A Discontinuous Galerkin Semi-Lagrangian Scheme for 1D Hamilton-Jacobi-Bellman Equations

Communications on Applied Mathematics and Computation
https://doi.org/10.1007/s42967-025-00557-4

ORIGINAL PAPER

A Discontinuous Galerkin Semi-Lagrangian Scheme for 1D Hamilton-Jacobi-Bellman Equations

C. De Simone[1] · **A. Festa**[1]

Received: 27 January 2025 / Revised: 24 October 2025 / Accepted: 27 November 2025

Abstract
The Hamilton-Jacobi-Bellman (HJB) equation, due to its nonlinearity, in general does not admit a classical solution, also for regular data. For this reason, the numerical approximation of the solution may pose some additional difficulties, compared to other cases. Over the past four decades, the literature has proposed several numerical approaches. The methods proposed include discontinuous Galerkin (DG) due to its properties of being local, flexible, and robust, but the approach remained underused, due to various technical difficulties, in particular its difficulty in selecting the correct viscosity solution of the problem. In this paper, a numerical method is proposed to solve the evolution HJB equation in one dimension. It consists of the combination of a Semi-Lagrangian (SL) scheme, aimed at reconstructing the characteristic directions, and a DG method, aimed at generating an approximate solution as a linear combination of discontinuous and compactly supported basis functions, defined a priori. In order to evaluate the performance of the proposed method, a collection of numerical experiments with regular, simply continuous, and discontinuous data is presented.

Keywords Semi-Lagrangian (SL) schemes · Hamilton-Jacobi (HJ) equations · Discontinuous Galerkin (DG) method

Mathematics Subject Classification 65M25 · 49L25

1 Introduction

An optimal control problem consists of determining the set of actions and choices that allow the dynamic system to simultaneously satisfy physical constraints and optimize a certain criterion, the objective function. Thanks to the Dynamic Programming Principle, developed by Bellman in the 1950s, the *value function* is defined as a map that links the initial condition with the value of the objective function when the optimal control is applied. According to this principle, the value function associated with a given initial condition can be expressed as the sum of the value function of the future state and the value of the objective function to optimally reach that future state. The Hamilton-Jacobi-Bellman (HJB) equation represents

✉ A. Festa
adriano.festa@polito.it

[1] Dipartimento di Scienze Matematiche "Giuseppe Luigi Lagrange", Corso Duca degli Abruzzi, 24, 10129 Turin, TO, Italy

this concept in differential terms. The practical applications are numerous, and the HJB equation is employed in fields such as aerospace engineering, energy industries, quantitative finance, and image processing.

The HJB equation belongs to the more general class of Hamilton-Jacobi (HJ) equations, which are first-order nonlinear partial differential equations originally studied by Hamilton and subsequently by Jacobi in the context of classical mechanics and calculus of variations. In general, these types of equations do not admit classical solutions, even for regular data. Therefore, in the 1980s, the theory of viscosity solutions was developed by Crandall et al. [8]. This class of solutions satisfies the equation in a weak sense and, under relatively general assumptions, ensures the well-posedness of associated problems. In particular, the theoretical framework that emerges is complete and capable of providing the necessary characterization of the *value function* as the unique viscosity solution for HJB equations.

In this context, a challenge arises in constructing a reliable numerical method capable of capturing the unique viscosity solution. In the literature, several schemes have been proposed to solve HJ equations. Crandall and Lions [10] developed a finite difference method and demonstrated that a monotone scheme of this type converges to the exact solution. From adaptations of strategies used for conservation laws, a class of *Essentially Non-Oscillatory* (ENO) schemes was proposed in [19], and later *Weighted* ENO (WENO) formulations were introduced to solve the HJ equations [3, 15], as well as *central-upwind* methods [17]. Regarding finite elements, Hu and Shu applied the discontinuous Galerkin (DG) method to the HJ equation reformulated as a conservation law [13]. Subsequently, Cheng and Shu developed a direct method for solving HJ equations using DG [5]. Extensions and variants have since emerged [6, 16, 18, 21]. At the same time, semi-Lagrangian (SL) approaches have also been employed for approximating HJ solutions [12]. In particular, the combination with a WENO method was developed in [4], whereas [2] proposed an SL DG method for first- and second-order non-stationary linear partial differential equations. We underline that in this latter work, the nonlinear case is not considered.

The goal of this paper is to introduce, in its easier setting, a new method inspired by Bokanowski and Simarmata [2], for approximating the viscosity solution of HJB equations in their fully nonlinear formulation. Moreover, it is shown that this method, derived from the SL scheme, is unconditionally stable with respect to the time step. At the same time, it preserves two typical qualities of the DG method it draws inspiration from: adaptivity of elements in refinement and polynomial degree (hp-adaptivity), and decomposition of the general problem into simpler, separately solvable parts. In addition to that, the method proposed has the advantage of being easily extended to high-order polynomial bases, without the use of special quadrature formulas different from those already used in the DG formulation. This document is organized as follows.

- Following this Introduction, we briefly recall the key elements of the modern theory of the HJ equations, their connection with an optimal control problem, and the theoretical framework of viscosity solutions, necessary for the well-posedness of the solution. A reader who already knows these techniques may skip this part.
- In Sect. 2, the proposed discontinuous Galerkin semi-Lagrangian (DGSL) method is described, progressively leading to the formulation and implementation aspects connected to the evolutionary HJB equation in one dimension.
- In Sect. 3, the performed tests are presented, and the discretization error is evaluated, enabling a comparison with other known methods from the literature and their high-order formulations.

– Finally, in Sect. 4, the test results are commented on, highlighting the strengths and weaknesses of the proposed method, and possible directions for future research are suggested.

1.1 A Coincise Introduction to Dynamical Programming

A control system (CS) in dimension d over the time interval $\mathcal{T} \equiv [t, T]$ is governed by the dynamic equations

$$\begin{cases} \dot{y}(s) = f(s, y(s), u(s)), & 0 \leqslant t < s < T, \\ y(t) = x, & x \in \mathbb{R}^d, \end{cases} \tag{CS}$$

where $y\colon \mathcal{T} \to \mathbb{R}^d$ represents the system state function; $u\colon \mathcal{T} \to U$ is the control function with values in the compact space $U \subset \mathbb{R}^m$; $f\colon \mathcal{T} \times \mathbb{R}^d \times U \to \mathbb{R}^d$ is the system dynamics map. For every control function belonging to the set $\mathcal{U}_{\text{ad}} = \{u\colon \mathcal{T} \to U,\ \text{measurable}\}$, Carathéodory's theorem guarantees the existence and the uniqueness of a continuous and almost everywhere differentiable solution for (CS), under the assumptions of f being continuous, Lipschitz in y (uniformly with respect to s and u), and measurable with respect to s [12, Theorem 8.1].

To formulate an optimal control problem, it is necessary to introduce a cost functional (CS) that one aims to minimize. In Bolza's formulation, the functional for finite-horizon problems, $T \in \mathbb{R}$, takes the form

$$J_{t,x}(u) = \int_t^T \ell(y_x^u(s), u(s))\mathrm{e}^{-\lambda(s-t)}\,\mathrm{d}s + L(y_x^u(T))\mathrm{e}^{-\lambda(T-t)},$$

where y_x^u is the trajectory resulting from the control $u \in \mathcal{U}_{\text{ad}}$ with the initial condition x at time t. The map $\ell\colon \mathbb{R}^d \times U \to \mathbb{R}$ is the running cost (per unit time), assumed to be Lipschitz in y (uniformly with respect to u) and bounded, while $L\colon \mathbb{R}^d \to \mathbb{R}$ is the terminal cost function, also assumed to be Lipschitz and bounded. $\lambda \geqslant 0$ is a parameter that makes costs comparable across different time instants, referred to in economic contexts as the discount factor.

The optimal control problem can thus be described in terms of the *value function*

$$v(t, x) = \inf_{u \in \mathcal{U}_{\text{ad}}} J_{t,x}(u),$$

whose meaning is the minimum cost associated with the initial condition (t, x). Under the assumptions made on the dynamics and cost functions, it follows that $v\colon \mathcal{T} \times \mathbb{R}^d \to \mathbb{R}$ is bounded and Lipschitz [1, Chap. 3, Proposition 3.1].

Moreover, the *value function* satisfies the Principle of Dynamic Programming (DPP), also known as Bellman's Principle of Optimality, i.e., for any $\tau \in (t, T]$

$$v(t, x) = \inf_{u \in \mathcal{U}_{\text{ad}}} \left\{ \int_t^\tau \ell(y_x^u(s), u(s))\mathrm{e}^{-\lambda(s-t)}\,\mathrm{d}s + v(\tau, y_x^u(\tau))\mathrm{e}^{-\lambda(\tau-t)} \right\}, \tag{DPP}$$

the proof of this result is detailed in [11, Sect. 10.3, Theorem 1]. Essentially, the initial problem has been divided into two parts (Fig. 1). Specifically, $v(t, x)$ is the value associated with the optimal control problem over the time interval $[t, \tau]$ with the running cost ℓ and the terminal cost $v(\tau, y_x^{u^*}(\tau))$. The latter is still an optimal control problem over the time interval $[\tau, T]$ with the running cost ℓ and the terminal cost $L(y_x^{u^*}(T))$.

