

Primal-dual hybrid gradient algorithms for computing time-implicit Hamilton-Jacobi equations*

Tingwei Meng[†] Wenbo Hao^{†§} Siting Liu^{†§} Stanley J. Osher[†] Wuchen Li^{‡¶}

October 2, 2023

Abstract

Hamilton-Jacobi (HJ) partial differential equations (PDEs) have diverse applications spanning physics, optimal control, game theory, and imaging sciences. This research introduces a first-order optimization-based technique for HJ PDEs, which formulates the time-implicit update of HJ PDEs as saddle point problems. We remark that the saddle point formulation for HJ equations is aligned with the primal-dual formulation of optimal transport and potential mean-field games (MFGs). This connection enables us to extend MFG techniques and design numerical schemes for solving HJ PDEs. We employ the primal-dual hybrid gradient (PDHG) method to solve the saddle point problems, benefiting from the simple structures that enables fast computations in updates. Remarkably, the method caters to a broader range of Hamiltonians, encompassing non-smooth and spatiotemporally dependent cases. The approach's effectiveness is verified through various numerical examples in both one-dimensional and two-dimensional examples, such as quadratic and L^1 Hamiltonians with spatial and time dependence.

1 Introduction

Hamilton-Jacobi (HJ) partial differential equations (PDEs) find applications in various fields such as physics [5, 15, 16, 25, 69], optimal control [7, 37, 44, 45, 74], game theory [9, 14, 42, 58], imaging sciences [26, 29, 32, 31], and machine learning [21]. In existing literature, a variety of approaches have been explored to numerically address HJ PDEs. In lower dimensions, high-order grid-based techniques such as essentially nonoscillatory schemes (ENO) [80], weighted ENO scheme [61], and discontinuous Galerkin method [54] are commonly employed, while in higher dimensions, strategies have been proposed to manage the challenges arising from the curse of dimensionality. These works include, but are not limited to, max-plus algebra methods [74, 1, 2, 35, 43, 48, 75, 76, 77], dynamic programming and reinforcement learning [3, 10], tensor decomposition techniques [34, 53, 86], sparse grids [11, 47, 64], model order reduction [4, 66], polynomial approximation [62, 63], optimization methods [26, 29, 32, 88, 24, 22, 23] and neural networks [28, 6, 33, 60, 51, 56, 57, 68, 78, 82, 83, 85, 30, 27].

In this study, we introduce an innovative optimization-based methodology for tackling time-implicit computation of HJ PDEs. Our approach formulates the HJ PDE as a min-max problem by introducing a Lagrange multiplier. We then employ the primal-dual hybrid gradient (PDHG) method [20] for finding the saddle point, which corresponds to the HJ PDE solution. In particular, the saddle point formulation of the HJ equations connects to the primal-dual formulation of optimal transport and potential mean-field games. Mean-field games (MFGs), introduced in [70, 55] are a mathematical framework employed for modeling and analyzing the equilibrium state of strategic interactions within a large population. This framework has found widespread application in various domains [67, 50, 49, 18, 17, 65, 36, 71]. The MFG system can be described

*T. Meng, S. Liu, and S. J. Osher are partially funded by AFOSR MURI FA9550-18-502, ONR N00014-18-1-2527 and N00014-20-1-2787. W. Li's work is partially supported by AFOSR YIP award No. FA9550-23-1-008, NSF DMS-2245097, and NSF RTG: 2038080.

[†]Department of Mathematics, UCLA, Los Angeles, CA 90025, USA.

[‡]Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA.

[§]W.H. and S.L. contributed equally to this work.

[¶]Corresponding author (wuchen@mailbox.sc.edu).

using a system of coupled PDEs: a Fokker-Planck equation evolves forward in time, and an HJ equation evolves backward in time. In this study, we leverage this relationship to address HJ PDEs. Using this connection, techniques developed for solving MFGs can be seamlessly extended to tackle the computation of the solution to HJ PDEs with initial conditions. For instance, PDHG method has been applied to numerically solve potential MFGs that can be cast into a saddle point formulation [81, 13, 12]. We adapt such methodology to our proposed saddle point problem, which in turn enables us to effectively address the original HJ PDEs. When compared to other grid-based methods, our current approach, while possessing a first-order accuracy level, attains numerical unconditional stability through the utilization of implicit time discretization. This feature enables us to adopt larger time steps. Compared to other optimization-based methods, our technique boasts the capability to handle a broader range of Hamiltonian functions, including those that exhibit non-smooth behavior and dependence on both x and t . Furthermore, our algorithm benefits from its straightforward saddle point formulation, which allows simple updates in each iteration. We emphasize that the updates in our method do not involve nonlinear inversion. The only non-trivial update is managed through the proximal point operator of the Hamiltonian H , which can be computed in parallel.

The method of transforming a PDE problem into a saddle point problem and subsequently utilizing PDHG for its solution is used in solving reaction-diffusion equations [72, 46, 19] as well as conservation laws [73]. These works use PDHG together with integration by parts to solve initial value problems with or without constraints, obtaining simple implicit in time updates. It is well-known in the literature that one-dimensional HJ PDEs and conservation laws are equivalent to each other. However, applying the method used for solving conservation laws in [73] directly to solve HJ PDEs is not straightforward. In [73], the key step for obtaining a saddle point problem involves employing integration by parts. The method of integration by parts is applicable to conservation laws, as these equations entail gradients of flux functions, a feature that is absent in HJ PDEs. Nevertheless, in this study, for convex HJ PDEs, we instead use duality via the Fenchel-Legendre transform in lieu of integration by parts, effectively avoiding nonlinear updates. The merit of this approach lies in the simplicity of the saddle point formulation. This simplicity facilitates updates within our method to have either explicit formulations or be conducive to parallel computation. It is pertinent to note that while the formulation in this work may resemble the adjoint method [40], the proposed numerical technique and saddle point formulation differ substantially.

We show several numerical examples in one dimension and two dimensions¹. These numerical examples show the ability of this method to handle certain Hamiltonians which may depend on (x, t) . In each iteration, the updates of the functions are independent on each point, which makes it possible to use parallel computing to accelerate the algorithm. Moreover, in the special case when the Hamiltonian $H(x, t, \cdot)$ is a separable and shifted 1-homogeneous function for any (x, t) , the algorithm has a simpler form, and we obtain an explicit formula for the updates of the dual variables.

The paper is organized as follows. In Section 2, we show the saddle point problem related to the HJ PDE and the proposed algorithm in the function space. In Section 3, we focus on the one-dimensional case and show both semi-discrete and fully-discrete formulations of the algorithm. In Section 4, we show the algorithm in two-dimensional case. In Section 5, we provide several numerical results which demonstrate the ability of the algorithm to handle certain Hamiltonians which may depend on (x, t) . In Section 6, we show the summary and future work. More details about the proposed method and different saddle point formulations with the corresponding algorithms are also shown in the appendix.

2 The saddle point formulation of HJ PDEs

In this section, we give details on the derivation of the saddle point formulation of the HJ PDEs. Furthermore, we devise a primal-dual hybrid gradient algorithm that solves it.

¹Codes are available at [todo: github page].

2.1 The saddle point formulation

In this paper, we solve the following HJ PDE in the domain $\Omega \times [0, +\infty)$

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + H(x, t, \nabla_x \phi(x, t)) = \epsilon \Delta_x \phi(x, t), & x \in \Omega, t \in [0, T], \\ \phi(x, 0) = g(x), & x \in \Omega, \end{cases} \quad (1)$$

where $\Omega = \prod_{i=1}^n [a_i, b_i]$ is the spatial domain in \mathbb{R}^n with periodic boundary condition, and $\epsilon \geq 0$ is the diffusion parameter. Note that when $\epsilon = 0$, equation (1) is the first-order HJ PDE. In general, we assume that the Hamiltonian $H(x, t, p)$ is convex with respect to p for any $x \in \Omega, t \in [0, T]$.

To address these equations, we treat them as constraints within the following optimization problem

$$\min_{\phi \text{ satisfying (1)}} -c \int_{\Omega} \phi(x, T) dx, \quad (2)$$

where $c > 0$ is a hyperparameter. In our numerical experiments, we observe that the constant c can influence the convergence of our proposed algorithm. Moreover, it is possible to formulate more intricate objective functions in this optimization problem. The question of selecting a suitable objective function may pose an interesting problem that requires further investigation.

Subsequently, we introduce the Lagrange multiplier ρ , leading to the following computation

$$\begin{aligned} & \min_{\phi \text{ satisfying (1)}} -c \int_{\Omega} \phi(x, T) dx \\ = & \min_{\phi} \max_{\rho} \int_0^T \int_{\Omega} \rho(x, t) \left(\frac{\partial \phi(x, t)}{\partial t} - \epsilon \Delta_x \phi(x, t) + H(x, t, \nabla_x \phi(x, t)) \right) dx dt - c \int_{\Omega} \phi(x, T) dx \\ = & \min_{\phi} \max_{\rho} \int_0^T \int_{\Omega} \rho(x, t) \left(\frac{\partial \phi(x, t)}{\partial t} - \epsilon \Delta_x \phi(x, t) + \max_{v \in \mathbb{R}^n} \{ \langle v, \nabla_x \phi(x, t) \rangle - H^*(x, t, v) \} \right) dx dt - c \int_{\Omega} \phi(x, T) dx. \end{aligned} \quad (3)$$

If ρ is non-negative, we can move the maximization with respect to v outside of the integral and formulate the following saddle point problem.

$$\min_{\phi} \max_{\substack{\rho, v \\ \rho \geq 0}} \mathcal{L}(\phi, \rho, v), \quad (4)$$

where

$$\mathcal{L}(\phi, \rho, v) = \int_0^T \int_{\Omega} \rho(x, t) \left(\frac{\partial \phi(x, t)}{\partial t} + \langle v(x, t), \nabla_x \phi(x, t) \rangle - \epsilon \Delta_x \phi(x, t) - H^*(x, t, v(x, t)) \right) dx dt - \int_{\Omega} c \phi(x, T) dx. \quad (5)$$

Here, $H^*(x, t, \cdot)$ is the Fenchel-Legendre transform of $H(x, t, \cdot)$ for any $x \in \Omega$ and $t \in [0, T]$, defined by $H^*(x, t, v) = \sup_{p \in \mathbb{R}^n} \{ \langle v, p \rangle - H(x, t, p) \}$. For more discussion about this saddle point problem, see Appendix A.1.

Remark 2.1. *The first-order optimality condition of (4) gives*

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + H(x, t, \nabla_x \phi(x, t)) \leq \epsilon \Delta_x \phi(x, t), & x \in \Omega, t \in [0, T], \\ \frac{\partial \rho(x, t)}{\partial t} + \nabla_x \cdot (\rho(x, t) v(x, t)) + \epsilon \Delta_x \rho(x, t) = 0, & x \in \Omega, t \in [0, T], \\ v(x, t) = \nabla_p H(x, t, \nabla_x \phi(x, t)), & x \in \Omega, t \in [0, T], \\ \phi(x, 0) = g(x), \quad \rho(x, T) = c, & x \in \Omega, \\ \rho(x, t) \left(\frac{\partial \phi(x, t)}{\partial t} + H(x, t, \nabla_x \phi(x, t)) - \epsilon \Delta_x \phi(x, t) \right) = 0, & x \in \Omega, t \in [0, T]. \end{cases} \quad (6)$$

This coupled system of PDEs differs slightly from the HJ PDE (1), with the distinction that the first row entails an inequality rather than an equality. As indicated by the final row, if $\rho(x, t) \neq 0$ for all x, t , the

initial inequality in the first row transforms into an equality, thus yielding a solution to the HJ PDE (1). The discrepancy between the first row and the HJ PDE (1) is attributed to the imposition of a constraint $\rho \geq 0$ in the saddle point problem (4). For more details, see Appendix A.1.

The structure of this coupled system of PDEs closely resembles that of the coupled PDEs encountered in mean-field control problems. Essentially, when the density solution ρ associated with the corresponding mean-field control problem maintains a positive value throughout the entire domain, the PDE system (6) gives a solution to the HJ PDE (1). This encompasses scenarios where $\epsilon > 0$, as the diffusion term introduces the Brownian motion component within the underlying stochastic process, ensuring a nonzero density function ρ . In the case where ϵ is zero, although we cannot establish a theoretical assurance, we do provide numerical validation in Section 5.

