C}_2$. Point **d** illustrates the situation where the larger stencil steps from region $\mathcal{C}_3$ into the finer region $\mathcal{C}_2$, which can only improve the interpolation error. In **e**, the larger stencil steps from region $\mathcal{C}_2$ into both the fine region $\mathcal{C}_1$ and the coarse region $\mathcal{C}_3$. The latter highlights the importance of the three-region construction, which guarantees that no fine stencil steps into the coarse mesh. This would make the scheme inconsistent for those points.

3.2 Mesh construction in one dimension

In our numerical tests we focus on one-dimensional examples, i.e. the case $\Omega = (x_{\min}, x_{\max}) \subset \mathbb{R}$. The construction of the refined mesh is simpler in this case and we report here for completeness its main steps as implemented in the code used in Section 4.

Let

$$\Delta_{\max} = \sqrt{\Delta x}\|\sigma\|_\infty,$$

where $\|\sigma\|_\infty \equiv \sup_{(t,x,\alpha)\in[0,T]\times\Omega\times\mathcal{A}} |\sigma^\alpha(t,x)|$, denote the maximum step of the SL scheme subject to the volatility σ. Defining a global maximum step is not strictly necessary and it is only done to make the a priori definition of the sets $\mathcal{C}_i$, $i = 1, 2, 3$ easier. The width of $\mathcal{C}_2$ especially is not optimal here, but this does not affect the complexity significantly. Define

$$x_{i_1} = \min\{x_i \in \mathcal{N} : x_{\min} + \Delta_{\max} \le x_i\}, \qquad x_{i_2} = \min\{x_i \in \mathcal{N} : x_{i_1} + \Delta_{\max} \le x_i\}$$
$$x_{i_3} = \max\{x_i \in \mathcal{N} : x_{\max} - \Delta_{\max} \ge x_i\}, \qquad x_{i_4} = \max\{x_i \in \mathcal{N} : x_{i_3} - \Delta_{\max} \ge x_i\}.$$

Then, the sets $\mathcal{C}_i$ are defined as follows:

$$\mathcal{C}_1 = [x_{\min}, x_{i_1}] \cup [x_{i_3}, x_{\max}], \quad \mathcal{C}_2 = (x_{i_1}, x_{i_2}] \cup [x_{i_4}, x_{i_3}), \quad \mathcal{C}_3 = (x_{i_2}, x_{i_4}).$$

At this point a refinement of the mesh proportional to $\Delta x^{3/2}$ is considered in $\mathcal{C}_1 \cup \mathcal{C}_2$ (see Figure 3) defining the new mesh $\mathcal{N}'$.

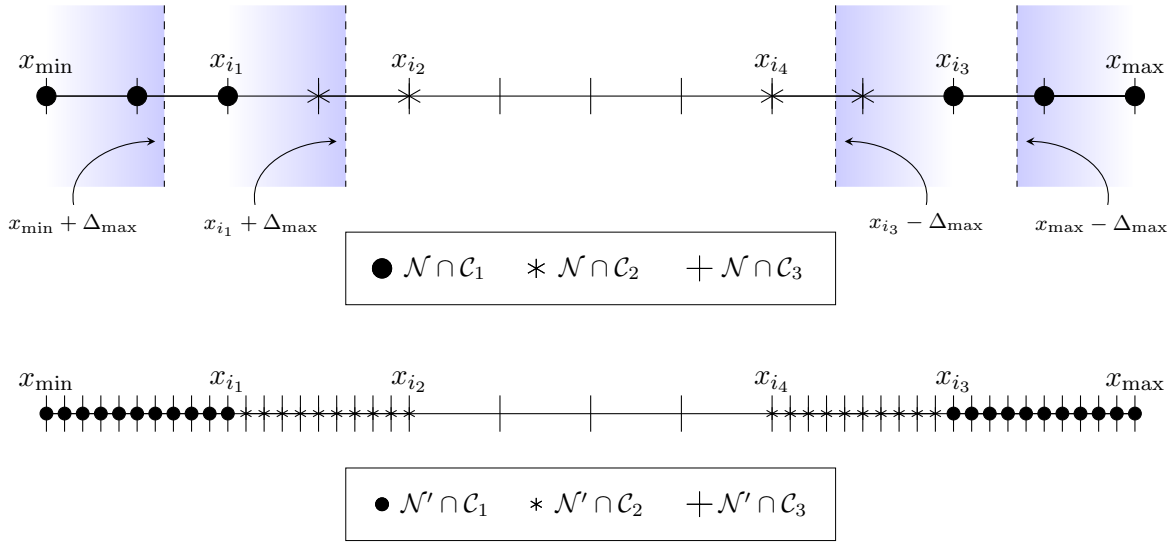

Figure 3: Mesh refinement in one dimension. Definition of sets $\mathcal{C}_i$, $i = 1, 2, 3$ (top) and refined mesh (bottom).