Under the assumption of a continuously differentiable *value function* (an assumption that can later be weakened), an infinitesimal version of (DPP) can be derived. This results in a

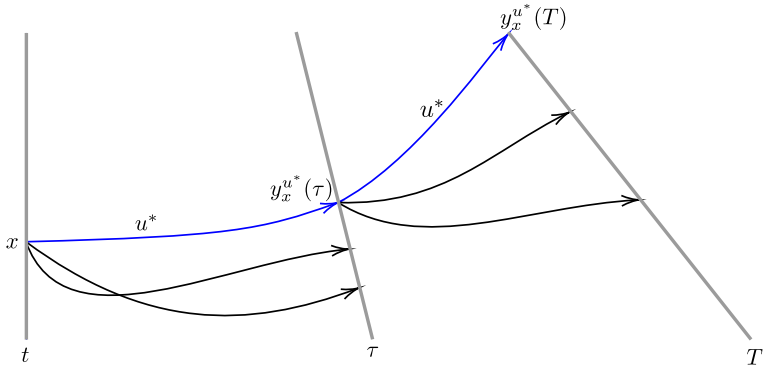

Fig. 1 The optimization problem is divided into two parts over the intervals $[t, \tau]$ and $[\tau, T]$. The optimal choices are represented in blue

partial differential equation that, coupled with an appropriate boundary condition, is solved by v. Indeed, starting from (CS), we have

$$\begin{aligned}
&y_x^u(t+h) = y_x^u(t) + hf(t, x, u(t)) + o(h),\\
&v(t+h, y_x^u(t+h)) = v(t,x) + h\frac{\partial v}{\partial t}(t,x) + \nabla_x v(t,x)\cdot hf(t,x,u(t)) + o(h),\\
&\int_t^{t+h} \ell(y_x^u(s), u(s))\mathrm{e}^{-\lambda(s-t)}\,\mathrm{d}s = h\ell(x,u(t)) + o(h).
\end{aligned}$$

Substituting these equalities into (DPP) for $\tau = t + h$ and assuming $\lambda = 0$, we obtain

$$\begin{aligned}
v(t,x) &= \inf_{u\in\mathcal{U}_{\mathrm{ad}}}\left\{h\ell(x,u(t)) + v(t,x) + h\frac{\partial v}{\partial t}(t,x) + \nabla_x v(t,x)\cdot hf(t,x,u(t)) + o(h)\right\},\\
0 &= \inf_{u\in\mathcal{U}_{\mathrm{ad}}}\left\{h\ell(x,u(t)) + h\frac{\partial v}{\partial t}(t,x) + \nabla_x v(t,x)\cdot hf(t,x,u(t)) + o(h)\right\},\\
0 &= \inf_{u\in\mathcal{U}_{\mathrm{ad}}}\left\{\ell(x,u(t)) + \frac{\partial v}{\partial t}(t,x) + \nabla_x v(t,x)\cdot f(t,x,u(t)) + \frac{o(h)}{h}\right\}.
\end{aligned}$$

Taking the limit as $h \to 0$, the higher-order infinitesimal term vanishes, and the terms involving $u(s)$ for $s > t$ disappear as well. Thus, the infimum is evaluated directly in the space U, yielding

$$\frac{\partial v}{\partial t}(t,x) = -\inf_{u\in U}\{\ell(x,u) + \nabla_x v(t,x)\cdot f(t,x,u)\}$$

for $0 \leqslant t < T$, $x \in \mathbb{R}^d$, with a boundary condition derived from the knowledge of the *value function* at (T, x)

$$v(T,x) = \inf_{u\in\mathcal{U}_{\mathrm{ad}}} J_{T,x}(u) = \inf_{u\in\mathcal{U}_{\mathrm{ad}}} L(y_x^u(T)) = L(x).$$

Equivalently, for $0 \leqslant t < T$ and $x \in \mathbb{R}^d$, the HJB equation can be written as

$$\begin{cases}
\frac{\partial v}{\partial t}(t,x) = \sup_{u\in U}\{-\ell(x,u) - \nabla_x v(t,x)\cdot f(t,x,u)\},\\
v(T,x) = L(x).
\end{cases} \tag{HJB}$$

The partial differential equation in (HJB), within the context of finite-horizon optimal control problems, is referred to as the HJB equation because it represents Bellman's principle in terms of the HJ evolutionary equation, given in (HJ). Specifically, for $0 < t \leqslant T$ and $x \in \mathbb{R}^d$ the formulation

$$\begin{cases} \frac{\partial v}{\partial t}(t,x) + H(t,x,\nabla_x v(t,x)) = 0, \\ v(0,x) = v_0(x) \end{cases} \tag{HJ}$$

with H being the Hamiltonian operator, is equivalent to (HJB) for optimal control problems when $H(t,x,p) = \inf_{u\in U}\{\ell(x,u) + p \cdot f(t,x,u)\}$ and using the variable transformation $t \to T - t$.

1.2 A Coincise Introduction to Viscosity Solutions

In general, it is known that the Cauchy problem associated with the HJ equations (HJ) does not admit a classical solution, while weakening the concept of the solution to a.e. solutions cannot always guarantee uniqueness.

Example 1 The one-dimensional (1D) problem with $x \in \mathbb{R}$ and $0 < t \leqslant T$:

$$\begin{cases} \frac{\partial v}{\partial t}(t,x) + \left|\frac{\partial v}{\partial x}(t,x)\right| = 0, \\ v(0,x) = |x|, \end{cases} \tag{1}$$

corresponds to (HJ) when $H(t,x,p) = |p|$ and $v_0(x) \equiv 0$. It is easy to verify that this problem admits infinitely many a.e. weak solutions. Below are three Lipschitz functions: $v_a(t,x) = |x| - t$,

$$v_b(t,x) = \begin{cases} |x| - t, & \text{if } |x| > t, \\ t - |x|, & \text{if } |x| \leqslant t, \end{cases} \qquad v_c(t,x) = \begin{cases} |x| - t, & \text{if } |x| > t, \\ 0, & \text{if } |x| \leqslant t, \end{cases}$$

that satisfy (1) almost everywhere, except at points lying on $x = 0$ (cases a and b) and on $x = \pm t$ (cases b and c).

For this reason, in the early 1980s, Crandall et al. proposed the notion of a viscosity solution [9] and its properties [8] for scalar partial differential equations, a class to which HJ equations belong. Such a solution exists, is unique, and is stable [1] and it follows the definition.

Definition 1 A function $v \in C((0,T)\times\mathbb{R}^d)$ is called a *viscosity subsolution* in $(0,T)\times\mathbb{R}^d$ for (HJ) if, for any $\phi \in C^1((0,T)\times\mathbb{R}^d)$,

$$\frac{\partial \phi}{\partial t}(t_{\mathrm{M}}, x_{\mathrm{M}}) + H(t_{\mathrm{M}}, x_{\mathrm{M}}, \nabla_x \phi(t_{\mathrm{M}}, x_{\mathrm{M}})) \leqslant 0 \tag{SubS}$$

at every point $(t_{\mathrm{M}}, x_{\mathrm{M}}) \in (0,T)\times\mathbb{R}^d$ where $v - \phi$ achieves a local maximum.

Definition 2 A function $v \in C((0,T)\times\mathbb{R}^d)$ is called a *viscosity supersolution* in $(0,T)\times\mathbb{R}^d$ for (HJ) if, for any $\phi \in C^1((0,T)\times\mathbb{R}^d)$,

$$\frac{\partial \phi}{\partial t}(t_{\mathrm{m}}, x_{\mathrm{m}}) + H(t_{\mathrm{m}}, x_{\mathrm{m}}, \nabla_x \phi(t_{\mathrm{m}}, x_{\mathrm{m}})) \geqslant 0 \tag{SuperS}$$

at every point $(t_{\mathrm{m}}, x_{\mathrm{m}}) \in (0,T)\times\mathbb{R}^d$ where $v - \phi$ achieves a local minimum.

Definition 3 A function $v \in C((0,T)\times\mathbb{R}^d)$ is called a *viscosity solution* in $(0,T)\times\mathbb{R}^d$ for (HJ) if it is both a subsolution and a supersolution.

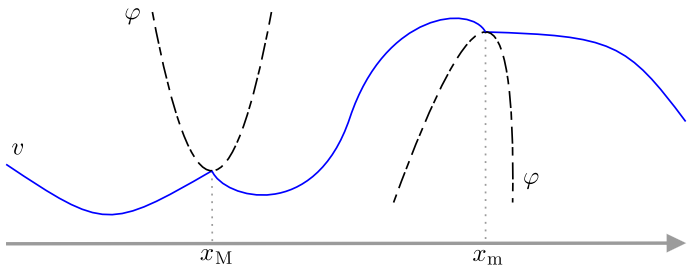

Fig. 2 The test function ϕ for v as a candidate viscosity subsolution (at x_M) and supersolution (at x_m)

The functions ϕ introduced in these definitions are called test functions.

By virtue of conditions (SubS) and (SuperS), it is never restrictive to test a candidate subsolution using functions ϕ such that $v(q_M) = \phi(q_M)$, and a candidate supersolution using functions ϕ such that $v(q_m) = \phi(q_m)$. See Fig. 2.

A peculiar property of solutions of this type is that, in general, they are not preserved under a sign change in the equation. Specifically, viscosity solutions for $\frac{\partial v}{\partial t}(t,x) + H(t,x,\nabla_x v(t,x)) = 0$ almost always differ from those for $-\frac{\partial v}{\partial t}(t,x) - H(t,x,\nabla_x v(t,x)) = 0$.

2 A Discontinuous Galerkin Semi-Lagrangian (DGSL) Method for the Hamilton-Jacobi-Bellman (HJB) Equation

In solving hyperbolic problems, numerical schemes sometimes require stability conditions related to the time step, especially in simulations involving long time durations. An SL method ensures the necessary stability without constraints on the discretization step in time and space [12, Subsect. 5.1.3, Stability]. Along with this property, the goal is to achieve, within a single scheme, the benefits provided by reconstructing the solution using Discontinuous Finite Elements.

2.1 The Semi-Lagrangian (SL) Scheme in the Linear Case

In general, SL methods are defined as such because, at each time step, the solution is determined by following the trajectories of individual parts, i.e., from the Lagrangian perspective. For hyperbolic equations, this translates to analyzing the system's evolution through the method of characteristics.

Consider a spatial domain of dimension $d = 1$ and the transport problem with a source term

$$\begin{cases} v_t(t,x) + f(t,x)v_x(t,x) = g(t,x), & (t,x) \in (0,T] \times \mathbb{R}, \\ v(0,x) = v_0(x), & x \in \mathbb{R}, \end{cases} \tag{2}$$

where $f\colon (0,T] \times \mathbb{R} \to \mathbb{R}$ is the convection term and $g\colon (0,T] \times \mathbb{R} \to \mathbb{R}$ represents the forcing term.