2.2 PDHG algorithm

Within existing literature, a widely recognized approach for solving saddle point problems is PDHG algorithm [20]. In this section, we provide a brief review of this method. It solves saddle point problems in the form of

$$\min_{x \in X} \max_{y \in Y} \langle Kx, y \rangle + g(x) - f(y),$$

where f and g are convex functions, and K represents a linear operator. This algorithm is an iterative method, where in each iteration, the primal variable x and the dual variable y are updated separately using the proximal point operators of g and f . With given suitable stepsizes $\sigma, \tau > 0$, the m -th iteration update can be written as follows:

$$\begin{cases} x^m &= \arg \min_{x \in X} \langle x, K^T \bar{y}^{m-1} \rangle + g(x) + \frac{1}{2\sigma} \|x - x^{m-1}\|_X^2, \\ y^m &= \arg \max_{y \in Y} \langle Kx^m, y \rangle - f(y) - \frac{1}{2\tau} \|y - y^{m-1}\|_Y^2, \\ \bar{y}^m &= 2y^m - y^{m-1}. \end{cases}$$

The above calculation requires proximal gradient descent (ascent) steps of the primal (dual) variables. The computation for this method has also been expanded in scope by [87] to encompass more generalized problems where the operator K can be nonlinear.

2.3 PDHG algorithm for solving HJ PDEs

In this section, we apply PDHG algorithm to solve the saddle point problem (4). Owing to the simplicity of (5), the primal and dual updates either possess explicit formulas or can be computed in parallel using the proximal point operator of the function $H(x, t, \cdot)$. This makes PDHG well-suited for solving the saddle point problem presented in (4). Consequently, we employ this method, incorporating pre-conditioning on ϕ , to solve this specific saddle point problem. The outlined algorithm is presented in Algorithm 1. Further details about pre-conditioning and other techniques are summarized in Remark 2.2. It is important to note that, although we have a simple formulation and updating rule in our method, in most cases, an explicit formula for the joint update of ρ and v is unavailable. As a consequence, each iteration in our algorithm involves multiple coordinate updating steps for ρ and v independently.

Remark 2.2. *The value of the constant c does not impact the solution of the HJ PDE (1), yet we have observed its effects on the convergence during our numerical experiments. While we opt for a constant value of c in this paper, it's noteworthy that it could potentially take on any positive function. This choice serves as the terminal condition for ρ in (6). Further investigation is required to understand the influence of this terminal condition on the convergence of the proposed method and to determine the appropriate method for its selection.*

There are several strategies to expedite the convergence of the algorithm. One approach is to modify the penalty terms in the updates of ϕ, ρ, v . Specifically, for ϕ , selecting a penalty term such as $\|\nabla\phi - \nabla\phi^\ell\|^2$ can enhance computational efficiency (this technique is called pre-conditioning), as demonstrated in previous

Algorithm 1: The proposed algorithm for solving (4)

Inputs : Stepsize $\tau, \sigma > 0$, error tolerance $\delta > 0$, inner maximal iteration number N_{inner} and outer maximal iteration number N_{outer} .

Outputs: the solution to the corresponding HJ PDE (1).

1 Initialize the functions by $\phi^0(x, t) = g(x)$ for all $x \in \Omega$ and $t \in [0, T]$, $\rho^0 \equiv c$, $v^0 \equiv 0$.

2 **for** $\ell = 0, 1, \dots, N_{outer} - 1$ **do**

3 Update the function $\phi: \Omega \times [0, T] \rightarrow \mathbb{R}$ using

$$\phi^{\ell+1} = \arg \min_{\phi: \phi(x, 0) = g(x)} \mathcal{L}(\phi, \rho^\ell, v^\ell) + \frac{1}{2\tau} \|\nabla \phi - \nabla \phi^\ell\|^2 = \phi^\ell + \tau(-\Delta)^{-1}(\partial_t \rho^\ell + \nabla_x \cdot (v^\ell \rho^\ell) + \epsilon \Delta_x \rho^\ell), \quad (7)$$

where ∇, Δ are differential operators with respect to both x and t , and $(-\Delta)^{-1}f$ means the solution to $-\Delta u = f$ in $\Omega \times [0, T]$ with periodic spatial condition, Dirichlet initial condition $u(x, 0) = 0$, and Neumann terminal condition $\partial_t u(x, T) = 0$.

4 **if** $\|\partial_t \phi^{\ell+1}(x, t) + H(x, t, \nabla_x \phi^{\ell+1}(x, t)) - \epsilon \Delta_x \phi^{\ell+1}(x, t)\|_1 \leq \delta$ **then**

5 | Return $\phi^{\ell+1}$.

6 **end**

7 Set $\bar{\phi}^{\ell+1} = 2\phi^{\ell+1} - \phi^\ell$.

8 Set $v^{\ell+1, 0} = v^\ell$, $\rho^{\ell+1, 0} = \rho^\ell$.

9 **for** $m = 0, 1, \dots, N_{inner} - 1$ **do**

10 Update v by

$$\begin{aligned} v^{\ell+1, m+1} &= \arg \max_v \mathcal{L}(\bar{\phi}^{\ell+1}, \rho^{\ell+1, m}, v) - \frac{1}{2\sigma} \|\rho^{\ell+1, m}(v - v^{\ell+1, m})\|^2 \\ &= \arg \min_v \int_0^T \int_\Omega H^*(x, t, v(x, t)) + \frac{\rho^{\ell+1, m}(x, t)}{2\sigma} \left\| v(x, t) - v^{\ell+1, m}(x, t) - \sigma \frac{\nabla_x \bar{\phi}^{\ell+1}(x, t)}{\rho^{\ell+1, m}(x, t)} \right\|^2 dx dt. \end{aligned} \quad (8)$$

11 Update ρ by

$$\rho^{\ell+1, m+1} = \arg \max_{\rho: \rho \geq 0} \mathcal{L}(\bar{\phi}^{\ell+1}, \rho, v^{\ell+1, m+1}) - \frac{1}{2\sigma} \|\rho - \rho^{\ell+1, m}\|^2 = \max\{\mu^{\ell+1, m+1}, 0\}, \quad (9)$$

where $\mu^{\ell+1, m+1}$ is a function defined by $\mu^{\ell+1, m+1}(x, t) = \rho^{\ell+1, m}(x, t) + \sigma(\partial_t \bar{\phi}^{\ell+1}(x, t) + \langle v^{\ell+1, m+1}(x, t), \nabla_x \bar{\phi}^{\ell+1}(x, t) \rangle - H^*(x, t, v^{\ell+1, m+1}(x, t)) - \epsilon \Delta_x \bar{\phi}^{\ell+1}(x, t))$.

12 **end**

13 Set $v^{\ell+1} = v^{\ell+1, N_{inner}}$, $\rho^{\ell+1} = \rho^{\ell+1, N_{inner}}$.

14 **end**

15 Return $\phi^{N_{outer}}$.

studies [59, 73]. In our exploration, we also investigated different penalty expressions for ρ and v . Notably, a quadratic penalty demonstrated its effectiveness for ρ , while regarding v , we observed comparable performance between penalty terms such as $\|\rho^{\ell+1,m}(v - v^{\ell+1,m})\|^2$ and $\rho^{\ell+1,m}\|v - v^{\ell+1,m}\|^2$. These alternatives exhibited more favorable results compared to the quadratic penalty. In the subsequent sections of this paper, unless otherwise explicitly stated, we employ the notation $\|f\|$ to signify the L^2 norm when f represents a function, and to indicate the ℓ^2 norm if f is a finite-dimensional vector.

Furthermore, we adopted a time interval partitioning strategy, employing the proposed algorithm separately within each subdivided interval. Although this necessitates solving multiple saddle point problems, the resulting smaller dimensions of each problem substantially expedite the algorithm's runtime, as detailed in prior research (Algorithm 3 in [73]).

3 One-dimensional HJ PDEs

In this section, we focus on one-dimensional HJ PDEs, where the spatial domain Ω is defined as the interval $[a, b]$. We begin by introducing the semi-discrete method in Section 3.1, and then proceed to explain the fully-discrete method in Section 3.2. Throughout the rest of this paper, we employ the notation $(a_i)_i$, $(a_{i,j})_{i,j}$, and $(a_{i,j,k})_{i,j,k}$ to represent a vector, matrix, and tensor, respectively, where the elements are denoted by a_i , $a_{i,j}$, and $a_{i,j,k}$.

3.1 Semi-discrete formulation

To solve the continuous HJ PDE (1), our initial step involves discretizing the spatial domain $[a, b]$. We define x_i as the i -th grid point on a uniformly spaced grid within the interval $[a, b]$. Specifically, x_i is calculated as $a + \frac{(b-a)(i-1)}{n_x-1}$, where n_x represents the total number of grid points. The semi-discrete approach for a general numerical Hamiltonian \hat{H} is presented as follows:

$$\begin{cases} \dot{\phi}_i(t) + \hat{H}(x_i, t, (D_x^+ \phi)_i(t), (D_x^- \phi)_i(t)) = \epsilon(D_{xx}\phi)_i(t), & i = 1, \dots, n_x, t \in [0, T], \\ \phi_i(0) = g(x_i), & i = 1, \dots, n_x. \end{cases} \quad (10)$$

According to the theory of first order monotone scheme, the numerical Hamiltonian \hat{H} needs to be consistent (i.e., $\hat{H}(x, t, p, p) = H(x, t, p)$) and monotone (i.e., non-increasing with respect to p^+ and non-decreasing with respect to p^-). For more details, we refer readers to [84]. In this paper, we use D_x^+ and D_x^- to represent the right and left finite difference approximations of the spatial derivative, respectively. Specifically, $(D_x^+ \phi)_i(t)$ is computed as $\frac{\phi_{i+1}(t) - \phi_i(t)}{\Delta x}$, and $(D_x^- \phi)_i(t)$ is calculated as $\frac{\phi_i(t) - \phi_{i-1}(t)}{\Delta x}$. Additionally, D_{xx} approximates the Laplace operator, given by $(D_{xx}\phi)_i(t) = \frac{\phi_{i+1}(t) - 2\phi_i(t) + \phi_{i-1}(t)}{\Delta x^2}$. Here, we assume $\hat{H}(x_i, t, p^+, p^-)$ is jointly convex with respect to (p^+, p^-) for any $i = 1, \dots, n_x$ and $t \in [0, T]$.

Similar to the continuous version, we incorporate these formulas into the constraints of an optimization problem and introduce the Lagrange multiplier ρ . Consequently, this semi-discrete equation can be solved using the following saddle point formulation

$$\begin{aligned} \min_{\substack{\phi_i \forall i \\ \phi_i(0) = g(x_i)}} \max_{\substack{\rho_i, v_i^+, v_i^- \forall i \\ \rho_i \geq 0}} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) + v_i^+(t)(D_x^+ \phi)_i(t) + v_i^-(t)(D_x^- \phi)_i(t) - \epsilon(D_{xx}\phi)_i(t) \right. \\ \left. - \hat{H}^*(x_i, t, v_i^+(t), v_i^-(t)) \right) dt - c \sum_{i=1}^{n_x} \phi_i(T), \end{aligned} \quad (11)$$

where $c > 0$ is a hyper-parameter, and $\hat{H}^*(x_i, t, \cdot, \cdot)$ is the Fenchel-Legendre transform of $\hat{H}(x_i, t, \cdot, \cdot)$. The derivation of this saddle point formula is similar to the continuous case and is therefore omitted here. For more details, please refer to Appendix A.2.

To solve this saddle point problem, we apply the PDHG method. Generally, deriving a direct updating formula for the combined variables (ρ_i, v_i^+, v_i^-) is challenging. As a result, an inner loop is utilized in which each iteration involves the sequential updates of (v_i^+, v_i^-) and ρ_i . Let's denote the objective function in (11) as $\mathcal{L}_{semi}((\phi_i)_i, (\rho_i)_i, (v_i^+)_i, (v_i^-)_i)$. The proposed algorithm is summarized in Algorithm 2.

Remark 3.1. In practice, many numerical Hamiltonians \hat{H} are separable, i.e., \hat{H} satisfies $\hat{H}(x, t, p^+, p^-) = \hat{H}_1(x, t, p^+) + \hat{H}_2(x, t, p^-)$, where \hat{H}_1 is non-increasing in p^+ , and \hat{H}_2 is non-decreasing in p^- . In this scenario, the Fenchel-Legendre transform obeys the relationship $\hat{H}^*(x, t, v^+, v^-) = \hat{H}_1^*(x, t, v^+) + \hat{H}_2^*(x, t, v^-)$. As a result, the updates of v^+ and v^- can be computed separately and in parallel.

3.2 Fully-discrete formulation

We proceed by introducing a fully-discrete method through time discretization. In this paper, we advocate implicit time discretization to enable the use of larger time steps and circumvent the restrictive Courant–Friedrichs–Lewy (CFL) time step restriction for explicit methods. The time interval $[0, T]$ is discretized into $n_t - 1$ equi-spatial subintervals, each of length $\Delta t = \frac{T}{n_t - 1}$. Given a function f defined on $[a, b] \times [0, T]$, we denote $f_{i,k}$ as the function value at the grid point (x_i, t_k) .