Given that the width of the stencil is of length $\mathcal{O}(\sqrt{\Delta x})$, the cardinality of $\mathcal{N}_1 \cup \mathcal{N}_2$ is $|\mathcal{N}_1 \cap \mathcal{N}_2| \sim \mathcal{O}(\sqrt{N_x})$. Moreover, after the refinement, the number of nodes in this region of width $\mathcal{O}(\sqrt{\Delta x})$ is $\mathcal{O}(N_x)$.

3.3 Properties of the refined scheme

Let $\Delta t, \Delta x > 0$ be the time and space mesh refinement parameters and $\mathcal{T}_{\Delta t} \subseteq [0,T]$ and $\mathcal{C} \subseteq \Omega \subseteq \overline{\mathcal{C}} \subset \mathbb{R}^d$ as above, with $N_t = |\mathcal{T}_{\Delta t}| - 1$ and $N_x = |\mathcal{N}| = \mathcal{O}(\Delta x^{-d})$.

The cardinality of the refined mesh $\mathcal{N}'$ in d dimensions is still $\mathcal{O}(N_x) = \mathcal{O}(\Delta x^{-d})$. Moreover, the analysis in Section 3.1 gives the following result.

Proposition 3.1 (Complexity and consistency)**.** *If we define sets of mesh cells according to Definition 3.1 and further refine the ones in $\mathcal{C}_1 \cup \mathcal{C}_2$ with mesh refinement parameter $\mathcal{O}(\Delta x^{3/2})$, the complexity of the method is $\mathcal{O}(N_t N_x) = \mathcal{O}(\Delta t^{-1} \Delta x^{-d})$. If for the nodes requiring truncation in $\mathcal{N}_1$ we use $k_1 \sim \mathcal{O}(\Delta x)$, then, globally the consistency error of this modified scheme becomes*

$$\frac{|1-2\theta|}{2}|\phi_{tt}|_0 \Delta t + C\left(\Delta t^2 |\phi_{ttt}|_0 + \Delta x(|D^2\phi|_0 + |D^3\phi|_0 + |D^4\phi|_0)\right). \tag{16}$$

As shown in Corollary 2.2, an undesirable side-effect of stencil truncation is the worsening of the CFL condition of the scheme. The local refinement of the grid results in a stricter CFL condition compared to the one in Corollary 2.2, as shown in the following.

Proposition 3.2 (Monotonicity and stability)**.** *Additionally, provided that all the nodes are $\mathcal{O}(\Delta x^{3/2})$ away from the boundary of the domain, a scheme with $\theta < 1$ requires $\Delta t \sim \mathcal{O}(\Delta x^{5/2})$ if only one side of the stencil oversteps or $\Delta t \sim \mathcal{O}(\Delta x^3)$ if both sides of the stencil overstep.*

Remark 3.2. Similar to Remark 2.1, ensuring that all nodes in the mesh are at least $\mathcal{O}(\Delta x^{3/2})$ away from the boundary of the domain can be achieved by removing the outermost layer of the cells inside the domain after refinement.

4 Numerical tests

We consider the HJB equation in a bounded set $\Omega := (x_{\min}, x_{\max})$. Denoted by N_x the number of mesh points, in one dimension,

$$\Delta x = \frac{x_{\max} - x_{\min}}{N_x - 1}.$$

Let (x_j) be a uniform mesh on $[x_{\min}, x_{\max}]$ with $x_j = x_{\min} + j\Delta x$, $j = 1, \dots, N_x$ and $t_n = n\Delta t$, $\Delta t = \frac{T}{N_t}$.
We are going to test both the explicit and the implicit scheme ($\theta = 0$ and $\theta = 1$, respectively, in (9)).