The idea of the method of characteristics is to identify, for each $x \in \mathbb{R}$, a curve in the region of the plane $[0,t] \times \mathbb{R}$ that connects (t,x) with the horizontal axis and along which the original problem can be rewritten as a first-order ordinary differential equation. Thus, considering the

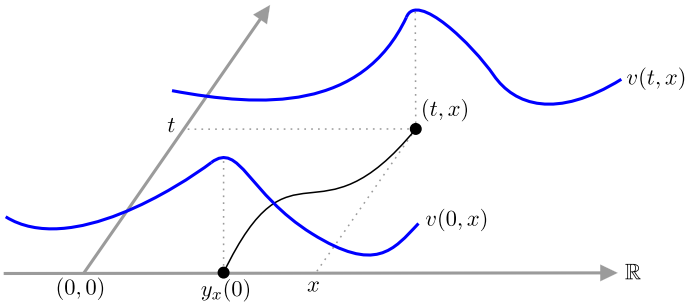

Fig. 3 The method of characteristics for the equation $v_t + f v_x = 0$ with non-constant f and a known initial condition. The characteristic $(s, y_x(s))$, $s \in [0, t]$, associated with the point (t, x) is represented in black

equation of problem (2), the solution v, assuming that it exists and is sufficiently regular, is derived over time along a curve of the form $(s, y_x(s))$

$$\begin{aligned}\frac{\mathrm{d}v}{\mathrm{d}t}(t,x) &= \frac{\mathrm{d}v}{\mathrm{d}t}(t, y_x(t)) = v_t(t, y_x(t)) + \dot{y}_x(t) v_x(t, y_x(t)) \\ &= v_t(t,x) + \dot{y}_x(t) v_x(t,x) = v_t(t,x) + f(t,x) v_x(t,x) = g(t,x),\end{aligned}$$

where the last equality comes imposing that $y_x(s)$ solves the following dynamic system:

$$\begin{cases} \dot{y}(s) = f(s, y(s)), & 0 < s < t \leqslant T, \\ y(t) = x, & x \in \mathbb{R}. \end{cases} \tag{3}$$

Then, the solution v is explicitly computed as

$$v(t,x) = v_0(y_x(0)) + \int_0^t g(s, y_x(s))\,\mathrm{d}s, \tag{4}$$

and $(s, y_x(s))$ is defined as the characteristic curve of problem (2). In particular, when the source term $g \equiv 0$, we obtain

$$\frac{\mathrm{d}v}{\mathrm{d}t}(t,x) = v_t(t,x) + \dot{y}_x(t) v_x(t,x) = 0 \implies v(t,x) = v_0(y_x(0)),$$

hence, knowing the initial condition v_0 and the characteristic curves, one has complete knowledge of $v(t,x)$, under some mild regularity assumptions on f, g, and v_0. From [12, Theorem 1.1], the following result is reported.

Theorem 1 *Regarding problem* (2), *let* $f, g \in C^1((0,T) \times \mathbb{R})$ *and* $v_0 \in C^1(\mathbb{R})$. *Then, the solution exists, is unique, and* $v \in C^1((0,T) \times \mathbb{R})$.

A graphical representation of the method of characteristics, in the case $g \equiv 0$, is shown in Fig. 3.

The SL scheme arises from evaluating the formula (4) at a finite number of points in the domain:

$$v(t_{n+1}, x_i) = v(t_n, y_{x_i}(t_n)) + \int_{t_n}^{t_{n+1}} g(s, y_{x_i}(s))\,\mathrm{d}s, \tag{SL}$$

which suggests a numerical scheme to compute the solution at the assigned spatial grid points $x_i \in \mathbb{R}$ at each selected time step, e.g., $t_n = n\Delta t \in (0, T]$.

However, when the solution of the dynamic system (3) is not known, approximations must be introduced. A possible and straightforward choice is the explicit Euler method, yielding

$$x_i = y_{x_i}(t_{n+1}) \approx y_{x_i}(t_n) + f(t_{n+1}, x_i)\Delta t \implies y_{x_i}(t_n) \approx x_i - f(t_{n+1}, x_i)\Delta t.$$

For the integral term, the simplest discretization in this context is

$$\int_{t_n}^{t_{n+1}} g(s, y_{x_i}(s))\,\mathrm{d}s \approx g(t_{n+1}, y_{x_i}(t_{n+1}))\Delta t = g(t_{n+1}, x_i)\Delta t.$$

By substituting the above approximations into (SL), we obtain [7]

$$\begin{cases} \tilde{v}(t_{n+1}, x_i) = \tilde{v}(t_n, x_i - f(t_{n+1}, x_i)\Delta t) + g(t_{n+1}, x_i)\Delta t, \\ \tilde{v}(0, x_i) = v_0(x_i). \end{cases} \tag{CIR}$$

The notation $\tilde{v}$ not only refers to the approximations introduced in the method but also to the numerical reconstruction of the solution across the spatial domain.

A fundamental characteristic of the scheme (CIR), as reported in [12, Subsect. 5.1.3] when discussing the consistency, stability, and convergence of the method, is the freedom to consider a node spacing x_i much smaller than the distance traveled by the information on the value of v in a time Δt.

2.2 A Galerkin Approximation for the SL Method

In a generic Galerkin approximation of the solution to a differential problem, we define:

Definition 4 In $\mathbb{R}^d$, a *Finite Element* is defined as the triplet $(E, V_{E,h}, \mathcal{L}_E)$, where

- $E \subset \mathbb{R}^d$ is a compact, connected, and non-empty set, such that its boundary is Lipschitz continuous and $E = \overline{E}$ (the closure of its interior);
- $V_{E,h}$ is a finite-dimensional linear space of scalar functions defined on E, with dimension N_E;
- $\mathcal{L}_E = \{l_j\colon\ j \in \mathbb{N} \text{ and } 1 \leqslant j \leqslant N_E\}$ is a set of linear forms $l_j\colon\ V_{E,h} \to \mathbb{R}$ that is unisolvent (i.e., the forms uniquely determine the functions in $V_{E,h}$) for $V_{E,h}$.

Since our domain of interest is $[a, b]$, we discretize it into N disjoint intervals, i.e., we consider the partition $\mathcal{T}_h = \{I_i : i \in \mathbb{N} \text{ and } 1 \leqslant i \leqslant N\}$ with $I_i = [x_{i-1}, x_i]$ such that

- $[a, b] = \bigcup_{i=1}^{N} I_i$;
- $a = x_0 < x_1 < \cdots < x_{N-1} < x_N = b$;
- $x_i - x_{i-1} = h_i \neq 0$ for every $i \in \{1, \cdots, N\}$;
- $h = \max_{i\in\{1,\cdots,N\}} h_i$.

Moreover, for simplicity, let us consider an equispaced grid so that $x_i = a + \mathrm{i}h$ with $h = (b - a)/N$.

Let $V_h \subset W$ be the finite-dimensional space used to approximate the solution $v \in V \subset W$, without necessarily imposing $V_h \subset V$. For example, in the case of continuous finite elements, one could have $V = H^1([a, b]) \subset L^2([a, b]) = W$ and

$$V_h = \{v \in C^0([a, b])\colon\ v_{|I_i} \in \mathbb{P}_r(I_i), \quad \forall I_i \in \mathcal{T}_h\}.$$

Thus, starting from (SL), the weak formulation is obtained by multiplying both sides by a generic $w \in W$ and integrating over $[a, b]$, i.e.,

$$\int_a^b v(t_{n+1}, x)w(x)\,\mathrm{d}x = \int_a^b v(t_n, y_x(t_n))w(x)\,\mathrm{d}x + \int_a^b \int_{t_n}^{t_{n+1}} g(s, y_x(s))w(x)\,\mathrm{d}s\mathrm{d}x,$$

from which the approximated problem is immediately derived: find $v_h \in V_h$ such that for any $w_h \in V_h$,

$$\int_a^b v_h^{n+1}(x)w_h(x)\,\mathrm{d}x = \int_a^b v_h^n(y_x(t_n))w_h(x)\,\mathrm{d}x + \int_a^b G(t_n, t_{n+1}, x)w_h(x)\,\mathrm{d}x,$$

where, to simplify notations

$$v_h^n(x) = v_h(t_n, x), \quad G(t_n, t_{n+1}, x) = \int_{t_n}^{t_{n+1}} g(s, y_x(s))\,\mathrm{d}s.$$

Every element of the finite-dimensional space V_h, of dimension N_h, can be expressed as a linear combination of its basis

$$v_h^n(x) = \sum_{k=1}^{N_h} v_k^n \varphi_k(x)$$

with φ_k: $[a, b] \to \mathbb{R}$ for each k.

Returning to the weak formulation, it is equivalent, due to the linearity property of the integral with respect to summation, to the linear system with unknowns $v_1^{n+1}, \cdots, v_{N_h}^{n+1}$ reported below. That is, for any $j \in \{1, \cdots, N_h\}$, it must hold that

$$\sum_{k=1}^{N_h} v_k^{n+1} \int_a^b \varphi_k(x)\varphi_j(x)\,\mathrm{d}x = \sum_{k=1}^{N_h} v_k^n \int_a^b \varphi_k(y_x(t_n))\varphi_j(x)\,\mathrm{d}x + \int_a^b G(t_n, t_{n+1}, x)\varphi_j(x)\,\mathrm{d}x.$$

In the matrix form, this corresponds to solving

$$\begin{aligned} &\mathsf{M}\mathsf{v}^{n+1} = \mathsf{C}^n\mathsf{v}^n + \mathsf{G}^n \\ &\text{with} \begin{cases} \mathsf{M} \in \mathbb{R}^{N_h \times N_h}, & \mathsf{M}_{jk} = \displaystyle\int_a^b \varphi_k(x)\varphi_j(x)\,\mathrm{d}x, \\ \mathsf{v}^n \in \mathbb{R}^{N_h}, & \mathsf{v}_k^n = v_k^n, \\ \mathsf{C}^n \in \mathbb{R}^{N_h \times N_h}, & \mathsf{C}_{jk}^n = \displaystyle\int_a^b \varphi_k(y_x(t_n))\varphi_j(x)\,\mathrm{d}x, \\ \mathsf{G}^n \in \mathbb{R}^{N_h}, & \mathsf{G}_j^n = \displaystyle\int_a^b G(t_n, t_{n+1}, x)\varphi_j(x)\,\mathrm{d}x. \end{cases} \end{aligned} \tag{5}$$

However, this general formulation does not highlight the potential and differences of the DG method compared to its more classical version.

2.3 DGSL Implementation-$\mathbb{P}_r$

Consider the space of discontinuous polynomial functions along the interfaces that form the *mesh*,

$$V_h = \mathbb{P}_r(\mathcal{T}_h) = \{v \in L^2([a, b])\colon v_{|I_i} \in \mathbb{P}_r(I_i), \quad \forall I_i \in \mathcal{T}_h\},$$

together with the Finite Element $(I, \mathbb{P}_r(I), \mathcal{L}_{\text{Lagr}})$ in each interval I of the *mesh*. With reference to Definition 4, it is understood that for any $i \in \{1, \cdots, N\}$

- $E = I_i$;
- $V_{E,h} = \mathbb{P}_r(I_i)$ and $N_E = r + 1$;
- $\mathcal{L}_E = \mathcal{L}_{\text{Lagr}} = \{l_j : l_j(v) = v(\xi_j)\}$;

where $x_{i-1} = \xi_1 < \xi_2 < \cdots < \xi_r < \xi_{r+1} = x_i$ and the set $\mathcal{L}_{\text{Lagr}}$ satisfies the unisolvence property for the space $\mathbb{P}_r(I)$[1]. Therefore, the canonical basis $\{\phi_1, \cdots, \phi_{r+1}\}$ in the interval I_i is identified by requiring

$$l_j(\phi_k) = \phi_k(\xi_j) = \delta_{jk}, \quad \forall j, k \in \{1, \cdots, r+1\}.$$

The union of the sets formed by the bases of all intervals $I \in \mathcal{T}_h$, once the domain they are defined on is appropriately extended, forms a basis for the space V_h. Explicitly, $N_h = \dim(V_h) = N(r+1)$, imposing

$$\varphi_k(x) = \begin{cases} \phi_j(x), & \text{if } x \in I_i, \\ 0, & \text{otherwise,} \end{cases}$$

through the relation $k = (i-1)(r+1) + j$, we obtain

$$v_h^n(x) = \sum_{k=1}^{N_h} v_k^n \varphi_k(x) = \sum_{i=1}^{N} \sum_{j=1}^{r+1} v_k^n \phi_j(x) = \sum_{i=1}^{N} \sum_{j=1}^{r+1} v_h^n|_{I_i}(\xi_j)\phi_j(x),$$

where the basis ϕ_j depends on the interval I_i it refers to.