We adopt the backward Euler scheme $(D_t^- \phi)_{i,k} = \frac{\phi_{i,k} - \phi_{i,k-1}}{\Delta t}$ to approximate the time derivative at (x_i, t_k) . Employing this discretization, the resulting numerical scheme is as follows:

$$\begin{cases} (D_t^- \phi)_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi)_{i,k}, (D_x^- \phi)_{i,k}) = \epsilon(D_{xx} \phi)_{i,k}, & i = 1, \dots, n_x, k = 2, \dots, n_t, \\ \phi_{i,1} = g(x_i), & i = 1, \dots, n_x, \end{cases} \quad (12)$$

and the saddle point problem becomes

$$\begin{aligned} \min_{\substack{\phi_{i,k} \forall i,k \\ \phi_{i,1} = g(x_i)}} \max_{\substack{\rho_{i,k}, v_{i,k}^+, v_{i,k}^- \forall i,k \\ \rho_{i,k} \geq 0}} & \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \rho_{i,k} \left((D_t^- \phi)_{i,k+1} + v_{i,k}^+ (D_x^+ \phi)_{i,k+1} + v_{i,k}^- (D_x^- \phi)_{i,k+1} - \epsilon(D_{xx} \phi)_{i,k+1} \right. \\ & \left. - \hat{H}^*(x_i, t_{k+1}, v_{i,k}^+, v_{i,k}^-) \right) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \phi_{i,n_t}. \end{aligned} \quad (13)$$

The set of equations in (12) constitutes a total of $n_x \times (n_t - 1)$ equations. As a result, the dual variables $\rho_{i,k}$, $v_{i,k}^+$, and $v_{i,k}^-$ possess indices ranging from $i = 1$ to n_x , and k ranges from 1 to $n_t - 1$. We apply PDHG to solve this saddle point problem, and the details are summarized in Algorithm 3 and Appendix A.3.

The objective function in (13) is linear when considering either ϕ or ρ . This linearity enables us to have explicit formulas for updating ϕ and ρ . When iteratively updating ϕ , we effectively address a discrete Poisson's equation within the temporal-spatial domain through the utilization of the Fourier transform, thus facilitating efficient computation. The only non-linear part in (13) pertains to updating (v^+, v^-) , involving solving the proximal point of $(v^+, v^-) \mapsto \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \hat{H}^*(x_i, t_{k+1}, v_{i,k}^+, v_{i,k}^-)$. The element at the (i, k) position of this proximal point corresponds to the proximal point of $\hat{H}^*(x_i, t_{k+1}, \cdot, \cdot)$, facilitating parallel updates for (v^+, v^-) . Furthermore, when \hat{H} takes the form $\hat{H}(x, t, p^+, p^-) = \hat{H}_1(x, t, p^+) + \hat{H}_2(x, t, p^-)$, the dual function simplifies to $\hat{H}^*(x, t, v^+, v^-) = \hat{H}_1^*(x, t, v^+) + \hat{H}_2^*(x, t, v^-)$, enabling further parallelization for v^+ and v^- .

In specific cases where the Hamiltonian H has particular structures, it becomes possible to update the variables (ρ, v^+, v^-) simultaneously, removing the need for the inner loop. An example of this occurs when $H(x, t, p)$ is convex, separable, and 1-homogeneous with respect to p . For more detailed information, please refer to Appendix B.2.

4 Two-dimensional HJ PDEs

In this section, we address the two-dimensional HJ PDEs and present the semi-discrete approach in Section 4.1 and the fully-discrete approach in Section 4.2. Due to the similarities between the two-dimensional and one-dimensional cases, certain details and explanations have been omitted.

4.1 Semi-discrete formulation

We apply discretization to the spatial domain $[a_1, b_1] \times [a_2, b_2]$ using n_x grid points in the first dimension and n_y grid points in the second dimension. The grid sizes in these dimensions are represented by $\Delta x = \frac{b_1 - a_1}{n_x}$

and $\Delta y = \frac{b_2 - a_2}{n_y}$. We denote x_i as the i -th grid point in the first dimension and y_j as the j -th grid point in the second dimension. The semi-discrete formulation for a general numerical Hamiltonian is as follows:

$$\begin{cases} \dot{\phi}_{i,j}(t) + \hat{H}(x_i, y_j, t, (D_x^+ \phi)_{i,j}(t), (D_x^- \phi)_{i,j}(t), (D_y^+ \phi)_{i,j}(t), (D_y^- \phi)_{i,j}(t)) \\ \quad = \epsilon(D_{xx}\phi)_{i,j}(t) + \epsilon(D_{yy}\phi)_{i,j}(t), & i = 1, \dots, n_x; j = 1, \dots, n_y, t \in [0, T], \\ \phi_{i,j}(0) = g(x_i, y_j), & i = 1, \dots, n_x; j = 1, \dots, n_y. \end{cases} \quad (14)$$

where D_x^+ , D_x^- , D_y^+ , D_y^- , D_{xx} , and D_{yy} are finite difference operators defined by

$$\begin{aligned} (D_x^+ \phi)_{i,j} &= \frac{\phi_{i+1,j}(t) - \phi_{i,j}(t)}{\Delta x}, & (D_x^- \phi)_{i,j} &= \frac{\phi_{i,j}(t) - \phi_{i-1,j}(t)}{\Delta x}, \\ (D_y^+ \phi)_{i,j} &= \frac{\phi_{i,j+1}(t) - \phi_{i,j}(t)}{\Delta y}, & (D_y^- \phi)_{i,j} &= \frac{\phi_{i,j}(t) - \phi_{i,j-1}(t)}{\Delta y}, \\ (D_{xx}\phi)_{i,j} &= \frac{\phi_{i+1,j}(t) - 2\phi_{i,j}(t) + \phi_{i-1,j}(t)}{\Delta x^2}, & (D_{yy}\phi)_{i,j} &= \frac{\phi_{i,j+1}(t) - 2\phi_{i,j}(t) + \phi_{i,j-1}(t)}{\Delta y^2}. \end{aligned}$$

Analogous to the one-dimensional case discussed in Section 3.1, we require that the numerical Hamiltonian \hat{H} exhibits both consistency, meaning that $\hat{H}(x, y, t, p_1, p_1, p_2, p_2) = H(x, y, t, p_1, p_2)$, and monotonicity, meaning that $\hat{H}(x, y, t, p_1^+, p_1^-, p_2^+, p_2^-)$ is non-increasing with respect to p_1^+ and p_2^+ , and non-decreasing with respect to p_1^- and p_2^- .

This semi-discrete equation can be solved using the following saddle point formulation:

$$\begin{aligned} \min_{\substack{\phi_{i,j}, \forall i,j \\ \phi_{i,j}(0) = g(x_i, y_j)}} \max_{\substack{\rho_{i,j}, v_{i,j}^+, v_{i,j}^-, \\ w_{i,j}^+, w_{i,j}^-; \rho_{i,j} \geq 0}} & \int_0^T \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \rho_{i,j}(t) \left(\dot{\phi}_{i,j}(t) + v_{i,j}^+(t) (D_x^+ \phi)_{i,j}(t) + v_{i,j}^-(t) (D_x^- \phi)_{i,j}(t) \right. \\ & \left. + w_{i,j}^+(t) (D_y^+ \phi)_{i,j}(t) + w_{i,j}^-(t) (D_y^- \phi)_{i,j}(t) - \epsilon(D_{xx}\phi)_{i,j}(t) - \epsilon(D_{yy}\phi)_{i,j}(t) \right. \\ & \left. - \hat{H}^*(x_i, y_j, t, v_{i,j}^+(t), v_{i,j}^-(t), w_{i,j}^+(t), w_{i,j}^-(t)) \right) dt - c \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j}(T), \end{aligned} \quad (15)$$

where $c > 0$ is a hyper-parameter, and $\hat{H}^*(x_i, y_j, t, \cdot, \cdot, \cdot, \cdot)$ is the Fenchel-Legendre transform of $\hat{H}(x_i, y_j, t, \cdot, \cdot, \cdot, \cdot)$. For more properties of this saddle point problem and the corresponding algorithm, see Appendix A.4 and Algorithm 4.

4.2 Fully-discrete formulation

In this section, we adopt implicit time discretization to derive a fully-discrete formulation, enabling the selection of larger time steps to bypass the CFL condition. We represent the number of grid points in the interval $[0, T]$ as n_t , and the spacing between consecutive grid points as $\Delta t = \frac{T}{n_t - 1}$. The fully-discrete HJ PDE with a numerical Hamiltonian \hat{H} is given by

$$\begin{cases} (D_t^- \phi)_{i,j,k} + \hat{H}(x_i, y_j, t_k, (D_x^+ \phi)_{i,j,k}, (D_x^- \phi)_{i,j,k}, (D_y^+ \phi)_{i,j,k}, (D_y^- \phi)_{i,j,k}) \\ \quad = \epsilon(D_{xx}\phi)_{i,j,k} + \epsilon(D_{yy}\phi)_{i,j,k}, & i = 1, \dots, n_x; j = 1, \dots, n_y; k = 2, \dots, n_t, \\ \phi_{i,j,1} = g(x_i, y_j), & i = 1, \dots, n_x; j = 1, \dots, n_y, \end{cases} \quad (16)$$

and the saddle point problem becomes

$$\begin{aligned} \min_{\substack{\phi_{i,j,k}, \forall i,j,k \\ \phi_{i,j,1} = g(x_i, y_j)}} \max_{\substack{\rho_{i,j,k}, v_{i,j,k}^+, v_{i,j,k}^-, \\ w_{i,j,k}^+, w_{i,j,k}^-; \rho_{i,j,k} \geq 0}} & \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t-1} \rho_{i,j,k} \left((D_t^- \phi)_{i,j,k+1} + v_{i,j,k}^+ (D_x^+ \phi)_{i,j,k+1} + v_{i,j,k}^- (D_x^- \phi)_{i,j,k+1} + w_{i,j,k}^+ (D_y^+ \phi)_{i,j,k+1} \right. \\ & \left. + w_{i,j,k}^- (D_y^- \phi)_{i,j,k+1} - \epsilon(D_{xx}\phi)_{i,j,k+1} - \epsilon(D_{yy}\phi)_{i,j,k+1} - \hat{H}^*(x_i, y_j, t_{k+1}, v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-) \right) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j,n_t}. \end{aligned} \quad (17)$$

This saddle point problem is solved using PDHG method. For more details about this problem and the algorithm, we refer readers to Section A.5 and Algorithm 5.

Just like the one-dimensional scenario, the objective function's linearity concerning ϕ or ρ leads to explicit update formulas for ϕ and ρ . On the other hand, updating (v^+, v^-, w^+, w^-) necessitates solving the proximal point of $(v^+, v^-, w^+, w^-) \mapsto \sum_{i,j,k} \hat{H}^*(x_i, y_j, t_{k+1}, v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-)$, which can be conducted in parallel for each point (x_i, y_j, t_{k+1}) . Furthermore, if \hat{H} is separable, i.e., it can be expressed as $\hat{H}_1^1(x_i, y_j, t_{k+1}, v_{i,j,k}^+) + \hat{H}_2^1(x_i, y_j, t_{k+1}, v_{i,j,k}^-) + \hat{H}_1^2(x_i, y_j, t_{k+1}, w_{i,j,k}^+) + \hat{H}_2^2(x_i, y_j, t_{k+1}, w_{i,j,k}^-)$, then updating $(v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-)$ can be further accomplished in parallel.

Analogous to the scenario in the one-dimensional case, if H is convex, separable, and 1-homogeneous with respect to p , there exists a specific formula for jointly updating ρ, v^+, v^-, w^+, w^- . This, in turn, eliminates the need for the inner loop. For more details, refer to Appendix B.4.

5 Numerical results

In this section, we display a range of numerical results that evaluate the performance of our proposed method. We initially use two simple experiments in Sections 5.1 and 5.2 to present error tables, which confirm that our method yields first-order accuracy in computing the viscosity solution for these examples. Subsequently, we utilize more intricate cases (examples 3 and 4 in Sections 5.3 and 5.4) to demonstrate the benefits of using larger time steps. These experiments highlight the ability of our method to handle Hamiltonians that depend on (x, t) and exhibit non-smooth behaviors.

Among these four examples, the second one involves Hamiltonians that are shifted 1-homogeneous with respect to p , i.e., $H(x, t, \cdot) - H(x, t, 0)$ is 1-homogeneous for any (x, t) . For this case, we implement Algorithms 6 and 7 explained in Appendices B.2 and B.4 to eliminate the necessity of the inner loop in the proposed method. On the other hand, in the other three examples, we use Algorithms 3 and 5. In these experiments, for one-dimensional cases, we apply Engquist-Osher scheme [79, 38, 39] and set the numerical Hamiltonian $\hat{H}(x, t, p^+, p^-) = H_-(x, t, p^+) + H_+(x, t, p^-)$, where H_+ is non-decreasing with respect to p^- , H_- is non-increasing with respect to p^+ , and they satisfy $H_+ + H_- = H$. For two-dimensional cases, if the Hamiltonian H can be written as $H(x, t, p_1, p_2) = H_1(x, t, p_1) + H_2(x, t, p_2)$ for some functions H_1 and H_2 , we handle each dimension separately and set the numerical Hamiltonian $\hat{H}(x, t, p_1^+, p_1^-, p_2^+, p_2^-) = H_{1,-}(x, t, p_1^+) + H_{1,+}(x, t, p_1^-) + H_{2,-}(x, t, p_2^+) + H_{2,+}(x, t, p_2^-)$, where $H_{i,+}$ ($i = 1, 2$) is non-decreasing with respect to p_i^- , $H_{i,-}$ ($i = 1, 2$) is non-increasing with respect to p_i^+ , and they satisfy $H_{i,+} + H_{i,-} = H_i$. Note that our approach also works for non-separable Hamiltonians. For these situations, we define the numerical Hamiltonians based on references that will be specified for each case.