4.1 Test 1: Linear equation

We first test the scheme on a linear second order equation since, already in this simplified case, the main features of the scheme can be observed. In particular, we consider a classical Black-Scholes equation in Ω with a smooth initial datum:

$$u_t - \tfrac{1}{2}(\sigma x)^2 u_{xx} - (bx)u_x = f(t,x), \qquad t \in (0,T),\ x \in \Omega, \tag{17}$$

$$u(t,x) = v(t,x), \qquad t \in (0,T), x \in \partial\Omega, \tag{18}$$

$$u(0,x) = \sin(\pi x), \qquad x \in \Omega, \tag{19}$$

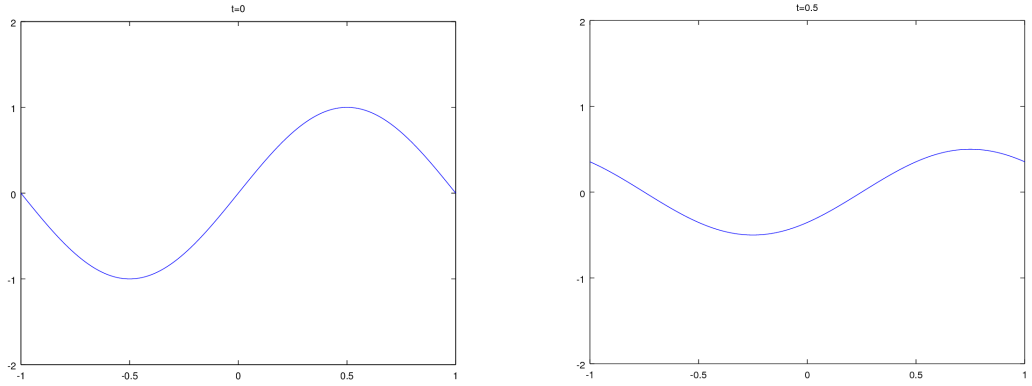

Figure 4: (Test 1) Plot of the value function at time $t = 0$ (left) and $t = T$ (right).

with

$$f(t,x) = \left(\frac{1}{2}(1-t)(\pi\sigma x)^2 - 1\right)\sin\left(\pi\left(x - \frac{t}{2}\right)\right) - (1-t)\left(\frac{\pi}{2} + \pi b x\right)\cos\left(\pi\left(x - \frac{t}{2}\right)\right).$$

The equation has the exact solution:

$$v(t,x) = (1-t)\sin\left(\pi\left(x - \frac{t}{2}\right)\right).$$

$x_{\min}$	$x_{\max}$	b	σ	T
-1	1	2	1	0.5

Table 1: Parameters used in numerical experiments for Test 1.

N_x	$N_t = \frac{N_x-1}{2}$		$N_t = \lfloor (N_x-1)^{3/2} \rfloor$		$N_t = \lfloor \frac{(N_x-1)^{5/2}}{20} \rfloor$	
	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
11	3.37E-01	-	2.23E-01	-	2.40E-01	-
21	3.68E-01	-0.13	1.30E-01	0.78	1.30E-01	0.89
41	1.82E+04	-15.59	7.16E-02	0.86	7.03E-02	0.88
81	6.53E+16	-41.71	3.73E-02	0.94	3.66E-02	0.94
161	2.20E+48	-104.73	1.89E-02	0.98	1.86E-02	0.98
321	8.83E+123	-251.15	9.48E-03	1.00	9.37E-03	0.99
641	4.96E+299	-583.83	4.72E-03	1.01	4.68E-03	1.00
1281	0.00E+00	NaN	2.35E-03	1.01	2.34E-03	1.00

Table 2: (Test 1) Explicit scheme. Truncated stencil without mesh refinement.

For our numerical tests we used the parameters in Table 1. The initial condition and the solution at terminal time are shown in Figure 4. The necessity of the CFL condition for

N_x	$N_t = \frac{N_x-1}{2}$		$N_t = \lfloor (N_x-1)^{3/2} \rfloor$		$N_t = \lfloor \frac{(N_x-1)^{5/2}}{20} \rfloor$	
	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
11	3.91E+01	-	1.61E-01	-	1.77E-01	-
21	2.52E+08	-22.62	8.90E-02	0.86	8.90E-02	1.00
41	2.11E+27	-62.86	4.83E+63	-215.05	5.90E-02	0.59
81	2.55E+74	-156.40	0.00E+00	NaN	3.20E-02	0.88
161	4.57E+187	-376.22	0.00E+00	NaN	1.74E-02	0.88
321	0.00E+00	NaN	0.00E+00	NaN	9.01E-03	0.95
641	0.00E+00	NaN	0.00E+00	NaN	4.57E-03	0.98
1281	0.00E+00	NaN	0.00E+00	NaN	2.30E-03	0.99

Table 3: (Test 1) Explicit scheme. Truncated stencil with mesh refinement.