Figure 3 shows a graphical representation of the global basis functions of the proposed method when $r = 1$ and a possible numerical solution that can be obtained (Fig. 4).

Thus, regarding the matrix formulation described in Eq. (5), we have

$$\mathsf{M} = \begin{bmatrix} \boxed{\mathsf{M}_1} & & 0 \\ & \ddots & \\ 0 & & \boxed{\mathsf{M}_N} \end{bmatrix} \text{ with } (\mathsf{M}_i)_{jk} = \int_{I_i} \phi_k(x)\phi_j(x)\mathrm{d}x,$$

since the integral over the entire space $[a, b]$ of two generic basis functions, ϕ_k and ϕ_j, equals 0 if the interval they refer to is different, otherwise it coincides with the integral over the interval I_i to which they belong. Moreover, thanks to the assumptions of an equispaced grid $\mathcal{T}_h$ and the same finite element for each interval of the *mesh*, all non-zero blocks of M are equal, i.e., $\mathsf{M}_i = \mathsf{S}_\mathsf{M}$ for each i.

Regarding the other matrices in Eq. (5), the only simplification relates to the integration domain, which becomes the interval I_i where the basis function $\varphi_j|_{I_i} = \phi_j$ is non-zero.

Thus, by reshaping the column vector v^n into a matrix data structure with $r+1$ rows (as the number of degrees of freedom of $\mathcal{L}_{\text{Lagr}}$) and N columns (as the number of intervals), we obtain the proposed DGSL method, that is

$$\mathsf{S}_\mathsf{M}\mathsf{v}^{n+1} = \mathsf{I}_\mathsf{C}^n(\mathsf{v}^n) + \mathsf{G}^n$$

[1] Let $f = a_0 + a_1 x + \cdots + a_k x^k \in \mathbb{P}_k$, evaluating f at $k+1$ points gives a linear system of $k+1$ equations in as many unknowns, $\mathsf{Va} = \mathsf{f}$, where V is the Vandermonde matrix and is invertible if $\det(\mathsf{V}) = \prod_{0 \leqslant i < j \leqslant k}(x_j - x_i) \neq 0$, i.e., if the $k+1$ points are distinct.

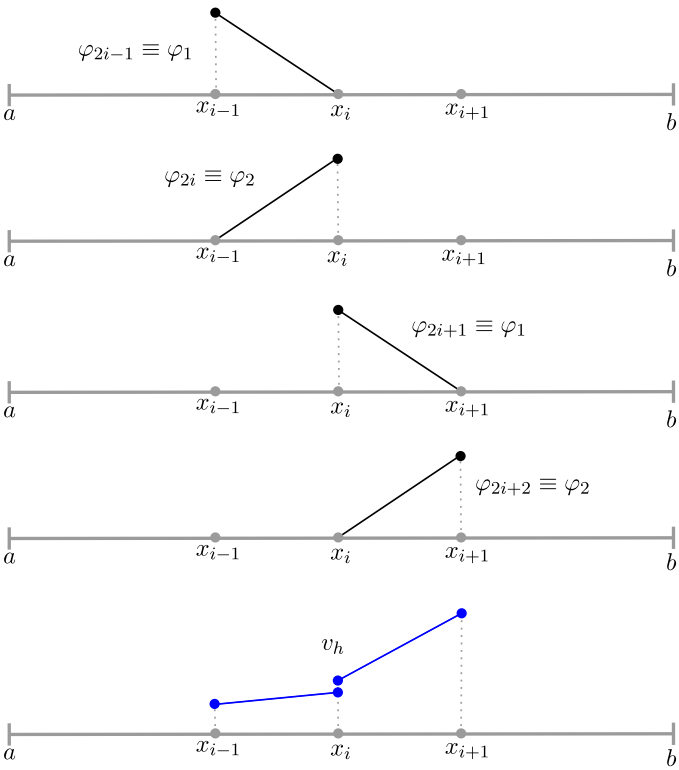

Fig. 4 Representation of the global basis functions (in black) of $(I_i, \mathbb{P}_1(I_i), \mathcal{L}_{\text{Lagr}})$ and $(I_{i+1}, \mathbb{P}_1(I_{i+1}), \mathcal{L}_{\text{Lagr}})$ for discontinuous finite elements and a possible approximation v_h (in blue)

$$\text{with} \begin{cases} \mathsf{S}_\mathsf{M} \in \mathbb{R}^{(r+1)\times(r+1)}, & (\mathsf{S}_\mathsf{M})_{jk} = \displaystyle\int_I \phi_k(x)\phi_j(x)\,\mathrm{d}x, \\ \mathsf{v}^n \in \mathbb{R}^{(r+1)\times N}, & \mathsf{v}^n_{jk} = v_h^n|_{I_k}(\xi_j), \\ \mathsf{I}^n_\mathsf{C}(\mathsf{v}^n) \in \mathbb{R}^{(r+1)\times N}, & (\mathsf{I}^n_\mathsf{C}(\mathsf{v}^n))_{jk} = \displaystyle\int_{I_k} v_h^n(y_x(t_n))\phi_j(x)\,\mathrm{d}x, \\ \mathsf{G}^n \in \mathbb{R}^{(r+1)\times N}, & \mathsf{G}^n_{jk} = \displaystyle\int_{I_k} G(t_n, t_{n+1}, x)\phi_j(x)\,\mathrm{d}x, \end{cases}$$

where $\mathsf{I}^n_\mathsf{C}\colon \mathbb{R}^{(r+1)\times N} \to \mathbb{R}^{(r+1)\times N}$ represents a function that operates on the elements of the matrices. This notation is preferred because, unlike what is described for the matrix M, there are no patterns simplifying the structure of C^n in Eq. (5), as the characteristic $y_x(t_n)$ generates a translation of the basis function that is not known a priori. Indeed, in general

$$\int_{I_i} \phi_k(y_x(t_n))\phi_j(x)\,\mathrm{d}x \neq 0$$

for any choice of j, k.

In this work, the integrals $(\mathsf{I}^n_\mathsf{C}(\mathsf{v}^n))_{jk}$ are computed directly, since there are $N(r+1)$ terms instead of $N^2(r+1)^2$, following a specific order that allows reuse of already performed calculations. Consider a generic quadrature formula, i.e., for a generic function $f\colon \Omega \subset \mathbb{R}^n \to \mathbb{R}$ a collection of nodes $(x_q \in \Omega)$ and their corresponding weights (ω_q) with the

number N_q such that $\int_\Omega f(x)\mathrm{d}\Omega \approx \sum_{q=1}^{N_q} \omega_q f(x_q)$, we then have

$$\begin{aligned}\int_{I_k} v_h^n(y_x(t_n))\phi_j(x)\,\mathrm{d}x &\approx \sum_{q=1}^{N_q} w_q v_h^n(y_{x_q}(t_n))\phi_j(x_q)\\ &= \sum_{q=1}^{N_q} w_q \left[\sum_{\ell=1}^{r+1} v_h^n|_{I_{q'}}(\xi_\ell)\phi_\ell(y_{x_q}(t_n))\right]\phi_j(x_q) \end{aligned} \tag{6}$$

with q' being the index associated with the interval in which the point $y_x(t_n)$ falls, formally

$$q'(y_{x_q}(t_n)) = \{i \in \{1, \cdots, N\}\colon y_{x_q}(t_n) \in I_i\}.$$

It is important to note that the term inside the square brackets in Eq. (6) is the quantity that does not need to be recalculated for each ϕ_j of the interval I_k, as it is common to all.

Finally, the new formulation highlights the local nature of the approximation with a discontinuous finite element method. Indeed, from the block-diagonal structure of M, it is suggested that the numerical solution in a specific interval of the discretization is not directly related to what happens in other intervals. In this sense, the solution of the partial differential equation is divided into smaller problems, which are solved separately, unlike continuous finite elements.

2.4 Boundary Conditions

In the theory and methods described so far, the presence of boundary conditions has not been mentioned, which arise naturally when defining a problem on a bounded domain.

In general, when evaluating the characteristic curve, it may happen that the curve exceeds the boundaries of the domain of interest, $y_x(t_n) \notin [a, b]$, creating ambiguity in the absence of information, which is where boundary conditions come into play.

In the case of periodic boundary conditions, which for the problem (2) formally translates to $v(t, a) = v(t, b)$ for every $t \in (0, T]$, there is no need to modify the previously described DGSL-$\mathbb{P}_r$ formulation. The only consideration is that the value assumed by the approximation v_h at any point outside the domain of interest is identified with an internal point according to the relation

$$v_h^n(y_x(t_n)) = v_h^n(y_x(t_n) \pm m(b-a)), \quad \forall m \in \mathbb{N}.$$

Instead, in the case of Dirichlet boundary conditions, generally non-homogeneous, it is necessary to have information about the boundary through functions $B(t)$, for example $B_a(t)$ for the left boundary ($x = a$) and $B_b(t)$ for the right boundary ($x = b$) with $t \in (0, T]$. Then, for a characteristic coming from outside the spatial domain, $y_x(t_n) \notin [a, b]$, we define

$$\theta_x = \sup\{s \in (t_n, t_{n+1}]\colon y_x(s) \in \mathbb{R}\backslash(a, b)\}$$

as the time of the first intersection with the boundary. If there are no intersections in the interval $(t_n, t_{n+1}]$, by convention, we set $\theta_x = \sup\{\varnothing\} = -\infty$.