5.1 First experiment: quadratic Hamiltonian

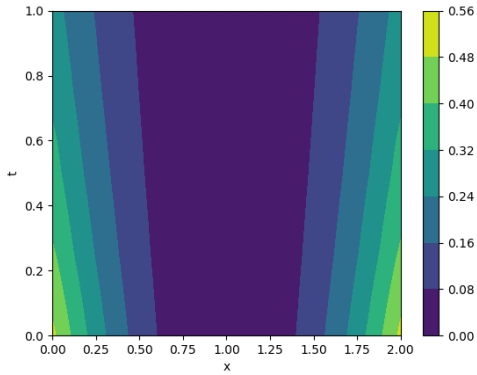
In the first experiment, we solve the following HJ PDE:

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + \frac{1}{2} \|\nabla_x \phi(x, t)\|^2 = 0, & x \in [0, 2]^n, t \in [0, 1], \\ \phi(x, 0) = \frac{1}{2} \|x - 1\|^2, & x \in [0, 2]^n. \end{cases} \quad (18)$$

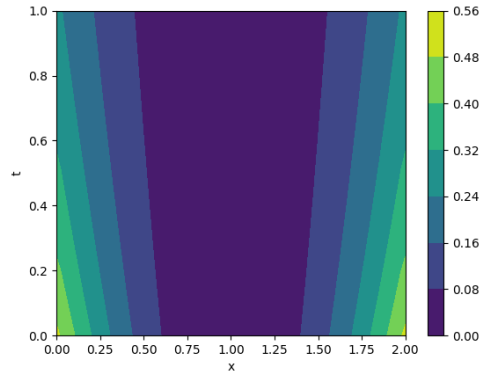
We apply Algorithms 3 and 5 to solve this problem. We use the Engquist-Osher scheme in the saddle point formulation. To be specific, we set the numerical Hamiltonian $\hat{H}(x, t, p^+, p^-) = H_-(p^+) + H_+(p^-)$ for the one-dimensional case and $\hat{H}(x, t, p_1^+, p_1^-, p_2^+, p_2^-) = H_-(p_1^+) + H_+(p_1^-) + H_-(p_2^+) + H_+(p_2^-)$ for the two-dimensional case, where $H_-(p^+) = \frac{1}{2} \min\{p^+, 0\}^2$ and $H_+(p^-) = \frac{1}{2} \max\{p^-, 0\}^2$. We provide the error tables for both one-dimensional and two-dimensional scenarios in Table 1 and Table 2 respectively.

Each table contains two rows of error measurements. In the first row, we showcase the average absolute value of the PDE residuals. This error is calculated using the following formula for one-dimensional cases:

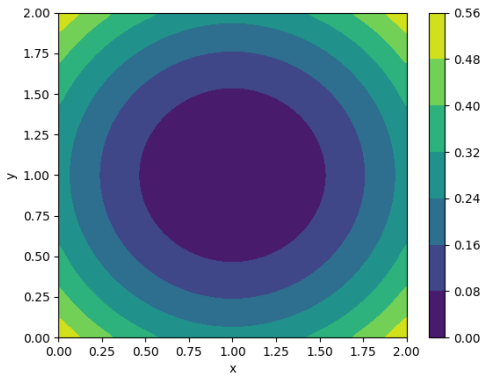
$$\frac{1}{n_x(n_t - 1)} \sum_{i=1}^{n_x} \sum_{k=2}^{n_t} |(D_t^- \phi)_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi)_{i,k}, (D_x^- \phi)_{i,k}) - \epsilon(D_{xx} \phi)_{i,k}|$$



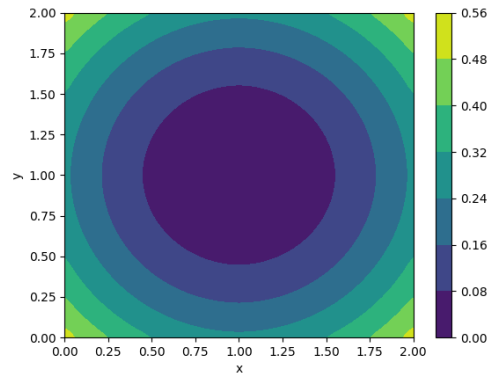
(a) 1D, $n_x = 80$, $n_t = 5$



(b) 1D, $n_x = 80$, $n_t = 41$



(c) 2D, $t = T$, $n_x = n_y = 80$, $n_t = 5$



(d) 2D, $t = T$, $n_x = n_y = 80$, $n_t = 41$

Figure 1: Contours illustrating the solution of the one-dimensional HJ PDE (18) in (a) and (b), along with the level sets for the two-dimensional HJ PDE in (c) and (d). In (a) and (c), a relatively larger time step of $\Delta t = 0.25$ is applied, whereas in (b) and (d), a smaller time step of $\Delta t = 0.025$ is employed.

$n_x \times n_t$	20×11	40×21	80×41	160×81
Averaged absolute residual of HJ PDE	9.99E-07	9.99E-07	9.89E-07	9.82E-07
ℓ^1 relative error	5.81E-02	3.24E-02	1.68E-02	8.27E-03

Table 1: Error table illustrating the performance of our proposed method for solving the one-dimensional HJ PDE (18)

$n_x \times n_y \times n_t$	$20 \times 20 \times 11$	$40 \times 40 \times 21$	$80 \times 80 \times 41$	$160 \times 160 \times 81$
Averaged absolute residual of HJ PDE	1.00E-06	1.00E-06	9.99E-07	1.00E-06
ℓ^1 relative error	5.52E-02	3.00E-02	1.46E-02	6.07E-03

Table 2: Error table illustrating the performance of our proposed method for solving the two-dimensional HJ PDE (18)

and for two-dimensional cases:

$$\frac{1}{n_x n_y (n_t - 1)} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=2}^{n_t} |(D_t^- \phi)_{i,j,k} + \hat{H}(x_i, y_j, t_k, (D_x^+ \phi)_{i,j,k}, (D_x^- \phi)_{i,j,k}, (D_y^+ \phi)_{i,j,k}, (D_y^- \phi)_{i,j,k}) - \epsilon(D_{xx} \phi + D_{yy} \phi)_{i,j,k}|.$$

The residual errors observed across all cases are consistently below 10^{-6} . Remarkably, we employ the residual error as the termination condition for the proposed method, setting the threshold δ to 10^{-6} . These errors confirm the convergence of the our algorithm in terms of reaching an error below the specified threshold for all tested grid sizes.

In the second row, we conduct a comparison between the numerical solution obtained through our proposed method and the reference solution, which we term the ‘‘ground truth’’. This reference solution, denoted as ϕ^{gt} , is generated using either the Lax-Oleinik formula [8, 41, 52] or the explicit Engquist-Osher scheme [79, 38, 39] on a finely discretized grid. Consequently, the ℓ^1 relative errors are presented. The calculation of the error is as follows:

- For one-dimensional cases:

$$\frac{1}{n_x n_t} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t} |\phi_{i,k} - \phi_{i,k}^{gt}|$$

$$\max \left\{ \frac{1}{n_x n_t} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t} |\phi_{i,k}^{gt}|, 1.0 \right\}$$

- For two-dimensional cases:

$$\frac{1}{n_x n_y n_t} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t} |\phi_{i,j,k} - \phi_{i,j,k}^{gt}|$$

$$\max \left\{ \frac{1}{n_x n_y n_t} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t} |\phi_{i,j,k}^{gt}|, 1.0 \right\}$$

This evaluation provides insights into the accuracy of our method when compared to well-established reference solutions. It’s noticeable that the error approximately halves when the grid size is doubled. This behavior demonstrates a first-order error reduction rate corresponding to the increase in grid size.

Furthermore, we present the one-dimensional solution in Fig 1 (a)-(b) and the two-dimensional solution in Fig 1 (c)-(d). In both this example and the subsequent examples, we visualize the one-dimensional solution through the representation of its level sets, where the x -axis corresponds to the spatial domain and the y -axis denotes the time domain. For the two-dimensional solution, we depict the level sets at a specific time instant (in this example, we select $t = 1$), employing the x and y axes to denote the two dimensions of the spatial domain.

In figures (a) and (c), we utilize a larger time step of $\Delta t = \frac{T}{n_t - 1} = 0.25$, while in figures (b) and (d), we opt for a smaller time step of $\Delta t = \frac{T}{n_t - 1} = 0.025$. Throughout all cases, we maintain a spatial discretization count of $n_x = 80$ for the one-dimensional case and $n_x = n_y = 80$ for the two-dimensional scenario. These visual representations reveal a satisfactory performance achieved by employing the larger time step discretization, indicating that the proposed method is not constrained by the CFL condition, thanks to the utilization of implicit time discretization.

5.2 Second experiment: L^1 Hamiltonian

In the second example, we solve the following HJ PDE in one-dimension and two-dimensions

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + \|\nabla_x \phi(x, t)\|_1 = 0, & x \in [0, 2]^n, t \in [0, 1], \\ \phi(x, 0) = \sum_{i=1}^n \sin \pi x_i, & x \in [0, 2]^n. \end{cases} \quad (19)$$

The Hamiltonian in this scenario exhibits non-smooth properties. We solve this problem using Algorithms 6 and 7. Employing the same methodology as illustrated in the previous example, we calculate the error tables. Specifically, the one-dimensional error table is presented in Table 3, while the two-dimensional error table is displayed in Table 4.

$n_x \times n_t$	20×11	40×21	80×41	160×81
Averaged absolute residual of HJ PDE	5.72E-07	7.64E-07	5.96E-07	6.53E-07
ℓ^1 relative error	1.03E-01	5.90E-02	3.20E-02	1.67E-02

Table 3: Error table illustrating the performance of our proposed method for solving the one-dimensional HJ PDE (19)

$n_x \times n_y \times n_t$	$20 \times 20 \times 11$	$40 \times 40 \times 21$	$80 \times 80 \times 41$	$160 \times 160 \times 81$
Averaged absolute residual of HJ PDE	8.62E-07	9.68E-07	9.54E-07	9.82E-07
ℓ^1 relative error	1.03E-01	5.74E-02	2.93E-02	1.36E-02

Table 4: Error table illustrating the performance of our proposed method for solving the two-dimensional HJ PDE (19)

The errors featured in the first rows of both error tables represent the averaged residual errors of the HJ PDE. Notably, all these errors remain below the 10^{-6} threshold. This indicates the convergence of our method in terms of achieving an HJ PDE residual beneath the predetermined threshold of 10^{-6} . Turning to the second rows of the error tables, they depict the ℓ^1 relative errors when compared to the reference solution. We observe that these errors reduce by approximately half as we increase the grid size by a factor of 2. This phenomenon is in alignment with the utilization of a first-order Engquist-Osher scheme.

Furthermore, the level sets of the solution for the one-dimensional case are depicted in Fig 2 (a), while the level sets of the solution at distinct time points ($t = 0.25, 0.5, 0.75$) are presented in Fig 2 (b)-(d). The results of the experiments and the error analysis indicate a higher error magnitude compared to the previous example. Consequently, we adopt a finer grid to enhance the visual representation in the figures.