N_x	$N_t = \frac{N_x-1}{2}$		$N_t = \lfloor (N_x-1)^{3/2} \rfloor$		$N_t = \lfloor \frac{(N_x-1)^{5/2}}{20} \rfloor$	
	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
11	4.00E-01	-	1.41E-01	-	1.52E-01	-
21	1.05E-01	1.09	1.16E-01	0.28	1.16E-01	0.39
41	7.52E-02	0.48	8.80E-02	0.40	8.86E-02	0.39
81	6.31E-02	0.25	6.73E-02	0.39	6.75E-02	0.39
161	5.38E-02	0.23	5.61E-02	0.26	5.62E-02	0.27
321	4.33E-02	0.31	4.43E-02	0.34	4.43E-02	0.34
641	3.38E-02	0.36	3.42E-02	0.37	3.42E-02	0.37
1281	2.61E-02	0.38	2.62E-02	0.38	2.62E-02	0.38

Table 4: (Test 1) Explicit scheme. Constant extrapolation of boundary conditions without mesh refinement.

N_x	$N_t = \frac{N_x-1}{2}$		$N_t = \lfloor (N_x-1)^{3/2} \rfloor$		$N_t = \lfloor \frac{(N_x-1)^{5/2}}{20} \rfloor$	
	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
11	3.40E+02	-	1.27E-01	-	1.52E-01	-
21	3.11E+08	-19.80	6.12E-02	1.06	6.83E-02	1.16
41	2.12E+24	-52.60	4.75E-02	0.37	4.92E-02	0.47
81	2.48E+62	-126.46	3.61E+90	-305.22	2.62E-02	0.91
161	1.78E+150	-291.85	0.00E+00	NaN	1.35E-02	0.96
321	3.98E+284	-446.30	0.00E+00	NaN	6.80E-03	0.98
641	0.00E+00	NaN	0.00E+00	NaN	3.37E-03	1.01
1281	0.00E+00	NaN	0.00E+00	NaN	1.70E-03	0.99

Table 5: (Test 1) Explicit scheme. Constant extrapolation of boundary conditions with mesh refinement.

N_x	N_t	Truncated stencil				Constant extrapolation			
		No refinement		With refinement		No refinement		With refinement	
		L^∞-error	rate	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
21	2	2.15E-01	-	2.15E-01	-	2.15E-01	-	2.15E-01	-
41	4	1.18E-01	0.86	1.21E-01	0.83	1.17E-01	0.88	1.18E-01	0.87
81	8	6.38E-02	0.89	6.43E-02	0.91	8.57E-02	0.45	6.34E-02	0.90
161	16	3.30E-02	0.95	3.31E-02	0.96	6.34E-02	0.43	3.26E-02	0.96
321	32	1.68E-02	0.98	1.69E-02	0.97	4.95E-02	0.36	1.66E-02	0.97
641	64	8.43E-03	0.99	8.52E-03	0.99	3.69E-02	0.42	8.52E-03	0.96
1281	128	4.27E-03	0.98	4.23E-03	1.01	2.72E-02	0.44	4.21E-03	1.01
2561	256	2.13E-03	1.00	2.15E-03	0.97	2.03E-02	0.42	2.11E-03	1.00

Table 6: (Test 1) Implicit scheme.

the explicit schemes ($N_t \sim N_x^{3/2}$ for the truncated stencil scheme without mesh refinement, $N_t \sim N_x^{5/2}$ for the scheme with mesh refinement) is confirmed by the results in Tables 2 to 5. The mesh refinement has little impact on the performance of the scheme with truncated stencil (last column of Tables 2 and 3). Clear improvements can be observed if a constant extrapolation of the boundary conditions is performed outside the domain (Tables 4 and 5). It is worthwhile to recall that in this last case the scheme is not consistent and the convergence is numerically achieved because consistency is lost only for points within a distance of $\sqrt{\Delta x}$ of the boundary, and Δx after mesh refinement. Similar results are obtained for the implicit scheme (Table 6). Here, no CFL condition is required and this allows us to take Δt proportional to Δx, i.e., N_t proportional to N_x.