Thus, considering the exact SL scheme (SL), taking into account the Dirichlet boundary conditions, we have

$$v(t_{n+1}, x_i) = \begin{cases} v(t_n, y_{x_i}(t_n)) + \displaystyle\int_{t_n}^{t_{n+1}} g(s, y_{x_i}(s))\,\mathrm{d}s, & \text{if } \theta_{x_i} \leqslant t_n, \\ B(\theta_{x_i}) + \displaystyle\int_{\theta_{x_i}}^{t_{n+1}} g(s, y_{x_i}(s))\,\mathrm{d}s, & \text{if } \theta_{x_i} > t_n. \end{cases}$$

Similarly, with reference to the DGSL-$\mathbb{P}_r$ method, in Eq. (6) the following case arises:

$$v_h^n(y_{x_q}(t_n)) = \begin{cases} \displaystyle\sum_{\ell=1}^{r+1} v_h^n|_{I_{q'}}(\xi_\ell)\phi_\ell(y_{x_q}(t_n)), & \text{if } \theta_{x_q} \leqslant t_n, \\ B(\theta_{x_q}), & \text{if } \theta_{x_q} > t_n. \end{cases}$$

Moreover, the term G_{jk}^n becomes

$$\mathsf{G}_{jk}^n = \int_{I_k} \int_{\max\{t_n, \theta_x\}}^{t_{n+1}} g(s, y_x(s))\phi_j(x)\,\mathrm{d}s\,\mathrm{d}x.$$

2.5 Application to the HJB Case

We recall from Sect. 1.1 that the value function of the optimal control problem governed by the dynamics f, current cost ℓ, terminal cost L, control space U, $\lambda = 0$, and domain $[0, T) \times \mathbb{R}$, is a viscosity solution of the HJB equation

$$\begin{cases} v_t(t, x) = \sup_{u \in U} \{-\ell(x, u) - f(t, x, u)v_x(t, x)\}. \\ v(T, x) = L(x). \end{cases} \tag{HJB}$$

Let us discuss the adaptation of DGSL to this fully nonlinear case. Let $[a, b]$ be the domain of interest, and consider the same *mesh* $\mathcal{T}_h$ from Sect. 2.2, and let $V_h = \mathbb{P}_r(\mathcal{T}_h)$. Then, on the interval I_k, the weak formulation described in Sect. 2.3 imposes

$$\begin{aligned} &\int_{I_k} V_h^{n+1}(x)\phi_j(x)\,\mathrm{d}x \\ &= \int_{I_k} \inf_{u \in \mathcal{U}_{\mathrm{ad}}^{\Delta t}} \left\{ V_h^n(y_x^u(t_n)) + \int_{t_n}^{t_{n+1}} \ell(y_x^u(s), u(s))\,\mathrm{d}s \right\} \phi_j(x)\,\mathrm{d}x, \end{aligned}$$

where $\{\phi_1, \cdots, \phi_{r+1}\}$ represents the canonical basis of the Finite Element $(I_k, \mathbb{P}_r(I_k), \mathcal{L}_{\mathrm{Lagr}})$. To compute the integral on the right-hand side, a numerical quadrature formula is directly introduced, so

$$\begin{aligned} &\int_{I_k} \inf_{u \in \mathcal{U}_{\mathrm{ad}}^{\Delta t}} \left\{ V_h^n(y_x^u(t_n)) + \int_{t_n}^{t_{n+1}} \ell(y_x^u(s), u(s))\,\mathrm{d}s \right\} \phi_j(x)\,\mathrm{d}x \\ &= \sum_{q=1}^{N_q} w_q \inf_{u \in \mathcal{U}_{\mathrm{ad}}^{\Delta t}} \left\{ V_h^n(y_{x_q}^u(t_n)) + \int_{t_n}^{t_{n+1}} \ell(y_{x_q}^u(s), u(s))\,\mathrm{d}s \right\} \phi_j(x_q) \\ &= \sum_{q=1}^{N_q} w_q \inf_{u \in U} \{V_h^n(x_q + f(t_{n+1}, x_q, u)\Delta t) + \ell(x_q, u)\Delta t\}\phi_j(x_q), \end{aligned}$$

where in the last step the approximations of the scheme (CIR) were used, i.e.,

$$x = y_x^u(t_{n+1}) \approx y_x(t_n) - f(t_{n+1}, x, u)\Delta t \implies y_x(t_n) \approx x + f(t_{n+1}, x, u)\Delta t,$$
$$\int_{t_n}^{t_{n+1}} \ell(y_x^u(s), u(s))\,\mathrm{d}s \approx \ell(y_x^u(t_{n+1}), u)\Delta t = \ell(x, u)\Delta t,$$

together with the simplification of a piecewise constant control,

$$\mathcal{U}_{\mathrm{ad}}^{\Delta t} \approx \{u\colon [t_n, t_{n+1}] \to U,\ \text{constant}\}.$$

In the proposed algorithm, the optimization problem is solved numerically by choosing N_a controls from U, $\{u_a\}_{a=1}^{N_a}$, and calculating the minimum among these values, explicitly

$$\inf_{u\in U}\{f_{\mathrm{obj}}(u)\} \approx \min_{a\in\{1,\cdots,N_a\}}\{f_{\mathrm{obj}}(u_a)\} = f_{\mathrm{obj}}(u_a^*).$$

Thus, where in the last step, the decomposition of the solution V_h^n as a linear combination of the $r+1$ basis functions on the interval $I_{q'} \in \mathcal{T}_h$ is implied, i.e.,

$$V_h^n(x_q + f(t_{n+1}, x_q, u)\Delta t) = \sum_{\ell=1}^{r+1} V_h^n|_{I_{q'}}(\xi_\ell)\phi_\ell(x_q + f(t_{n+1}, x_q, u)\Delta t)$$

with $q'(y_{x_q}(t_n)) = \{i \in \{1, \cdots, N\}\colon y_{x_q}(t_n) \in I_i\}$.

In summary, to approximate (HJB), the proposed DGSL-$\mathbb{P}_r$ method has the form where

$$[\cdots] = \min_{a\in\{1,\cdots,N_a\}}\left\{\sum_{\ell=1}^{r+1} V_h^n|_{I_{q'}}(\xi_\ell)\phi_\ell(x_q + f(t_{n+1}, x_q, u_a)\Delta t) + \ell(x_q, u_a)\Delta t\right\}.$$

Remark 1 (The DGSL-$\mathbb{P}_0$ case) It is worth emphasizing that, in the piecewise constant case (i.e., when $r = 0$), the DGSL scheme reduces to the classical SL method equipped with a piecewise continuous interpolator. In this setting, the standard theoretical framework applies [12], and the convergence of the scheme follows from the customary monotonicity arguments.

3 Numerical Simulations

To assess the effectiveness and accuracy of the DGSL-$\mathbb{P}_r$ method in solving HJB equations, some benchmark problems from the scientific literature were chosen [5, 16, 18]. These are specific problems where the viscosity solution has an exact known expression but is numerically difficult to capture.

3.1 Simulation Setup

The numerical experiments were conducted using the MATLAB software [14], implementing the DGSL-$\mathbb{P}_r$ algorithm to solve (HJB). With reference to this equation, the problem is characterized by prescribing the domain boundaries a and b, the final time T, the functions $\ell(x, u)$ and $f(t, x, u)$, the initial condition $L(x)$, and the extremes of the control value space $U = [u_{\mathrm{m}}, u_{\mathrm{M}}]$. When the boundary conditions are not periodic but Dirichlet, it is necessary to also specify this information along with the boundary functions $B_a(t)$ and $B_b(t)$.

To proceed with the simulation, the following are specified:

- the number N of disjoint and equispaced intervals $\{I_i\}_{i=1}^N$ to divide the spatial domain;

Table 1 Coordinates of the nodes and their corresponding weights of the 5-point Gauss-Legendre quadrature on the reference interval $[-1, 1]$

q	1	2	3	4	5
x_q	$-\frac{\sqrt{5+2\sqrt{\frac{10}{7}}}}{3}$	$-\frac{\sqrt{5-2\sqrt{\frac{10}{7}}}}{3}$	0	$\frac{\sqrt{5-2\sqrt{\frac{10}{7}}}}{3}$	$\frac{\sqrt{5+2\sqrt{\frac{10}{7}}}}{3}$
ω_q	$\frac{322-13\sqrt{70}}{900}$	$\frac{322+13\sqrt{70}}{900}$	$\frac{128}{225}$	$\frac{322+13\sqrt{70}}{900}$	$\frac{322-13\sqrt{70}}{900}$

- the number M of uniform time steps to divide the interval $(0, T]$;
- the maximum degree r of the polynomials defined on each interval of the *mesh* to reconstruct the numerical solution;
- the number N_a of equidistant control values between u_{m} and u_{M}, inclusive, to approximate the control space U.

In the following, unless otherwise specified, we set $N_a = 21$. For the calculation of the integrals required by the DGSL-$\mathbb{P}_r$ method, we use the Gauss-Legendre method, as it guarantees the highest accuracy and the lowest computational cost for the same number of nodes. In particular, the quadrature formula considered has $N_q = 5$ nodes, thus providing polynomial accuracy up to degree $2N_q + 1 = 11$. In Table 1, we report the values of the nodes x_q, and the weights w_q on the reference interval $[-1, 1]$.

The performance of the proposed numerical method was evaluated by analyzing the discretization (spatial) error e_h, which represents the deviation at a certain time instant of the approximate solution V_h from the exact solution V. This quantity, measured in the L^2, L^1, and L^∞ norms, is computed as follows:

$$\|e_h\|_{L^2} = \|V - V_h\|_{L^2([a,b])} = \sqrt{\sum_{i=1}^{N} \int_{I_i} (V(x) - V_h(x))^2 \,\mathrm{d}x},$$

$$\|e_h\|_{L^1} = \|V - V_h\|_{L^1([a,b])} = \sum_{i=1}^{N} \int_{I_i} |V(x) - V_h(x)| \,\mathrm{d}x,$$

$$\|e_h\|_{L^\infty} = \|V - V_h\|_{L^\infty([a,b])} = \sup_{i\in\{1,\cdots,N\}} \sup_{x\in I_i} |V(x) - V_h(x)|.$$

Furthermore, in the following sections, the error made by the proposed scheme is compared with some DG methods from the literature, particularly with respect to the Courant-Friedrichs-Lewy (CFL) number. This is a dimensionless parameter that represents a necessary condition for the stability of an explicit scheme in the context of hyperbolic problems with initial conditions [20, Observation 13.2]. The typical requirement for nonlinear finite differences is

$$\mathrm{CFL} = \frac{\|F\|_{L^\infty} \Delta t}{\Delta x} \leqslant 1. \tag{7}$$

3.2 Test 1—Quadratic *H*, Given C^∞, Comparison with Previous Literature

The first test comes from the article [18, Example 4.4],

$$\begin{cases} V_t(t,x) + \frac{(V_x(t,x))^2}{2} = 0, & (t,x) \in (0, 1.5] \times [0, 2\pi], \\ V(0,x) = -\cos(x), & x \in [0, 2\pi]. \end{cases} \tag{8}$$

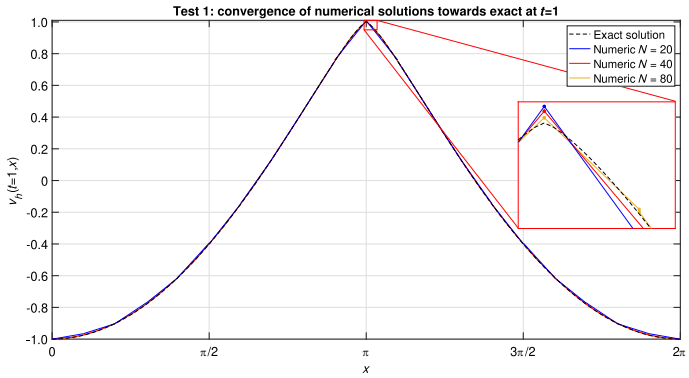

Fig. 5 Zoomed-in view of the comparison between DGSL-$\mathbb{P}_1$ numerical solutions ($N = 20, N = 40, N = 80$) and the exact solution of the problem (T1)

This is an HJ evolution equation (HJ) with $H(t, x, p) = p^2/2$, a C^∞ class initial condition, and periodic boundary conditions, $V(t, 0) = V(t, 2\pi)$ for every $t \in (0, 1.5]$.