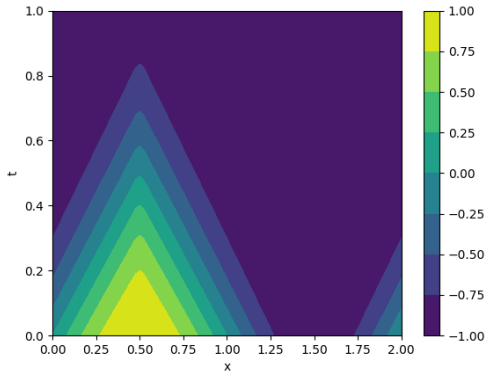
5.3 Third experiment: spatially-dependent Hamiltonian

In this experiment, we solve the following HJ PDE whose Hamiltonian depends on the spatial variable x :

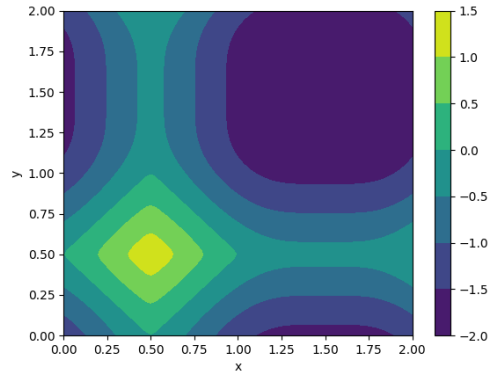
$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + \|\nabla_x \phi(x, t)\| + f(x) = 0, & x \in [0, 2]^n, t \in [0, 1], \\ \phi(x, 0) = \sum_{i=1}^n \sin \pi x_i, & x \in [0, 2]^n, \end{cases} \quad (20)$$

where f is defined by $f(x) = 3 \exp(-4\|x - 1\|^2) + 1$. We utilize Algorithms 3 and 5 to solve this problem. Notably, the Hamiltonian represented in (20) is non-separable, meaning it cannot be expressed as $\sum_{i=1}^n H_i(x, t, p_i)$ for certain functions H_1, \dots, H_n . Under this circumstance, we adopt the numerical Hamiltonian introduced in [79]. For one-dimensional instances, the numerical Hamiltonian is defined as $(\max\{p^-, 0\}^2 + \min\{p^+, 0\}^2)^{1/2} + f(x)$, whereas for two-dimensional scenarios, it is defined as $(\max\{p_1^-, 0\}^2 + \min\{p_1^+, 0\}^2 + \max\{p_2^-, 0\}^2 + \min\{p_2^+, 0\}^2)^{1/2} + f(x)$.

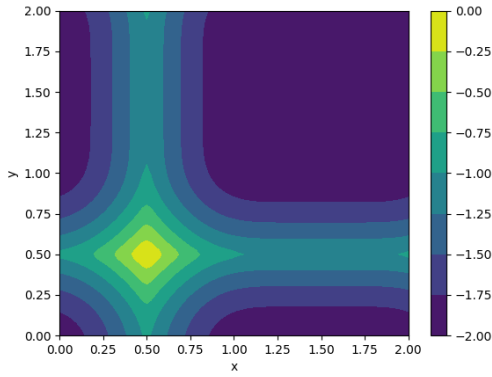
The outcomes derived through our proposed approach are depicted in Figure 3. In Figure 3 (a), we present the level sets of the one-dimensional solution achieved using a relatively larger time step of $\Delta t = 0.25$, while



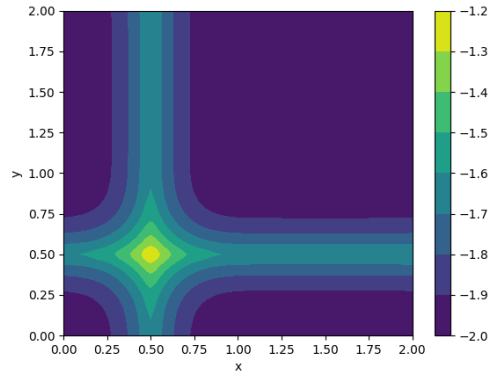
(a) 1D, $n_t = 41$, $n_x = 80$



(b) 2D, $t = 0.25$, $n_t = 41$, $n_x = n_y = 80$



(c) 2D, $t = 0.5$, $n_t = 41$, $n_x = n_y = 80$



(d) 2D, $t = 0.75$, $n_t = 41$, $n_x = n_y = 80$

Figure 2: Visualization of the solution to the one-dimensional HJ PDE (19) in (a), and the evolution of level sets for the two-dimensional HJ PDE solution at various time instances ($t = 0.25$, $t = 0.5$, and $t = 0.75$) in (b)-(d).

in Figure 3 (b), we display the level sets of the one-dimensional solution attained using a finer time step of $\Delta t = 0.025$. The outcomes of the two-dimensional solution, obtained using a time step of $\Delta t = 0.025$, are depicted in Figures 3 (c) and (d). Figure 3 (c) illustrates the solution at $t = 0.25$, while Figure 3 (d) displays the solution at $t = 0.75$. This example demonstrates the capability of our proposed method to effectively manage some complex Hamiltonians with spatial dependency.

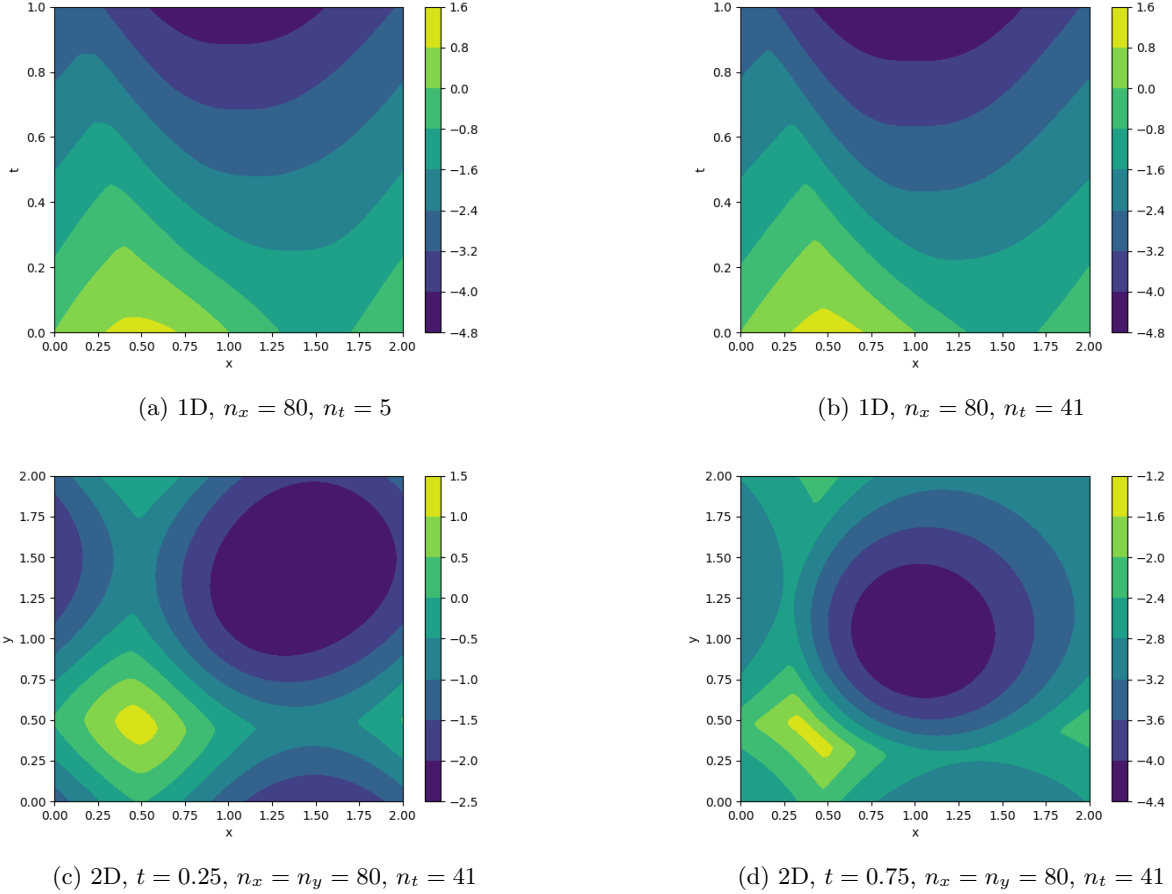


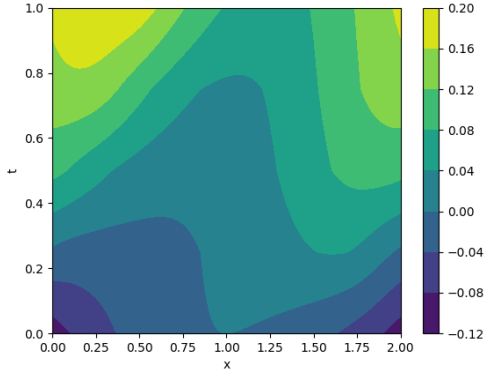
Figure 3: Contours illustrating the solution of the one-dimensional HJ PDE (20) in (a) and (b), along with the level sets for the two-dimensional HJ PDE in (c) and (d). In (a), a relatively larger time step of $\Delta t = 0.25$ is applied, whereas in (b)-(d), a smaller time step of $\Delta t = 0.025$ is employed.

5.4 Fourth experiment: spatiotemporally-dependent Hamiltonian

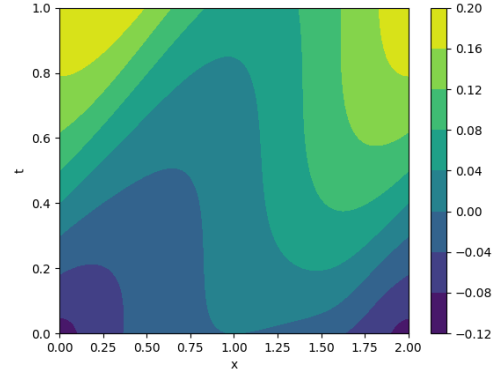
In this experiment, we solve the following viscous HJ PDE whose Hamiltonian depends on both the spatial variable x and the time variable t :

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + \frac{1}{2} \|\nabla_x \phi(x, t)\|^2 + f(x, t) = 0.1 \Delta_x \phi(x, t), & x \in [0, 2]^n, t \in [0, 1], \\ \phi(x, 0) = -\frac{1}{10} \|x - 1\|^2, & x \in [0, 2]^n, \end{cases} \quad (21)$$

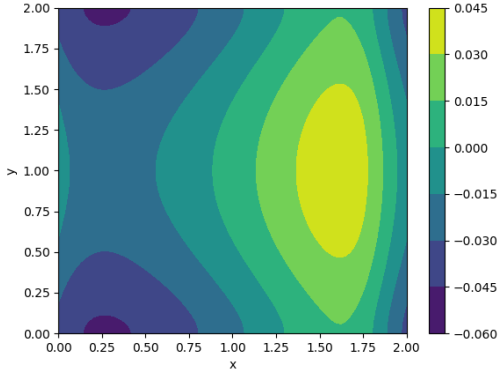
where $f(x, t) = -\frac{1}{2} \min\{(x_1 - t - 0.5)^2, (x_1 - t + 1.5)^2, (x_1 - t - 2.5)^2\} - \frac{1}{4} \sum_{j=2}^n (x_j - 1)^2$. We apply Algorithms 3 and 5 to solve this problem. We use the Engquist-Osher scheme in the saddle point formulation. To be specific, we set the numerical Hamiltonian $\hat{H}(x, t, p^+, p^-) = H_-(p^+) + H_+(p^-) + f(x, t)$ for the one-dimensional case and $\hat{H}(x, t, p_1^+, p_1^-, p_2^+, p_2^-) = H_-(p_1^+) + H_+(p_1^-) + H_-(p_2^+) + H_+(p_2^-) + f(x, t)$ for the two-dimensional case, where $H_-(p^+) = \frac{1}{2} \min\{p^+, 0\}^2$ and $H_+(p^-) = \frac{1}{2} \max\{p^-, 0\}^2$.



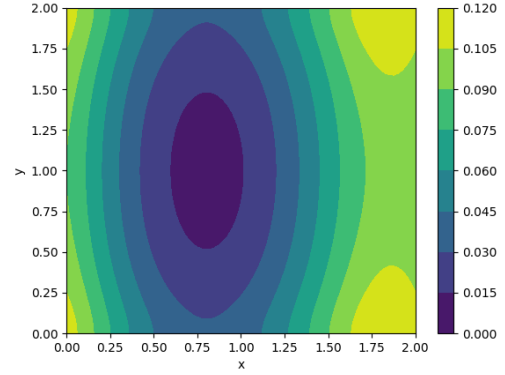
(a) 1D, $n_x = 80$, $n_t = 5$



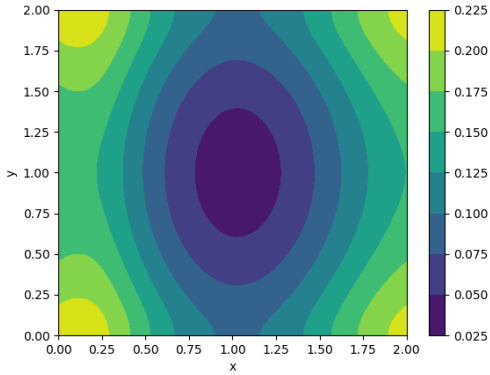
(b) 1D, $n_x = 80$, $n_t = 41$



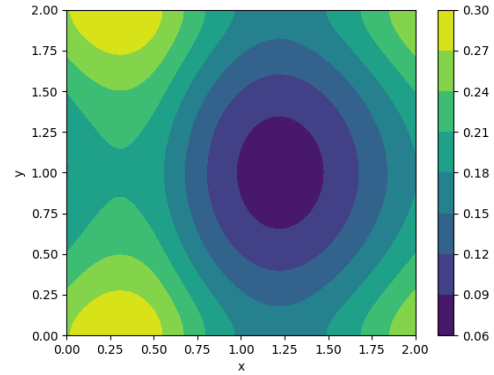
(c) 2D, $t = 0.25$, $n_x = n_y = 80$, $n_t = 5$



(d) 2D, $t = 0.5$, $n_x = n_y = 80$, $n_t = 5$



(e) 2D, $t = 0.75$, $n_x = n_y = 80$, $n_t = 5$



(f) 2D, $t = 1$, $n_x = n_y = 80$, $n_t = 5$

Figure 4: Numeric solutions for the one-dimensional and two-dimensional HJ PDE (21) are depicted. In (a), we display the level sets of the one-dimensional solution obtained using a larger time step of $\Delta t = 0.25$, whereas in (b), we showcase the one-dimensional solution achieved with a smaller time step of $\Delta t = 0.025$. Furthermore, level sets of the two-dimensional solution are presented at distinct time points ($t = 0.25, 0.5, 0.75, 1.0$) in (c)-(f).