4.2 Test 2: Controlled drift-diffusion equation

The second test we propose is a HJB equation with coefficients independent of (t, x):

$$u_t + \sup_{a_1,a_2} \Big\{ -\tfrac{1}{2}a^2 u_{xx} - 2a u_x \Big\} = 0, \qquad t \in (0,T),\ x \in \Omega, \tag{20}$$

$$u(t,x) = \psi(t,x), \qquad t \in (0,T), x \in \partial\Omega, \tag{21}$$

$$u(0,x) = \psi(0,x), \qquad x \in \Omega, \tag{22}$$

where $\psi(0,\cdot)$ is defined by

$$\psi(0,x) = \begin{cases} 5x(1+x)^4 & \text{if } -1 < x \le 0 \\ 5x(1-x)^4 & \text{if } 0 \le x < 1 \\ 0 & \text{if } |x| \ge 1. \end{cases}$$

We use the parameters in Table 7. In order to compare our numerical results we use a reference numerical solution computed in a larger domain $\Omega_1 = (-5, 5)$ with zero boundary conditions, using a second order (in time and space) BDF scheme (see [4]) with $N_x = 5121 \times 5$ and $N_t = (N_x - 1)/10$. The boundary value $\psi(t, x)$ at $x = \pm 1$ is obtained using this reference numerical solution. The initial condition and the solution at terminal time are shown in Figure 5.

$x_{\min}$	$x_{\max}$	a_1	a_2	T
-1	1	0.2	0.6	0.25

Table 7: Parameters used in numerical experiments for Test 2.

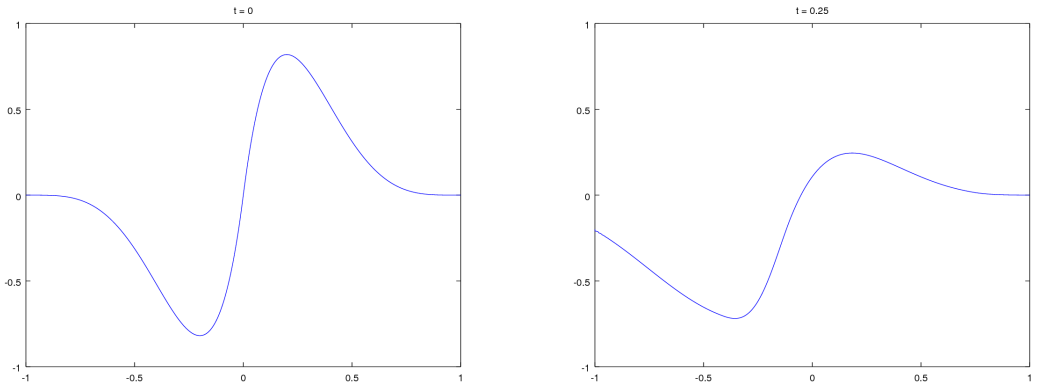

Figure 5: (Test 2) Plot of the value function at time $t = 0$ (left) and $t = T$ (right).

We report here, in Table 8, only the results obtained by the implicit scheme. However, under the CFL condition $N_t \sim N_x^{5/2}$, similar results can be obtained using the explicit scheme. The truncated scheme converges with order one (first two columns in Table 8). The scheme with constant extrapolation of boundary conditions converges with asymptotic order 1/2 (third column in Table 8) which becomes order 1 applying the mesh refinement (fourth column in Table 8). We point out that in this example the main contribution to the error comes from the points where the solution changes concavity (Figure 6), which correspond to discontinuities in the second order derivative. This explains the almost absence of differences in Table 8 (columns 1,2,4) and the fact that the order 1/2 in the third column is recovered after a quite big number of mesh refinements. Figure 6 compares the error obtained using the constant extrapolation of boundary conditions (left) and the truncated stencil (right). One can notice that the use of the constant extrapolation creates some instability at the left hand boundary, that is where such a scheme strongly modifies the nature of the exact solution outside the computational domain.

4.3 Test 3: Nonsmooth initial data

Let us consider the following problem:

$$u_t + \sup_{\sigma_{\min},\sigma_{\max}} \Big\{ -\tfrac{1}{2}(\sigma x)^2 u_{xx} \Big\} - bxu_x + ru = 0, \qquad t \in (0,T),\ x \in \Omega, \tag{23}$$