The test (8) is equivalent to a problem (HJB), in fact, let $\ell(x, u) = u^2/2$, $F(t, x, u) = u$, and $L(x) = -\cos(x)$,

$$\begin{cases} V_t(t,x) - \inf\limits_{u\in U}\left\{\frac{u^2}{2} + uV_x(t,x)\right\} = 0, & (t,x) \in (0,1.5]\times[0,2\pi], \\ V(0,x) = -\cos(x), & x \in [0,2\pi]. \end{cases} \tag{T1}$$

Fixing (t, x), the function to be optimized is a convex parabola in u and the domain U is compact, therefore the minimum exists,

$$f_{\text{obj}}(u) = \tfrac{u^2}{2} + uV_x(t,x), \quad f'_{\text{obj}}(u) = u + V_x(t,x) = 0 \implies u^* = -V_x(t,x),$$
$$f_{\text{obj}}(u^*) = \tfrac{(-V_x(t,x))^2}{2} + (-V_x(t,x))V_x(t,x) = -\tfrac{(V_x(t,x))^2}{2},$$

thus

$$V_t(t,x) - \inf_{u\in U}\left\{\frac{u^2}{2} + uV_x(t,x)\right\} = V_t(t,x) - f_{\text{obj}}(u^*) = V_t(t,x) + \frac{(V_x(t,x))^2}{2}.$$

However, to ensure the equivalence between the two formulations, it is important to verify that $V_x(t, x) \in U$ for every $(t, x) \in (0, 1.5] \times [0, 2\pi]$, in this case it is sufficient to have $U = [-1, 1]$.

For $t \in (0, 1]$, the viscosity solution is obtained by calculating

$$y + \cos(y)t = x - \frac{\pi}{2}, \quad V(t,x) = \sin(y) + \frac{\cos^2(y)}{2}t.$$

At $t = 1$, a singularity forms in the derivative due to a shock wave, which makes the exact solution simply continuous.

In Fig. 5, a comparison is shown, at $t = 1$, between the exact solution (in black) and its approximation (in blue) obtained with the DGSL-$\mathbb{P}_1$ method. The considered mesh consists of $N = 320$ disjoint and equidistant intervals, and $M = 13$ uniform time steps were performed, so CFL ≈ 4.

Graphically, the proposed scheme appears to have correctly solved the problem, as well as the sharp point at $x = \pi$.

Table 2 Error committed in the region $[0, 2\pi]\backslash(3, 2\pi - 3)$ by the DGSL-$\mathbb{P}_1$ method for test (T1) at $t = 1$ with CFL ≈ 0.45

N	10	20	40	80	160	Order
$\|\|e_h\|\|_{L^2}$	4.701 3E−02	2.009 8E−02	9.206 1E−03	4.193 3E−03	2.377 6E−03	1.087 2
$\|\|e_h\|\|_{L^1}$	8.873 5E−02	3.264 5E−02	1.490 3E−02	6.318 1E−03	3.857 8E−03	1.141 7

Table 3 Error committed in the region $[0, 2\pi]\backslash(3, 2\pi - 3)$ by the DGSL-$\mathbb{P}_1$ method for test (T1) at $t = 1$ with CFL ≈ 2 and $N_a = 41$

N	10	20	40	80	160	Order
$\|\|e_h\|\|_{L^2}$	5.409 2E−02	1.241 9E−02	3.234 7E−03	9.638 5E−04	4.238 8E−04	1.767 9
$\|\|e_h\|\|_{L^1}$	1.036 9E−01	2.404 8E−02	6.597 4E−03	1.938 4E−03	8.616 6E−04	1.745 5

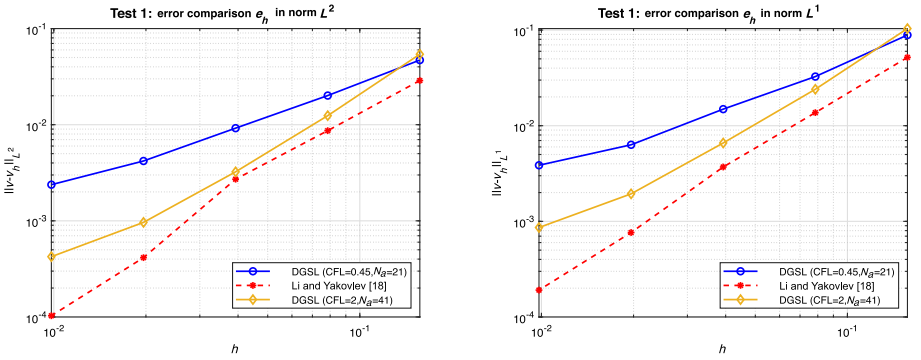

Fig. 6 Error trend in L^2 (left) and L^1 (right) norm as h increases for the DGSL-$\mathbb{P}_1$ numerical solution (in blue and yellow) of problem (T1), compared with the results from the article [18, Table 4] (in red)

In Table 2, the error $e_h = V - V_h$ made at $t = 1$, with CFL ≈ 0.45, is reported over the entire spatial domain excluding the open interval $(3, 2\pi - 3)$, in order to compare the results with [18, Table 4], at the same spatial and temporal refinement. The proposed scheme is less accurate than the *Central DG* method developed by Li and Yakovlev [18] and shows a lower rate of error convergence in both norms, L^2 and L^1. However, SL methods, including the one proposed, are known to be stable independently of the condition (7). In fact, with $N = 160$ and CFL ≈ 2, the *Central DG* method becomes completely unstable and the computed solution diverges, while DGSL-$\mathbb{P}_1$ converges, and the error data e_h are collected in Table 3, where $N_a = 41$ was also set.

In Fig. 6, the data of the proposed scheme are synthetically represented and compared with the Li and Yakovlev method, on the error committed in solving problem (8), in the L^2 and L^1 norms, respectively. In particular, it is clear that, by appropriately adjusting the CFL value and increasing the number N_a of controls to evaluate, the DGSL-$\mathbb{P}_1$ method becomes comparable to *Central DG* both in terms of accuracy and convergence rate.

3.3 Test 2—Non-differentiable Linear *H*, Given C^∞, Comparison with Previous Literature and High-Order Schemes

The second test is derived from the article [5, Example 4.2.1],

$$\begin{cases} V_t(t,x) + \text{sign}(\cos(x))V_x(t,x) = 0, & (t,x) \in (0,1] \times [0,2\pi], \\ V(0,x) = \sin(x), & x \in [0,2\pi]. \end{cases} \tag{9}$$

This is an evolution HJ equation with $H(t,x,p) = \text{sign}(\cos(x))p$, which is linear in p but with a coefficient exhibiting jump discontinuities at $x = \pi/2$ and $x = 3\pi/2$, causing the formation of a shock wave and a rarefaction wave, respectively, for V_x. The initial condition is of class C^∞ and periodic boundary conditions are imposed, $V(t,0) = V(t,2\pi)$ for all $t \in (0,1]$.

As reported in the article [16, Example 4.4], the solution to this problem coincides with the solution of the homogeneous eikonal evolution equation, that is

$$\begin{cases} V_t(t,x) + |V_x(t,x)| = 0, & (t,x) \in (0,1] \times [0,2\pi], \\ V(0,x) = \sin(x), & x \in [0,2\pi]. \end{cases}$$

Moreover, this formulation is equivalent to the problem (HJB), where $\ell(x,u) = 0$, $F(t,x,u) = u$, $U = [-1,1]$, and $L(x) = \sin(x)$,

$$\begin{cases} V_t(t,x) - \inf\limits_{u \in U}\{uV_x(t,x)\} = 0, & (t,x) \in (0,1] \times [0,2\,\pi], \\ V(0,x) = \sin(x), & x \in [0,2\,\pi]. \end{cases} \tag{T2}$$

Fixing (t,x), the function to optimize is a line in u, and the domain U is compact, so the minimum exists. Let $f_{\text{obj}}(u) = uV_x(t,x)$. Then two cases arise:

$$\begin{cases} V_x(t,x) > 0 \implies u^* = -1 \implies f_{\text{obj}}(u^*) = -V_x(t,x), \\ V_x(t,x) \leqslant 0 \implies u^* = 1 \implies f_{\text{obj}}(u^*) = V_x(t,x), \end{cases}$$

in summary

$$u^* = -\,\text{sign}(V_x(t,x)) \implies f_{\text{obj}}(u^*) = -|V_x(t,x)|,$$

therefore,

$$V_t(t,x) - \inf_{u \in U}\{uV_x(t,x)\} = V_t(t,x) - f_{\text{obj}}(u^*) = V_t(t,x) + |V_x(t,x)|.$$

The viscosity solution of the problem for $t \in (0, \pi/2]$ is

$$V(t,x) = \begin{cases} \sin(x-t), & x \in [0, \pi/2], \\ \sin(x+t), & x \in (\pi\,/2, 3\pi/2 - t], \\ -1, & x \in (3\pi/2 - t, 3\pi/2 + t], \\ \sin(x-t), & x \in (3\pi/2 + t, 2\pi]. \end{cases}$$

Figure 7 shows a comparison at $t = 1$ between the exact solution (in black) and its approximation (in blue) calculated using the DGSL-$\mathbb{P}_1$ method. The mesh considered consists of $N = 640$ disjoint and equidistant intervals, with $M = 25$ uniform time steps, giving CFL ≈ 4.

Graphically, the approximation provided by the proposed scheme for solving the problem appears correct, particularly near the corner point $x = \pi/2$ and in the rarefaction region.