The solution is illustrated in Fig 4. In (a)-(b), we present the level sets depicting the solution to the one-dimensional HJ PDE. In (a), a larger time step of $\Delta t = 0.25$ is utilized, while in (b), a smaller time step of $\Delta t = 0.025$ is employed. Additionally, we provide contour plots showcasing the solution to the two-dimensional PDE at distinct time instances ($t = 0.25, 0.5, 0.75, 1.0$) in (c)-(f). Due to computational complexity, we adopt a larger time step of $\Delta t = 0.25$ in the two-dimensional case. This example demonstrates the capability of our proposed method to handle intricate Hamiltonians dependent on (x, t) .

6 Summary

This paper solves HJ PDEs using a saddle point formulation, which is solved using PDHG method. We provide numerical validation that this method can compute the viscosity solutions with errors related to the grid size. We can handle certain Hamiltonians which depend on (x, t) . Moreover, we use implicit time discretization, which circumvents the restrictive CFL time step restriction for explicit methods. In other words, we can choose big time step to speed up the computation. The merit of this approach lies in the simplicity of the saddle point formulation. This simple formulation is achieved by capitalizing on the Fenchel-Legendre transform and the duality inherent in HJ PDEs. This simplicity facilitates updates within our method to have either explicit formulations or be conducive to parallel computation. In a special case where the Hamiltonian $H(x, t, p)$ is separable and 1-homogeneous with respect to p , the saddle point formulation takes the standard form in [20], ensuring the convergence of the proposed algorithms through PDHG theory (see Remark B.1).

Although it is a first-order method, it has the potential to serve as an initialization for more accurate methods, particularly in applications that demand smaller errors. It may be also an interesting future direction to combine this method with high order schemes. Moreover, this method converts an equation to a saddle point problem which can fit pretty well under the framework of machine learning. For solving HJ PDEs or problems related to HJ PDEs, the formulas provided in this paper can provide some ideas on design of loss functions.

Acknowledgement

We would like to express our thanks to Dr. Liu Yang for providing the Jax code of Algorithm 6, which greatly assisted in the numerical implementation of our research.

References

- [1] M. AKIAN, R. BAPAT, AND S. GAUBERT, *Max-plus algebra*, Handbook of linear algebra, 39 (2006).
- [2] M. AKIAN, S. GAUBERT, AND A. LAKHOVA, *The max-plus finite element method for solving deterministic optimal control problems: basic properties and convergence analysis*, SIAM Journal on Control and Optimization, 47 (2008), pp. 817–848.
- [3] A. ALLA, M. FALCONE, AND L. SALUZZI, *An efficient DP algorithm on a tree-structure for finite horizon optimal control problems*, SIAM Journal on Scientific Computing, 41 (2019), pp. A2384–A2406.
- [4] A. ALLA, M. FALCONE, AND S. VOLKWEIN, *Error analysis for POD approximations of infinite horizon problems via the dynamic programming approach*, SIAM Journal on Control and Optimization, 55 (2017), pp. 3091–3115.
- [5] V. I. ARNOL'D, *Mathematical methods of classical mechanics*, vol. 60 of Graduate Texts in Mathematics, Springer-Verlag, New York, 1989. Translated from the 1974 Russian original by K. Vogtmann and A. Weinstein, Corrected reprint of the second (1989) edition.
- [6] A. BACHOUCH, C. HURÉ, N. LANGRENÉ, AND H. PHAM, *Deep neural networks algorithms for stochastic control problems on finite horizon: numerical applications*, arXiv preprint arXiv:1812.05916, (2018).

- [7] M. BARDI AND I. CAPUZZO-DOLCETTA, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, Systems & Control: Foundations & Applications, Birkhäuser Boston, Inc., Boston, MA, 1997. With appendices by Maurizio Falcone and Pierpaolo Soravia.
- [8] M. BARDI AND L. EVANS, *On Hopf's formulas for solutions of Hamilton-Jacobi equations*, Nonlinear Analysis: Theory, Methods & Applications, 8 (1984), pp. 1373 – 1381.
- [9] E. BARRON, L. EVANS, AND R. JENSEN, *Viscosity solutions of Isaacs' equations and differential games with Lipschitz controls*, Journal of Differential Equations, 53 (1984), pp. 213 – 233.
- [10] D. P. BERTSEKAS, *Reinforcement learning and optimal control*, Athena Scientific, Belmont, Massachusetts, (2019).
- [11] O. BOKANOWSKI, J. GARCKE, M. GRIEBEL, AND I. KLONPMAKER, *An adaptive sparse grid semi-Lagrangian scheme for first order Hamilton-Jacobi Bellman equations*, Journal of Scientific Computing, 55 (2013), pp. 575–605.
- [12] L. BRICENO-ARIAS, D. KALISE, Z. KOBEISSI, M. LAURIERE, A. M. GONZÁLEZ, AND F. J. SILVA, *On the implementation of a primal-dual algorithm for second order time-dependent mean field games with local couplings*, ESAIM: Proceedings and Surveys, 65 (2019), pp. 330–348.
- [13] L. M. BRICENO-ARIAS, D. KALISE, AND F. J. SILVA, *Proximal methods for stationary mean field games with local couplings*, SIAM Journal on Control and Optimization, 56 (2018), pp. 801–836.
- [14] R. BUCKDAHN, P. CARDALIAGUET, AND M. QUINCAMPOIX, *Some recent aspects of differential game theory*, Dynamic Games and Applications, 1 (2011), pp. 74–114.
- [15] C. CARATHÉODORY, *Calculus of variations and partial differential equations of the first order. Part I: Partial differential equations of the first order*, Translated by Robert B. Dean and Julius J. Brandstatter, Holden-Day, Inc., San Francisco-London-Amsterdam, 1965.
- [16] ———, *Calculus of variations and partial differential equations of the first order. Part II: Calculus of variations*, Translated from the German by Robert B. Dean, Julius J. Brandstatter, translating editor, Holden-Day, Inc., San Francisco-London-Amsterdam, 1967.
- [17] P. CARDALIAGUET AND C.-A. LEHALLE, *Mean field game of controls and an application to trade crowding*, Mathematics and Financial Economics, 12 (2018), pp. 335–363.
- [18] R. CARMONA, J. FOUQUE, AND L. SUN, *Mean field games and systemic risk*, Communications in Mathematical Sciences, (2015).
- [19] J. A. CARRILLO, L. WANG, AND C. WEI, *Structure preserving primal dual methods for gradient flows with nonlinear mobility transport distances*, arXiv preprint arXiv:2303.16534, (2023).
- [20] A. CHAMBOLLE AND T. POCK, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of mathematical imaging and vision, 40 (2011), pp. 120–145.
- [21] P. CHAUDHARI, A. OBERMAN, S. OSHER, S. SOATTO, AND G. CARLIER, *Deep relaxation: partial differential equations for optimizing deep neural networks*, Research in the Mathematical Sciences, 5 (2018), pp. 1–30.
- [22] P. CHEN, J. DARBON, AND T. MENG, *Hopf-type representation formulas and efficient algorithms for certain high-dimensional optimal control problems*, arXiv preprint arXiv:2110.02541, (2021).
- [23] ———, *Lax-Oleinik-type formulas and efficient algorithms for certain high-dimensional optimal control problems*, arXiv preprint arXiv:2109.14849, (2021).
- [24] Y. T. CHOW, J. DARBON, S. OSHER, AND W. YIN, *Algorithm for overcoming the curse of dimensionality for state-dependent Hamilton-Jacobi equations*, Journal of Computational Physics, 387 (2019), pp. 376–409.

- [25] R. COURANT AND D. HILBERT, *Methods of mathematical physics. Vol. II*, Wiley Classics Library, John Wiley & Sons, Inc., New York, 1989. Partial differential equations, Reprint of the 1962 original, A Wiley-Interscience Publication.
- [26] J. DARBON, *On convex finite-dimensional variational methods in imaging sciences and Hamilton–Jacobi equations*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 2268–2293.
- [27] J. DARBON, P. M. DOWER, AND T. MENG, *Neural network architectures using min-plus algebra for solving certain high-dimensional optimal control problems and Hamilton–Jacobi PDEs*, Mathematics of Control, Signals, and Systems, 35 (2023), pp. 1–44.
- [28] J. DARBON, G. P. LANGLOIS, AND T. MENG, *Overcoming the curse of dimensionality for some Hamilton–Jacobi partial differential equations via neural network architectures*, Res. Math. Sci., 7 (2020), p. 20.
- [29] J. DARBON AND T. MENG, *On decomposition models in imaging sciences and multi-time Hamilton–Jacobi partial differential equations*, SIAM Journal on Imaging Sciences, 13 (2020), pp. 971–1014.
- [30] ———, *On some neural network architectures that can represent viscosity solutions of certain high dimensional hamilton–jacobi partial differential equations*, Journal of Computational Physics, 425 (2021), p. 109907.
- [31] J. DARBON, T. MENG, AND E. RESMERITA, *On Hamilton–Jacobi PDEs and image denoising models with certain nonadditive noise*, Journal of Mathematical Imaging and Vision, 64 (2022), pp. 408–441.
- [32] J. DARBON AND S. OSHER, *Algorithms for overcoming the curse of dimensionality for certain Hamilton–Jacobi equations arising in control theory and elsewhere*, Research in the Mathematical Sciences, 3 (2016), p. 19.
- [33] B. DJERIDANE AND J. LYGEROS, *Neural approximation of PDE solutions: An application to reachability computations*, in Proceedings of the 45th IEEE Conference on Decision and Control, Dec 2006, pp. 3034–3039.
- [34] S. DOLGOV, D. KALISE, AND K. KUNISCH, *A tensor decomposition approach for high-dimensional Hamilton–Jacobi–Bellman equations*, arXiv preprint arXiv:1908.01533, (2019).
- [35] P. M. DOWER, W. M. MCENEANEY, AND H. ZHANG, *Max-plus fundamental solution semigroups for optimal control problems*, in 2015 Proceedings of the Conference on Control and its Applications, SIAM, 2015, pp. 368–375.
- [36] W. E, J. HAN, Q. LI, ET AL., *A mean-field optimal control formulation of deep learning*, Research in the Mathematical Sciences, 6 (2019), pp. 1–41.
- [37] R. J. ELLIOTT, *Viscosity solutions and optimal control*, vol. 165 of Pitman Research Notes in Mathematics Series, Longman Scientific & Technical, Harlow; John Wiley & Sons, Inc., New York, 1987.
- [38] B. ENGQUIST AND S. OSHER, *Stable and entropy satisfying approximations for transonic flow calculations*, Mathematics of Computation, 34 (1980), pp. 45–75.
- [39] ———, *One-sided difference approximations for nonlinear conservation laws*, Mathematics of Computation, 36 (1981), pp. 321–351.
- [40] L. C. EVANS, *Adjoint and compensated compactness methods for Hamilton–Jacobi PDE*, Archive for rational mechanics and analysis, 197 (2010), pp. 1053–1088.
- [41] L. C. EVANS, *Partial differential equations*, vol. 19 of Graduate Studies in Mathematics, American Mathematical Society, Providence, RI, second ed., 2010.
- [42] L. C. EVANS AND P. E. SOUGANIDIS, *Differential games and representation formulas for solutions of Hamilton–Jacobi–Isaacs equations*, Indiana University Mathematics Journal, 33 (1984), pp. 773–797.