$$u(t,x) = 0, \qquad t \in (0,T), x \in \partial\Omega, \tag{24}$$

$$u(0,x) = \psi(0,x), \qquad x \in \Omega, \tag{25}$$

where $\psi(0,\cdot)$ is defined by

$$\psi(0,x) = \max\Big(-2K_1|x - K_2| + K_1, 0 \Big)$$

N_x	N_t	Truncated stencil				Constant extrapolation			
		No refinement		With refinement		No refinement		With refinement	
		L^∞-error	rate	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
21	2	2.61E-01	-	2.63E-01	-	2.61E-01	-	2.63E-01	-
41	4	1.75E-01	0.58	1.75E-01	0.58	1.75E-01	0.58	1.75E-01	0.58
81	8	1.05E-01	0.73	1.05E-01	0.73	1.05E-01	0.74	1.05E-01	0.73
161	16	5.62E-02	0.91	5.62E-02	0.91	5.61E-02	0.91	5.61E-02	0.91
321	32	2.88E-02	0.96	2.88E-02	0.96	2.87E-02	0.96	2.88E-02	0.96
641	64	1.41E-02	1.03	1.41E-02	1.03	1.40E-02	1.03	1.41E-02	1.03
1281	128	6.50E-03	1.11	6.50E-03	1.11	1.08E-02	0.38	6.50E-03	1.11
2561	256	3.45E-03	0.91	3.45E-03	0.91	8.58E-03	0.33	3.45E-03	0.91

Table 8: (Test 2) Implicit scheme.

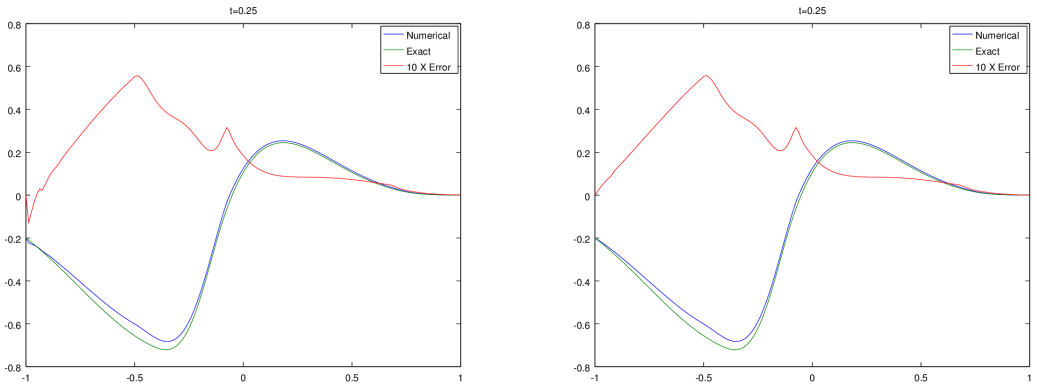

Figure 6: (Test 2) Exact solution (green), numerical solution (blue) and error (red) obtained with $N_x = 161$ and $N_t = 16$ using the constant extrapolation of boundary conditions (left) and the truncated stencil (right).

with $K_1, K_2 > 0$. This type of equations typically arises in financial applications when pricing options under uncertain volatility model, see [13]. The initial condition $\psi(0, \cdot)$ (Figure 7 (left)) has the shape of a butterfly payoff, for which a detailed study of finite difference numerical approximation has been proposed in [15].
In our numerical tests we use the parameters in Table 9. As in the previous example, in order to compute errors and rate of convergence, a reference numerical solution computed using a second order BDF scheme with $N_x = 5120$ and $N_t = (N_x - 1)/10$. The initial condition and the solution at terminal time are shown in Figure 7. The convergence orders are as previously.

Figure 8 compares the error obtained using the constant extrapolation of boundary conditions (left) and the truncated stencil (right). The instability at the boundary created by the constant extrapolation of the boundary conditions is even more evident in this example.

Last, we test the stencil cropping presented in [10] (see Section 2.3). In order to

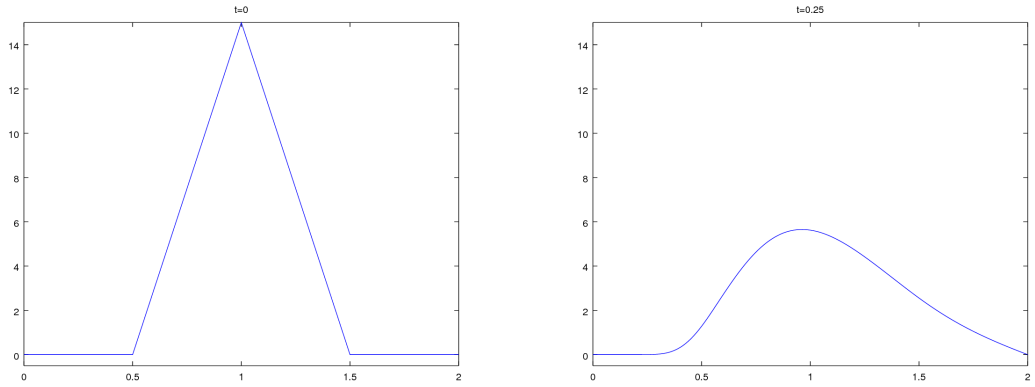

Figure 7: (Test 3) Plot of the value function at time $t = 0$ (left) and $t = T$ (right).