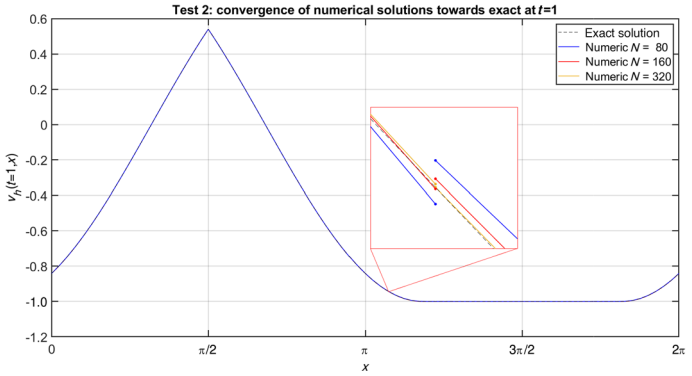

Fig. 7 Zoomed-in view of the comparison between DGSL-$\mathbb{P}_1$ numerical solutions ($N = 80$, $N = 160$, $N = 320$) and the exact solution of the problem (T2)

Table 4 Error committed by the DGSL-$\mathbb{P}_1$ method for test (T2) at $t = 1$ with CFL ≈ 0.1

N	40	80	160	320	640	Order
$\|\|e_h\|\|_{L^2}$	7.010 8E−02	2.270 9E−02	6.377 8E−03	1.797 5E−03	5.775 4E−04	1.750 6
$\|\|e_h\|\|_{L^1}$	1.102 6E−01	3.721 8E−02	1.157 7E−02	3.707 1E−03	1.302 3E−03	1.613 5
$\|\|e_h\|\|_{L^\infty}$	6.114 1E−02	1.832 2E−02	4.692 7E−03	1.174 1E−03	3.257 2E−04	1.906 9

Table 5 Error committed by the DGSL-$\mathbb{P}_1$ method for test (T2) at $t = 1$ with CFL ≈ 3

N	40	80	160	320	640	Order
$\|\|e_h\|\|_{L^2}$	3.054 2E−03	6.886 9E−04	2.118 0E−04	4.604 1E−05	1.156 2E−05	1.999 3
$\|\|e_h\|\|_{L^1}$	5.747 8E−03	1.400 8E−03	4.244 5E−04	8.186 5E−05	2.082 3E−05	2.031 4
$\|\|e_h\|\|_{L^\infty}$	2.998 4E−03	6.349 3E−04	2.388 3E−04	4.723 3E−05	1.125 3E−05	1.986 4

Table 4 shows the error $e_h = V - V_h$ at time $t = 1$, considering CFL ≈ 0.1, to compare the results with [5, Table 4.7], at the same spatial and temporal refinement.

The proposed scheme is less accurate than the method developed by Cheng and Shu [5], and the order of convergence of the error is slightly lower in all considered norms, L^2, L^1, and L^∞. However, it should be noted that, in this case, the approach developed by Cheng and Shu requires an additional procedure to correct and redirect the numerical solution to the viscosity solution. Furthermore, the CFL parameter plays a fundamental role in the stability of this numerical method, which can be verified by attempting to solve the problem (9) with $N = 160$ and CFL > 0.6, in which case the obtained solution is strongly divergent.

In contrast, a high CFL parameter does not limit the stability of the proposed method and can improve its accuracy. In Table 5, the results are reported for the approximation obtained by the DGSL-$\mathbb{P}_1$ method with CFL ≈ 3.

In Fig. 8, the data are synthetically represented, within the same graph, comparing the proposed scheme and the method by Cheng and Shu, regarding the error committed in solving the test example, respectively, in L^2 and L^∞ norms. From the images, it can be seen that by properly adjusting the CFL value, the DGSL-$\mathbb{P}_1$ scheme becomes comparable to the

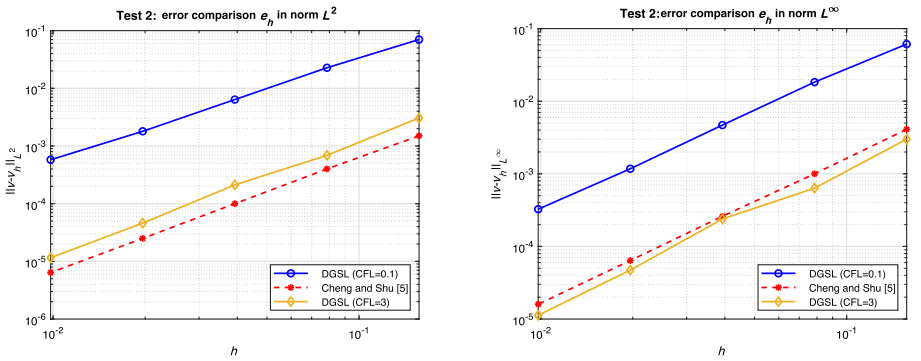

Fig. 8 Behavior of the error in the L^2 (left) and L^∞ (right) norm as h increases for the DGSL-$\mathbb{P}_1$ numerical solution (in blue and yellow) of the problem (T2), compared with the results from [5, Table 4.7]

Table 6 Error in L^2 norm by the DGSL-$\mathbb{P}_r$ method with various values of r for test (T2) at $t = 1$ with CFL ≈ 3

N	40	80	160	320	640	Order
$r = 1$	3.275 8E−03	1.157 7E−03	6.772 8E−04	4.579 7E−05	1.151 5E−05	2.096 5
$r = 2$	1.454 4E−04	1.331 5E−04	5.435 8E−05	7.416 3E−08	2.188 2E−08	3.620 7
$r = 3$	9.465 4E−05	4.320 3E−05	1.903 9E−05	2.975 2E−08	1.935 4E−08	3.501 5

Table 7 Error in the L^2 norm on the interval $[0, 2\pi] \setminus (\pi/2 - 2\Delta x, \pi/2 + \Delta x)$ (smooth regions of the solution) by the DGSL-$\mathbb{P}_r$ method with various values of r for test (T2) at $t = 1$ with CFL ≈ 3

N	40	80	160	320	640	Order
$r = 1$	3.054 2E−03	6.886 9E−04	2.118 0E−04	4.604 1E−05	1.156 2E−05	1.999 3
$r = 2$	2.978 1E−05	3.618 0E−06	7.751 1E−07	5.541 2E−08	7.721 1E−09	2.985 5
$r = 3$	1.238 7E−07	5.659 1E−09	3.556 5E−10	1.214 8E−11	3.512 1E−12	3.985 6

Cheng and Shu method both in terms of accuracy and convergence speed, or slightly better in terms of the L^∞ norm of the error.

We now proceed to compare the performance of the DGSL scheme at higher orders, namely for $r = 1, 2, 3$. From the previous tests, we have already observed that the error produced by the method exhibits a similar behavior across the standard norms L^2, L^1, and L^∞. Therefore, we restrict the following analysis to the L^2 norm.

Table 6 reports the error values for the DGSL-$\mathbb{P}_r$ scheme with $r = 1, 2, 3$. It can be observed that, although the method converges in all three cases, the convergence rates are not fully satisfactory: the convergence is non-uniform, and no significant improvement is detected when moving from $r = 2$ to $r = 3$. This behavior is due to the presence of a singularity in the solution.

When the analysis is restricted to the smooth regions of the solution, specifically, to the interval $[0, 2\pi]\setminus(\pi/2 - 2\Delta x,\ \pi/2 + 2\Delta x)$, the convergence rates become more regular, exhibiting an observed order of accuracy close to $r + 1$ (see Table 7). These results are also illustrated in Fig. 9, where the different performances of the high-order DGSL schemes can be appreciated both over the entire domain and within the smooth portions of the solution.

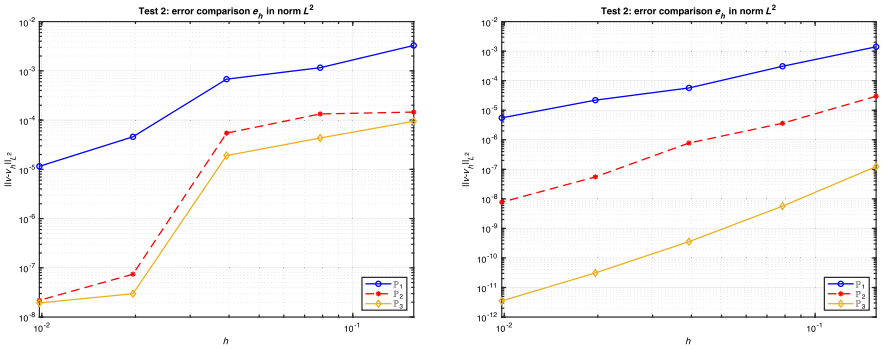

Fig. 9 Behavior of the error in the L^2 in the whole domain (left) and in the smooth regions of the solution (right) as h increases for the DGSL-$\mathbb{P}_r$ numerical solution

3.4 Test 3—Quadratic *H*, Non-differentiable Data, Comparison with Previous Literature

The third test is taken from the article [16, Example 4.3],

$$\begin{cases} V_t(t,x) + \frac{(V_x(t,x))^2}{2} = 0, & (t,x) \in (0,1] \times [0,2\pi], \\ V(0,x) = |x-\pi|, & x \in [0,2\pi]. \end{cases} \tag{10}$$

This is an evolutionary (HJ) with $H(t,x,p) = p^2/2$, a continuous initial condition, not differentiable at $x = \pi$, which leads to the formation of a rarefaction wave for V_x. There are also periodic boundary conditions, $V(t,0) = V(t,2\pi)$ for all $t \in (0,1]$.

As seen in the first test (Sect. 3.2), this problem can be rewritten in the form (HJB), considering $\ell(x,u) = u^2/2$, $F(t,x,u) = u$, $L(x) = |x-\pi|$, and $U = [-1,1]$,

$$\begin{cases} V_t(t,x) - \inf\limits_{u \in U} \left\{ \frac{u^2}{2} + u V_x(t,x) \right\} = 0, & (t,x) \in (0,1] \times [0,2\pi], \\ V(0,x) = |x-\pi|, & x \in [0,2\pi]. \end{cases} \tag{T3}$$

The viscosity solution of the problem, for $t \in (0,\pi]$, is

$$V(t,x) = \begin{cases} \pi - x - \frac{t}{2}, & x \in [0, \pi - t], \\ x - \pi - \frac{t}{2}, & x \in [\pi + t, 2\pi], \\ \frac{(x-\pi)^2}{2t}, & x \in (\pi - t, \pi + t). \end{cases}$$

Figure 10 shows the comparison, at $t = 1$, between the exact solution (in black) and its approximation (in blue) calculated with the DGSL-$\mathbb{P}_1$ method. The mesh used consists of $N = 160$ disjoint and equally spaced intervals, and $M = 25$ uniform time steps were performed, so CFL ≈ 1.