- [43] W. FLEMING AND W. MCENEANEY, *A max-plus-based algorithm for a Hamilton–Jacobi–Bellman equation of nonlinear filtering*, SIAM Journal on Control and Optimization, 38 (2000), pp. 683–710.
- [44] W. H. FLEMING AND R. W. RISHEL, *Deterministic and stochastic optimal control*, Bulletin of the American Mathematical Society, 82 (1976), pp. 869–870.
- [45] W. H. FLEMING AND H. M. SONER, *Controlled Markov processes and viscosity solutions*, vol. 25, Springer Science & Business Media, 2006.
- [46] G. FU, S. OSHER, AND W. LI, *High order spatial discretization for variational time implicit schemes: Wasserstein gradient flows and reaction-diffusion systems*, arXiv preprint arXiv:2303.08950, (2023).
- [47] J. GARCKE AND A. KRÖNER, *Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids*, Journal of Scientific Computing, 70 (2017), pp. 1–28.
- [48] S. GAUBERT, W. MCENEANEY, AND Z. QU, *Curse of dimensionality reduction in max-plus based approximation methods: Theoretical estimates and improved pruning algorithms*, in 2011 50th IEEE Conference on Decision and Control and European Control Conference, IEEE, 2011, pp. 1054–1061.
- [49] D. GOMES, L. NURBEKYAN, AND E. PIMENTEL, *Economic models and mean-field games theory*, 07 2015.
- [50] D. A. GOMES AND J. SAÚDE, *Mean field games models—a brief survey*, Dynamic Games and Applications, 4 (2014), pp. 110–154.
- [51] J. HAN, A. JENTZEN, AND W. E, *Solving high-dimensional partial differential equations using deep learning*, Proceedings of the National Academy of Sciences, 115 (2018), pp. 8505–8510.
- [52] E. HOPF, *Generalized solutions of non-linear equations of first order*, J. Math. Mech., 14 (1965), pp. 951–973.
- [53] M. B. HOROWITZ, A. DAMLE, AND J. W. BURDICK, *Linear Hamilton Jacobi Bellman equations in high dimensions*, in 53rd IEEE Conference on Decision and Control, IEEE, 2014, pp. 5880–5887.
- [54] C. HU AND C. SHU, *A discontinuous Galerkin finite element method for Hamilton–Jacobi equations*, SIAM Journal on Scientific Computing, 21 (1999), pp. 666–690.
- [55] M. HUANG, R. P. MALHAMÉ, AND P. E. CAINES, *Large population stochastic dynamic games: closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle*, (2006).
- [56] C. HURÉ, H. PHAM, A. BACHOUCH, AND N. LANGRENÉ, *Deep neural networks algorithms for stochastic control problems on finite horizon, part I: convergence analysis*, arXiv preprint arXiv:1812.04300, (2018).
- [57] C. HURÉ, H. PHAM, AND X. WARIN, *Some machine learning schemes for high-dimensional nonlinear PDEs*, arXiv preprint arXiv:1902.01599, (2019).
- [58] H. ISHII, *Representation of solutions of Hamilton-Jacobi equations*, Nonlinear Analysis: Theory, Methods & Applications, 12 (1988), pp. 121 – 146.
- [59] M. JACOBS, F. LÉGER, W. LI, AND S. OSHER, *Solving large-scale optimization problems with a convergence rate independent of grid size*, SIAM Journal on Numerical Analysis, 57 (2019), pp. 1100–1123.
- [60] F. JIANG, G. CHOU, M. CHEN, AND C. J. TOMLIN, *Using neural networks to compute approximate and guaranteed feasible Hamilton-Jacobi-Bellman PDE solutions*, arXiv preprint arXiv:1611.03158, (2016).
- [61] G. JIANG AND D. PENG, *Weighted ENO schemes for Hamilton–Jacobi equations*, SIAM Journal on Scientific Computing, 21 (2000), pp. 2126–2143.

- [62] D. KALISE, S. KUNDU, AND K. KUNISCH, *Robust feedback control of nonlinear PDEs by numerical approximation of high-dimensional Hamilton-Jacobi-Isaacs equations*, arXiv preprint arXiv:1905.06276, (2019).
- [63] D. KALISE AND K. KUNISCH, *Polynomial approximation of high-dimensional Hamilton-Jacobi-Bellman equations and applications to feedback control of semilinear parabolic PDEs*, SIAM Journal on Scientific Computing, 40 (2018), pp. A629–A652.
- [64] W. KANG AND L. C. WILCOX, *Mitigating the curse of dimensionality: sparse grid characteristics method for optimal feedback control and HJB equations*, Computational Optimization and Applications, 68 (2017), pp. 289–315.
- [65] A. C. KIZILKALE, R. SALHAB, AND R. P. MALHAMÉ, *An integral control formulation of mean field game based large scale coordination of loads in smart grids*, Automatica, 100 (2019), pp. 312–322.
- [66] K. KUNISCH, S. VOLKWEIN, AND L. XIE, *HJB-POD-based feedback design for the optimal control of evolution problems*, SIAM Journal on Applied Dynamical Systems, 3 (2004), pp. 701–722.
- [67] A. LACHAPELLE AND M.-T. WOLFRAM, *On a mean field game approach modeling congestion and aversion in pedestrian crowds*, Transportation research part B: methodological, 45 (2011), pp. 1572–1589.
- [68] P. LAMBRIANIDES, Q. GONG, AND D. VENTURI, *A new scalable algorithm for computational optimal control under uncertainty*, arXiv preprint arXiv:1909.07960, (2019).
- [69] L. LANDAU AND E. LIFSCHIC, *Course of theoretical physics. vol. 1: Mechanics*, Oxford, 1978.
- [70] J.-M. LASRY AND P.-L. LIONS, *Mean field games*, Japanese journal of mathematics, 2 (2007), pp. 229–260.
- [71] W. LEE, S. LIU, H. TEMBINE, W. LI, AND S. OSHER, *Controlling propagation of epidemics via mean-field control*, SIAM Journal on Applied Mathematics, 81 (2021), pp. 190–207.
- [72] S. LIU, S. LIU, S. OSHER, AND W. LI, *A first-order computational algorithm for reaction-diffusion type equations via primal-dual hybrid gradient method*, 2023.
- [73] S. LIU, S. OSHER, W. LI, AND C.-W. SHU, *A primal-dual approach for solving conservation laws with implicit in time approximations*, Journal of Computational Physics, 472 (2023), p. 111654.
- [74] W. MCENEANEY, *Max-plus methods for nonlinear control and estimation*, Springer Science & Business Media, 2006.
- [75] ———, *A curse-of-dimensionality-free numerical method for solution of certain HJB PDEs*, SIAM Journal on Control and Optimization, 46 (2007), pp. 1239–1276.
- [76] W. M. MCENEANEY, A. DESHPANDE, AND S. GAUBERT, *Curse-of-complexity attenuation in the curse-of-dimensionality-free method for HJB PDEs*, in 2008 American Control Conference, IEEE, 2008, pp. 4684–4690.
- [77] W. M. MCENEANEY AND L. J. KLUBERG, *Convergence rate for a curse-of-dimensionality-free method for a class of HJB PDEs*, SIAM Journal on Control and Optimization, 48 (2009), pp. 3052–3079.
- [78] K. N. NIARCHOS AND J. LYGEROS, *A neural approximation to continuous time reachability computations*, in Proceedings of the 45th IEEE Conference on Decision and Control, Dec 2006, pp. 6313–6318.
- [79] S. OSHER AND J. A. SETHIAN, *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations*, Journal of computational physics, 79 (1988), pp. 12–49.
- [80] S. OSHER AND C. SHU, *High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations*, SIAM Journal on Numerical Analysis, 28 (1991), pp. 907–922.

- [81] N. PAPADAKIS, G. PEYRÉ, AND E. OUDET, *Optimal transport with proximal splitting*, SIAM Journal on Imaging Sciences, 7 (2014), pp. 212–238.
- [82] C. REISINGER AND Y. ZHANG, *Rectified deep neural networks overcome the curse of dimensionality for nonsmooth value functions in zero-sum games of nonlinear stiff systems*, arXiv preprint arXiv:1903.06652, (2019).
- [83] V. R. ROYO AND C. TOMLIN, *Recursive regression with neural networks: Approximating the HJI PDE solution*, arXiv preprint arXiv:1611.02739, (2016).
- [84] C.-W. SHU, *High order numerical methods for time dependent Hamilton-Jacobi equations*, in Mathematics and computation in imaging science and information processing, World Scientific, 2007, pp. 47–91.
- [85] J. SIRIGNANO AND K. SPILIOPOULOS, *DGM: A deep learning algorithm for solving partial differential equations*, Journal of Computational Physics, 375 (2018), pp. 1339 – 1364.
- [86] E. TODOROV, *Efficient computation of optimal actions*, Proceedings of the national academy of sciences, 106 (2009), pp. 11478–11483.
- [87] T. VALKONEN, *A primal–dual hybrid gradient method for nonlinear operators with applications to MRI*, Inverse Problems, 30 (2014), p. 055012.
- [88] I. YEGOROV AND P. M. DOWER, *Perspectives on characteristics based curse-of-dimensionality-free numerical approaches for solving Hamilton–Jacobi equations*, Applied Mathematics & Optimization, (2017), pp. 1–49.

A More details about the algorithms

A.1 More discussion about the continuous saddle point formula

The first order optimization condition for the saddle point problem in (3) is

$$\begin{cases} \frac{\partial \phi(x, t)}{\partial t} + H(x, t, \nabla_x \phi(x, t)) = \epsilon \Delta_x \phi(x, t), & x \in \Omega, t \in [0, T], \\ \frac{\partial \rho(x, t)}{\partial t} + \nabla_x \cdot (\rho(x, t) \nabla_p H(x, t, \nabla_x \phi(x, t))) + \epsilon \Delta_x \rho(x, t) = 0, & x \in \Omega, t \in [0, T], \\ \phi(x, 0) = g(x), \quad \rho(x, T) = c, & x \in \Omega, \end{cases} \quad (22)$$

which is similar to the mean-field control problem, except that we do not restrict ρ to be non-negative. There is a gap between (3) and (4), namely, whether we require ρ to be non-negative or not. Although the theoretical understanding of when this condition holds is lacking, numerically we have found that our algorithm works and computes the viscosity solution.

Note that in the last line of (3), v is a vector, while v takes the form of a function in (4). We abuse the notation v here. In (3), there are infinitely many finite-dimensional optimization problems, which depends on (x, t) . Upon their consolidation into a single optimization problem, the variable transforms into a function that depends on (x, t) , denoted as $v(x, t)$ in (4), whose value corresponds to the original variable v in (3).

A.2 One-dimensional semi-discrete method

To solve the semi-discrete equation (10), we propose to solve the saddle point problem (11), whose first order optimality condition is

$$\begin{cases} \dot{\phi}_i(t) + \hat{H}(x_i, t, (D_x^+ \phi)_i(t), (D_x^- \phi)_i(t)) \leq \epsilon (D_{xx} \phi)_i(t), & i = 1, \dots, n_x; t \in [0, T], \\ \dot{\rho}_i(t) + (D_x^-(\rho v^+))_i(t) + (D_x^+(\rho v^-))_i(t) + \epsilon (D_{xx} \rho)_i(t) = 0, & i = 1, \dots, n_x; t \in [0, T], \\ (v_i^+(t), v_i^-(t)) = \nabla_{(p^+, p^-)} \hat{H}(x_i, t, (D_x^+ \phi)_i(t), (D_x^- \phi)_i(t)), & i = 1, \dots, n_x; t \in [0, T], \\ \phi_i(0) = g(x_i), \quad \rho_i(T) = c, & i = 1, \dots, n_x, \\ \rho_i(t) \left(\dot{\phi}_i(t) + \hat{H}(x_i, t, (D_x^+ \phi)_i(t), (D_x^- \phi)_i(t)) - \epsilon (D_{xx} \phi)_i(t) \right) = 0, & i = 1, \dots, n_x; t \in [0, T]. \end{cases} \quad (23)$$

Similar to the continuous setting, if the corresponding mean-field control problem possesses a solution with a positive ρ , the inequality in the first row becomes an equality. This implies that the proposed saddle point problem addresses the semi-discrete HJ PDE (10). This inclusively covers situations where a positive diffusion coefficient ϵ is present. It's important to note that by ensuring an appropriate discretization for the two PDEs in (6), both the discretize-then-optimize and optimize-then-discretize approaches yield the same method. These comments hold true for the following sections and we will not repeat them.

The algorithm for the proposed one-dimensional semi-discrete method is outlined in Algorithm 2, where we denote the objective function in (11) as \mathcal{L}_{semi} .

A.3 One-dimensional fully-discrete method

To solve the fully-discrete equation (12), we propose solving the saddle point problem (13), the first-order optimality condition of which is given by:

$$\left\{ \begin{array}{ll} (D_t^- \phi)_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi)_{i,k}, (D_x^- \phi)_{i,k}) \leq \epsilon(D_{xx}\phi)_{i,k}, & i = 1, \dots, n_x; k = 2, \dots, n_t, \\ (D_t^+ \rho)_{i,k} + (D_x^-(\rho v^+))_{i,k} + (D_x^+(\rho v^-))_{i,k} + \epsilon(D_{xx}\rho)_{i,k} = 0, & i = 1, \dots, n_x; k = 1, \dots, n_t - 1, \\ (v_{i,k}^+, v_{i,k}^-) = \nabla_{(p^+, p^-)} \hat{H}(x_i, t_{k+1}, (D_x^+ \phi)_{i,k+1}, (D_x^- \phi)_{i,k+1}), & i = 1, \dots, n_x; k = 1, \dots, n_t - 1, \\ \phi_{i,1} = g(x_i), \quad \rho_{i,n_t} = c, & i = 1, \dots, n_x, \\ \rho_{i,k-1} \left((D_t^- \phi)_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi)_{i,k}, (D_x^- \phi)_{i,k}) - \epsilon(D_{xx}\phi)_{i,k} \right) = 0, & i = 1, \dots, n_x; k = 2, \dots, n_t. \end{array} \right. \quad (27)$$

The proposed algorithm for solving (13) is presented in Algorithm 3. Within this algorithm, we employ \mathcal{L}_{full} to represent the objective function in (13). During the updating of ϕ , the term $(D_t^+ \rho)_{i,n_t-1}$ represents $\frac{c - \rho_{i,n_t-1}}{\Delta t}$, aligning with the terminal condition for ρ in (27).