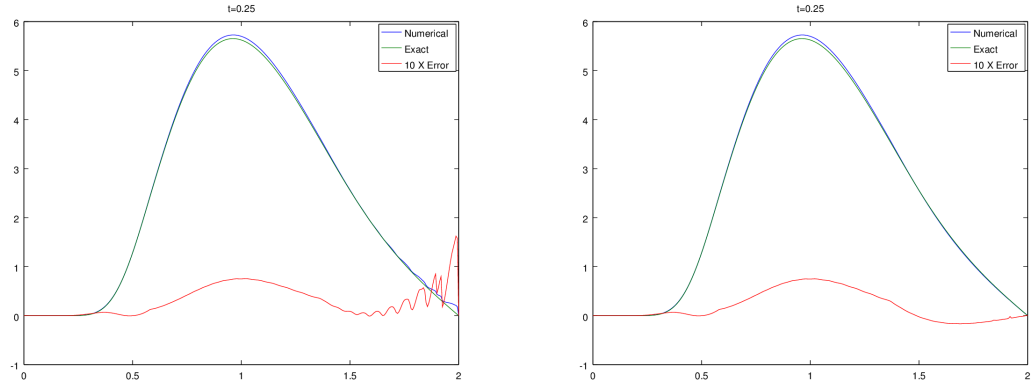

Figure 8: (Test 3) Exact solution (green), numerical solution (blue) and error (red) obtained with $N_x = 321$ and $N_t = 32$ using the constant extrapolation of boundary conditions (left) and the truncated stencil (right).

$x_{\min}$	$x_{\max}$	K_1	K_2	r	b	$\sigma_{\min}$	$\sigma_{\max}$	T
0	2	15	1	1	0.1	0.5	0.7	0.25

Table 9: Parameters used in numerical experiments for Test 3.

N_x	N_t	Truncated stencil				Constant extrapolation			
		No refinement		With refinement		No refinement		With refinement	
		L^∞-error	rate	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
21	2	1.20E+00	-	1.20E+00	-	1.20E+00	-	1.21E+00	-
41	4	5.71E-01	1.07	5.67E-01	1.08	5.74E-01	1.07	5.69E-01	1.09
81	8	2.91E-01	0.97	2.90E-01	0.97	2.93E-01	0.97	2.90E-01	0.97
161	16	1.48E-01	0.98	1.48E-01	0.97	2.03E-01	0.53	1.48E-01	0.97
321	32	7.51E-02	0.98	7.51E-02	0.98	1.62E-01	0.32	7.52E-02	0.98
641	64	3.77E-02	0.99	3.77E-02	0.99	1.26E-01	0.37	3.77E-02	0.99
1281	128	1.88E-02	1.00	1.88E-02	1.00	9.46E-02	0.41	1.88E-02	1.00
2561	256	9.41E-03	1.00	9.41E-03	1.00	6.88E-02	0.46	9.41E-03	1.00

Table 10: (Test 3) Implicit scheme.

avoid the case where the cropped stencil falls exactly on mesh points (this would make the scheme consistent since the interpolation would not be performed) we progressively divide the stencil by π until it fits into the domain $\overline{\Omega}$. The scheme shows a first order of convergence even without mesh refinement (Table 11). We observed the same also for the other examples in this section (not reproduced here). However, since consistency is not satisfied for those points where the cropping is performed, we cannot expect this to hold in general and the observed behavior in our tests might also be related to the fact that our solutions are almost linear close to the boundary, which makes the contribution to the error coming from the interpolation negligible.

N_x	N_t	No refinement		With refinement	
		L^∞-error	rate	L^∞-error	rate
21	2	1.19E+00	-	1.19E+00	-
41	4	5.72E-01	1.06	5.68E-01	1.07
81	8	2.91E-01	0.97	2.90E-01	0.97
161	16	1.48E-01	0.98	1.48E-01	0.97
321	32	7.52E-02	0.98	7.51E-02	0.98
641	64	3.77E-02	0.99	3.77E-02	0.99
1281	128	1.88E-02	1.00	1.88E-02	1.00
2561	256	9.41E-03	1.00	9.41E-03	1.00

Table 11: (Test 3) Implicit scheme. 'Cropped' stencil with and without mesh refinement.