Graphically, the approximation obtained using the proposed method appears correct, particularly around the point $x = \pi$ where the rarefaction region forms.

Even if the proposed scheme is less accurate for small CFL values than, for example, the alternative DG formulation developed by Ke and Guo [16], our scheme has the freedom of allowing big time steps and as a consequence, high CFL values (i.e., that violate the condition (7)). In fact, Table 8 shows the data for the approximation of problem (T3) with CFL ≈ 4.

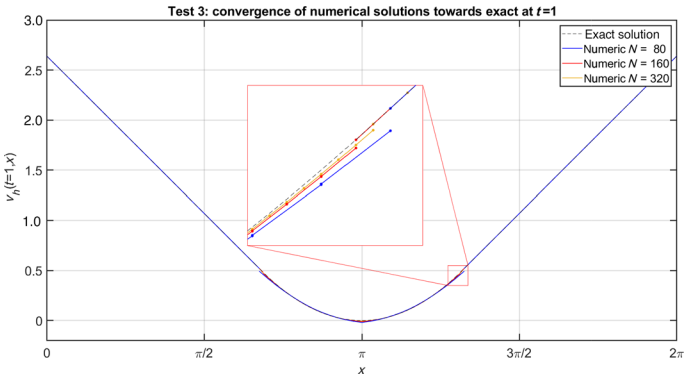

Fig. 10 Zoomed-in view of the comparison between DGSL-$\mathbb{P}_1$ numerical solutions ($N = 80$, $N = 160$, $N = 320$) and the exact solution of the problem (T3)

Table 8 Error committed by the DGSL-$\mathbb{P}_1$ method for test (T3) at $t = 1$ with CFL ≈ 4

N	40	80	160	320	640	Order
$\|\|e_h\|\|_{L^2}$	1.648 7E−02	9.449 0E−03	5.004 4E−03	2.500 7E−03	1.829 3E−03	0.826 2
$\|\|e_h\|\|_{L^\infty}$	1.772 7E−02	1.065 1E−02	5.887 5E−03	3.080 1E−03	2.174 3E−03	0.784 5

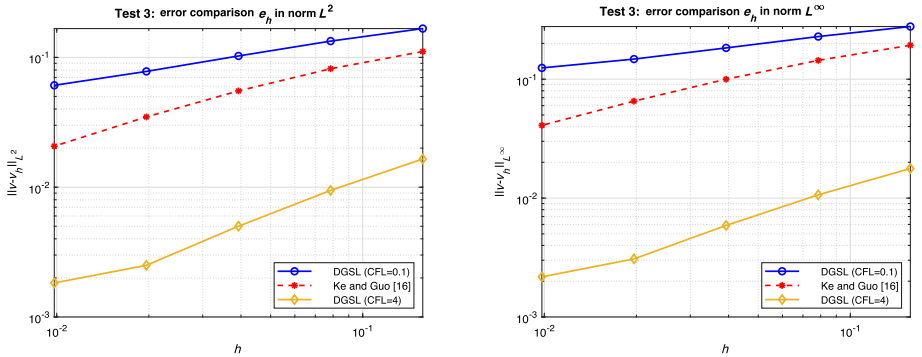

Fig. 11 Trend of the L^2 (left) and L^∞ (right) error as h increases for the DGSL-$\mathbb{P}_1$ numerical solution (in blue and yellow) of problem (T3), compared with the results from the article [16, Table 4, $\mathbb{P}_1$] (in red)

Figure 11 shows the comparison, in the same plot, between the data from the proposed scheme and Ke and Guo's method, for the error in solving the test example, in L^2 and L^∞ norms, respectively. From the images, it can be seen that adjusting the CFL value alone improves the performance of the DGSL-$\mathbb{P}_1$ scheme, which then outperforms the method of Ke and Guo [16] in terms of accuracy and error convergence speed.

4 Conclusions

In this work, a numerical method, DGSL-$\mathbb{P}_r$, has been proposed to solve 1D evolutionary HJB equations.

In light of the tests carried out, the approach based on the SL representation formula and the reconstruction of the function with Discontinuous Finite Elements has proven effective in capturing the rich structure (generated by discontinuities in the derivative and non-differentiable data everywhere) of the viscosity solutions. With reference to the objectives set in the Introduction, the experiments show the stability of the DGSL-$\mathbb{P}_r$ method regardless of the chosen time and space step size. Furthermore, as anticipated, the structure of the proposed scheme ensures both hp adaptivity and the decoupling of the linear system of equations into smaller parts, which can be solved separately.

A peculiar aspect that has emerged from the numerical experiments concerns accuracy and convergence order. Indeed, when compared with other methods known in the literature, the proposed scheme proved to be less effective for CFL values lower than one, but became comparable or even superior for larger time steps.

Building on the work discussed in this article, several future research directions open up. First, the study could be extended to the two-dimensional (or multidimensional) case, with attention given to the reconstruction of the characteristic in the SL phase of the method. Moreover, to reduce computational times, it may be reasonable to implement Legendre basis functions for each element, as continuity at the interface between elements is not required. Additionally, the DGSL-$\mathbb{P}_r$ method could be directed toward applications, with the appropriate corrections, involving viscosity solutions that develop jump-type discontinuities in finite time. Finally, it could be interesting to investigate further the connection between the error behavior and the CFL number, deriving some theoretical estimates on the discretization error and the convergence order of the method in the regions of the domain where the solution is smooth.

Acknowledgements AF would like to thank his former Ph.D. supervisor, Prof. Maurizio Falcone, to whom this work is dedicated.

Funding Open access funding provided by Politecnico di Torino within the CRUI-CARE Agreement. This work has been partially supported by the Gruppo Nazionale per il Calcolo Scientifico' (GNCS-INdAM) and by the PRIN project 2022 (No. 2022238YY5) "Optimal control problems: analysis, approximation".

Data Availability Data will be made available on reasonable request.

Declarations

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Bardi, M., Capuzzo-Dolcetta, I.: Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations. Systems & Control: Foundations & Applications. Birkhäuser Boston, Inc., Boston (1997). https://doi.org/10.1007/978-0-8176-4755-1 (**With appendices by Maurizio Falcone and Pierpaolo Soravia**)

2. Bokanowski, O., Simarmata, G.: Semi-Lagrangian discontinuous Galerkin schemes for some first- and second-order partial differential equations. ESAIM Math. Model. Numer. Anal. **50**(6), 1699–1730 (2016). https://doi.org/10.1051/m2an/2016004
3. Bryson, S., Levy, D.: High-order semi-discrete central-upwind schemes for multi-dimensional Hamilton-Jacobi equations. J. Comput. Phys. **189**(1), 63–87 (2003). https://doi.org/10.1016/S0021-9991(03)00201-8
4. Carlini, E., Ferretti, R., Russo, G.: A weighted essentially nonoscillatory, large time-step scheme for Hamilton-Jacobi equations. SIAM J. Sci. Comput. **27**(3), 1071–1091 (2005). https://doi.org/10.1137/040608787
5. Cheng, Y., Shu, C.-W.: A discontinuous Galerkin finite element method for directly solving the Hamilton-Jacobi equations. J. Comput. Phys. **223**(1), 398–415 (2007). https://doi.org/10.1016/j.jcp.2006.09.012
6. Cheng, Y., Wang, Z.: A new discontinuous Galerkin finite element method for directly solving the Hamilton-Jacobi equations. J. Comput. Phys. **268**, 134–153 (2014). https://doi.org/10.1016/j.jcp.2014.02.041
7. Courant, R., Isaacson, E., Rees, M.: On the solution of nonlinear hyperbolic differential equations by finite differences. Commun. Pure Appl. Math. **5**, 243–255 (1952). https://doi.org/10.1002/cpa.3160050303
8. Crandall, M.G., Evans, L.C., Lions, P.-L.: Some properties of viscosity solutions of Hamilton-Jacobi equations. Trans. Am. Math. Soc. **282**(2), 487–502 (1984). https://doi.org/10.2307/1999247
9. Crandall, M.G., Lions, P.-L.: Viscosity solutions of Hamilton-Jacobi equations. Trans. Am. Math. Soc. **277**(1), 1–42 (1983). https://doi.org/10.2307/1999343
10. Crandall, M.G., Lions, P.-L.: Two approximations of solutions of Hamilton-Jacobi equations. Math. Comput. **43**(167), 1–19 (1984). https://doi.org/10.2307/2007396
11. Evans, L.C.: Partial Differential Equations. Graduate Studies in Mathematics, vol. 19, 2nd edn., p. 749. American Mathematical Society, Providence (2010). https://doi.org/10.1090/gsm/019
12. Falcone, M., Ferretti, R.: Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations. Society for Industrial and Applied Mathematics, Philadelphia (2013). https://doi.org/10.1137/1.9781611973051
13. Hu, C., Shu, C.-W.: A discontinuous Galerkin finite element method for Hamilton-Jacobi equations. SIAM J. Sci. Comput. **21**(2), 666–690 (1999). https://doi.org/10.1137/S1064827598337282
14. Inc., T.M.: MATLAB Version 23.2.0 (R2023b). The MathWorks Inc., Natick, Massachusetts, United States (2023). https://www.mathworks.com
15. Jiang, G.-S., Peng, D.: Weighted ENO schemes for Hamilton-Jacobi equations. SIAM J. Sci. Comput. **21**(6), 2126–2143 (2000). https://doi.org/10.1137/S10648275973245 5X
16. Ke, G., Guo, W.: An alternative formulation of discontinous Galerkin schemes for solving Hamilton-Jacobi equations. J. Sci. Comput. **78**(2), 1023–1044 (2019). https://doi.org/10.1007/s10915-018-0794-7
17. Kurganov, A., Noelle, S., Petrova, G.: Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton-Jacobi equations. SIAM J. Sci. Comput. **23**(3), 707–740 (2001). https://doi.org/10.1137/S1064827500373413
18. Li, F., Yakovlev, S.: A central discontinuous Galerkin method for Hamilton-Jacobi equations. J. Sci. Comput. **45**(1/2/3), 404–428 (2010). https://doi.org/10.1007/s10915-009-9340-y
19. Osher, S., Shu, C.-W.: High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations. SIAM J. Numer. Anal. **28**(4), 907–922 (1991). https://doi.org/10.1137/0728049
20. Quarteroni, A.: Modellistica Numerica per Problemi Differenziali, 6th edn. UNITEXT—La Matematica per il 3+2, vol. 100, p. 651. Springer, Milan (2016). https://doi.org/10.1007/978-88-470-5782-1
21. Yan, J., Osher, S.: A local discontinuous Galerkin method for directly solving Hamilton-Jacobi equations. J. Comput. Phys. **230**(1), 232–244 (2011). https://doi.org/10.1016/j.jcp.2010.09.022