5 Conclusions

In this paper we have presented and discussed a local mesh refinement for SL schemes approximating viscosity solutions to second order HJB equations in bounded domains with Dirichlet boundary conditions. In order to test our mesh refinement, we have considered different treatments for the 'overstepping' phenomena, which typically arise when this kind of wide stencil schemes is used. When a constant extrapolation of boundary conditions (Section 2.2) is applied, the local refinement of the mesh improves the observed order of convergence from order 1/2 to order 1. The scheme remains non-consistent with the differential operator in the neighborhood of the boundary and the mesh refinement has only the role of reducing the region of non-consistency. For the scheme with cropped stencil described in [10] (see also Section 2.3) the tests we performed do not show any benefits coming from the local mesh refinement since the scheme alone, even if not consistent up to the boundary, already shows a first order of convergence. The effects of the mesh refinement on the scheme with truncated stencil defined in [16] (also Section 2.4) consist in the improvement of the global truncation error from order 1/2 to order 1. Among other things, this permits to estimate the order of convergence of the scheme using the techniques in [1, 2, 8]. However, from the numerical point of view, no substancial difference in the rate of convergence of the scheme is observable, due to the fact that the truncation error of order $\sqrt{\Delta x}$ of the scheme without mesh refinement only occurs in a region of diameter $\sqrt{\Delta x}$.

In the numerical tests presented in Section 4 only the one-dimensional case was taken into account. Already in this simple setting the main feature of the scheme and the effects of the mesh refinement can be clearly observed. Numerical experiments in more dimensions and a deeper understanding of the numerical instabilities observed when a constant extrapolation of the boundary conditions is applied are the subject of future research.

References

[1] G. Barles and E.R. Jakobsen. Error bounds for monotone approximation schemes for Hamilton–Jacobi–Bellman equations. *SIAM Journal on Numerical Analysis*, 43(2):540–558, 2005.

[2] G. Barles and E.R. Jakobsen. Error bounds for monotone approximation schemes for parabolic Hamilton–Jacobi–Bellman equations. *Mathematics of Computation*, 76(260):1861–1893, 2007.

[3] G. Barles and P.E. Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, 4(3):271–283, 1991.

[4] O. Bokanowski, A. Picarelli, and C. Reisinger. High-order filtered schemes for time-dependent second order HJB equations, 2017. Preprint available at https://arxiv.org/pdf/1611.04939.pdf.

[5] L. Bonaventura and R. Ferretti. Semi-Lagrangian methods for parabolic problems in divergence form. *SIAM Journal on Scientific Computing*, 36(5):A2458–A2477, 2014.

[6] F. Camilli and M. Falcone. An approximation scheme for the optimal control of diffusion processes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 29(1):97–122, 1995.

[7] M.G. Crandall, H. Ishii, and P.-L. Lions. User's guide to viscosity solutions of second order partial differential equations. *Bulletin of the American Mathematical Society*, 27(1):1–67, 1992.

[8] K. Debrabant and E.R. Jakobsen. Semi-Lagrangian schemes for linear and fully nonlinear diffusion equations. *Mathematics of Computation*, 82(283):1433–1462, 2013.

[9] M. Falcone and R. Ferretti. *Semi-Lagrangian Approximation Schemes for Linear and Hamilton–Jacobi Equations*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2013.

[10] X. Feng and M. Jensen. Convergent semi-lagrangian methods for the monge–ampère equation on unstructured grids. *SIAM Journal on Numerical Analysis*, 55(2):691–712, 2017.

[11] R. Ferretti. On the relationship between Semi-Lagrangian and Lagrange-Galerkin schemes. *Numerische Mathematik*, 124(1):31–56, 2013.

[12] P.-L. Lions. Optimal control of diffusion processes and Hamilton–Jacobi–Bellman equations part 2: viscosity solutions and uniqueness. *Communications in Partial Differential Equations*, 8(11):1229–1276, 1983.

[13] T.J. Lyons. Uncertain volatility and the risk-free synthesis of derivatives. *Applied Mathematical Finance*, 2(2):117–133, 1995.

[14] J.-L. Menaldi. Some estimates for finite difference approximations. *SIAM Journal on Control and Optimization*, 27(3):579–607, 1989.

[15] D.M. Pooley, P.A. Forsyth, and K.R. Vetzal. Numerical convergence properties of option pricing PDEs with uncertain volatility. *IMA Journal of Numerical Analysis*, 23:241–267, 2003.

[16] C. Reisinger and J. Rotaetxe Arto. Boundary treatment and multigrid preconditioning for semi-Lagrangian schemes applied to Hamilton–Jacobi–Bellman equations. *Journal of Scientific Computing*, 72(1):198–230, 2017.