A.4 Two-dimensional semi-discrete method

To solve the semi-discrete equation (14), we propose solving the saddle point problem (15), whose first order optimality condition is

$$\left\{ \begin{array}{ll} \dot{\phi}_{i,j}(t) + \hat{H}(x_i, y_j, t, (D_x^+ \phi)_{i,j}(t), (D_x^- \phi)_{i,j}(t), (D_y^+ \phi)_{i,j}(t), (D_y^- \phi)_{i,j}(t)) \\ \leq \epsilon(D_{xx}\phi)_{i,j}(t) + \epsilon(D_{yy}\phi)_{i,j}(t), & i = 1, \dots, n_x; j = 1, \dots, n_y; t \in [0, T], \\ \dot{\rho}_{i,j}(t) + (D_x^-(\rho v^+))_{i,j}(t) + (D_x^+(\rho v^-))_{i,j}(t) + (D_y^-(\rho w^+))_{i,j}(t) \\ + (D_y^+(\rho w^-))_{i,j}(t) + \epsilon(D_{xx}\rho)_{i,j}(t) + \epsilon(D_{yy}\rho)_{i,j}(t) = 0, & i = 1, \dots, n_x; j = 1, \dots, n_y; t \in [0, T], \\ (v_{i,j}^+(t), v_{i,j}^-(t), w_{i,j}^+(t), w_{i,j}^-(t)) = \nabla_{(p_1^+, p_1^-, p_2^+, p_2^-)} \hat{H}(x_i, y_j, t, \\ (D_x^+ \phi)_{i,j}(t), (D_x^- \phi)_{i,j}(t), (D_y^+ \phi)_{i,j}(t), (D_y^- \phi)_{i,j}(t)), & i = 1, \dots, n_x; j = 1, \dots, n_y; t \in [0, T], \\ \phi_{i,j}(0) = g(x_i, y_j), \quad \rho_{i,j}(T) = c, & i = 1, \dots, n_x; j = 1, \dots, n_y, \\ \rho_{i,j}(t) \left(\dot{\phi}_{i,j}(t) + \hat{H}(x_i, y_j, t, (D_x^+ \phi)_{i,j}(t), (D_x^- \phi)_{i,j}(t), (D_y^+ \phi)_{i,j}(t), \\ (D_y^- \phi)_{i,j}(t)) - \epsilon(D_{xx}\phi)_{i,j}(t) - \epsilon(D_{yy}\phi)_{i,j}(t) \right) = 0, & i = 1, \dots, n_x; j = 1, \dots, n_y; t \in [0, T]. \end{array} \right. \quad (31)$$

The algorithm for the proposed two-dimensional semi-discrete method is outlined in Algorithm 4. For the sake of simplicity, we use α to represent $(\alpha_{i,j})_{i,j}$ for any given quantity α . This algorithm closely resembles the one-dimensional case, and thus certain details have been omitted. We adopt the quadratic function as the penalty term for all functions except ϕ . For ϕ , our choice of penalty term for preconditioning is $\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(\|\dot{\phi}_{i,j} - \dot{\phi}_{i,j}^\ell\|^2 + \|(D_x^+ \phi)_{i,j} - (D_x^+ \phi)_{i,j}^\ell\|^2 + \|(D_y^+ \phi)_{i,j} - (D_y^+ \phi)_{i,j}^\ell\|^2 \right)$.

Algorithm 2: The proposed algorithm for solving (11)

Inputs : Stepsize $\tau, \sigma > 0$, error tolerance $\delta > 0$, inner maximal iteration number N_{inner} and outer maximal iteration number N_{outer} .

Outputs: Solution to the saddle point problem (11).

1 For all $i = 1, \dots, n_x$, initialize the functions by $\phi_i^0(t) = g(x_i), \forall t \in [0, T], \rho_i^0 \equiv c, v_i^{0,+} = v_i^{0,-} \equiv 0$.

2 **for** $\ell = 0, 1, \dots, N_{outer} - 1$ **do**

3 Update the functions $\phi_i: [0, T] \rightarrow \mathbb{R}$ for all $i = 1, \dots, n_x$ by

$$\begin{aligned} (\phi_i^{\ell+1})_i &= \arg \min_{\phi_i: \forall i: \phi_i(0)=g(x_i)} \mathcal{L}_{semi}((\phi_i)_i, (\rho_i^\ell)_i, (v_i^{\ell,+})_i, (v_i^{\ell,-})_i) + \frac{1}{2\tau} \sum_{i=1}^{n_x} \left(\|\dot{\phi}_i - \dot{\phi}_i^\ell\|^2 + \|(D_x^+ \phi)_i - (D_x^+ \phi^\ell)_i\|^2 \right) \\ &= (\phi_i^\ell)_i + \tau(-\partial_t^2 - D_{xx})^{-1} \left(\dot{\rho}^\ell + D_x^-(\rho^\ell v^{\ell,+}) + D_x^+(\rho^\ell v^{\ell,-}) + \epsilon D_{xx} \rho^\ell \right), \end{aligned} \quad (24)$$

where $(-\partial_t^2 - D_{xx})^{-1}(f_i)_i$ (here each f_i is a function of t) denotes the solution $(u_i)_i$ to the ODE system $-\ddot{u}_i - (D_{xx}u)_i = f_i$ in $[0, T]$ for all $i = 1, \dots, n_x$ with periodic spatial condition, Dirichlet initial condition $u_i(0) = 0$, and Neumann terminal condition $\dot{u}_i(T) = 0$.

4 **if** $\sum_{i=1}^{n_x} \|\dot{\phi}_i^{\ell+1}(t) + \hat{H}(x_i, t, (D_x^+ \phi^{\ell+1})_i(t), (D_x^- \phi^{\ell+1})_i(t)) - \epsilon(D_{xx} \phi^{\ell+1})_i(t)\|_1 \leq \delta$ **then**

5 | Return $(\phi_i^{\ell+1})_i$.

6 **end**

7 Set $\bar{\phi}_i^{\ell+1} = 2\phi_i^{\ell+1} - \phi_i^\ell$ for all $i = 1, \dots, n_x$.

8 Set $v_i^{\ell+1,0,+} = v_i^{\ell,+}, v_i^{\ell+1,0,-} = v_i^{\ell,-}, \rho_i^{\ell+1,0} = \rho_i^\ell$ for all $i = 1, \dots, n_x$.

9 **for** $m = 0, 1, \dots, N_{inner} - 1$ **do**

10 | Update v_i^+ and v_i^- for all $i = 1, \dots, n_x$ by

$$\begin{aligned} (v_i^{\ell+1,m+1,+}, v_i^{\ell+1,m+1,-})_i &= \arg \max_{v_i^+, v_i^- \forall i} \left\{ \mathcal{L}_{semi}((\bar{\phi}_i^{\ell+1})_i, (\rho_i^{\ell+1,m})_i, (v_i^+)_i, (v_i^-)_i) \right. \\ &\quad \left. - \frac{1}{2\sigma} \sum_{i=1}^{n_x} \left(\|v_i^{\ell+1,m}(v_i^+ - v_i^{\ell+1,m,+})\|^2 + \|v_i^{\ell+1,m}(v_i^- - v_i^{\ell+1,m,-})\|^2 \right) \right\} \\ &= \arg \min_{v_i^+, v_i^- \forall i} \left\{ \sum_{i=1}^{n_x} \int_0^T \hat{H}^*(x_i, t, v_i^+(t), v_i^-(t)) + \frac{\rho_i^{\ell+1,m}(t)}{2\sigma} \left(v_i^+(t) - v_i^{\ell+1,m,+}(t) - \sigma \frac{(D_x^+ \bar{\phi}^{\ell+1})_i(t)}{\rho_i^{\ell+1,m}(t)} \right)^2 \right. \\ &\quad \left. + \frac{\rho_i^{\ell+1,m}(t)}{2\sigma} \left(v_i^-(t) - v_i^{\ell+1,m,-}(t) - \sigma \frac{(D_x^- \bar{\phi}^{\ell+1})_i(t)}{\rho_i^{\ell+1,m}(t)} \right)^2 dt \right\}. \end{aligned} \quad (25)$$

Update ρ_i for all $i = 1, \dots, n_x$ by

$$\begin{aligned} (\rho_i^{\ell+1,m+1})_i &= \arg \max_{\rho_i \forall i: \rho_i \geq 0} \mathcal{L}_{semi}((\bar{\phi}_i^{\ell+1})_i, (\rho_i)_i, (v_i^{\ell+1,m+1,+})_i, (v_i^{\ell+1,m+1,-})_i) - \frac{1}{2\sigma} \sum_{i=1}^{n_x} \|\rho_i - \rho_i^{\ell+1,m}\|^2 \\ &= (\max\{\mu_i^{\ell+1,m+1}, 0\})_i, \end{aligned} \quad (26)$$

where $\mu_i^{\ell+1,m+1}$ is a function defined by

$$\begin{aligned} \mu_i^{\ell+1,m+1} &= \rho_i^{\ell+1,m} + \sigma(\dot{\bar{\phi}}_i^{\ell+1}(t) + v_i^{\ell+1,m+1,+}(t)(D_x^+ \bar{\phi}^{\ell+1})_i(t) + v_i^{\ell+1,m+1,-}(t)(D_x^- \bar{\phi}^{\ell+1})_i(t) - \\ &\quad \hat{H}^*(x_i, t, v_i^{\ell+1,m+1,+}(t), v_i^{\ell+1,m+1,-}(t)) - \epsilon(D_{xx} \bar{\phi}^{\ell+1})_i(t)). \end{aligned}$$

11 **end**

12 Set $v_i^{\ell+1,+} = v_i^{\ell+1,N_{inner},+}, v_i^{\ell+1,-} = v_i^{\ell+1,N_{inner},-}, \rho_i^{\ell+1} = \rho_i^{\ell+1,N_{inner}}$ for all $i = 1, \dots, n_x$.

13 **end**

14 Return $(\phi_i^{N_{outer}})_i$.

Algorithm 3: The proposed algorithm for solving (13)

Inputs : Step size $\tau, \sigma > 0$, error tolerance $\delta > 0$, inner maximal iteration number N_{inner} and outer maximal iteration number N_{outer} .

Outputs: Solution to the saddle point problem (13).

1 For each $i = 1, \dots, n_x$, initialize the matrices by $\phi_{i,k}^0 = g(x_i)$ for $k = 1, \dots, n_t$, $\rho_{i,k}^0 = c$,
 $v_{i,k}^{0,+} = v_{i,k}^{0,-} = 0$ for $k = 1, \dots, n_t - 1$.

2 **for** $\ell = 0, 1, \dots, N_{outer} - 1$ **do**

3 Update the matrix $\phi_{i,k}$ for $i = 1, \dots, n_x$; $k = 1, \dots, n_t$ by

$$\begin{aligned} (\phi_{i,k}^{\ell+1})_{i,k} &= \arg \min_{\substack{\phi_{i,k} \forall i,k \\ \phi_{i,1} = g(x_i)}} \mathcal{L}_{full}((\phi_{i,k})_{i,k}, (\rho_{i,k}^{\ell})_{i,k}, (v_{i,k}^{\ell,+})_{i,k}, (v_{i,k}^{\ell,-})_{i,k}) \\ &\quad + \frac{1}{2\tau} \sum_{i=1}^{n_x} \sum_{k=2}^{n_t} \left(((D_t^- \phi)_{i,k} - (D_t^- \phi^{\ell})_{i,k})^2 + ((D_x^+ \phi)_{i,k} - (D_x^+ \phi^{\ell})_{i,k})^2 \right) \\ &= (\phi_{i,k}^{\ell})_{i,k} + \tau (-D_{tt} - D_{xx})^{-1} \left(D_t^+ \rho^{\ell} + D_x^- (\rho^{\ell} v^{\ell,+}) + D_x^+ (\rho^{\ell} v^{\ell,-}) + \epsilon D_{xx} \rho^{\ell} \right), \end{aligned} \quad (28)$$

where $(-D_{tt} - D_{xx})^{-1}(f_{i,k})_{i,k}$ (for a matrix f with elements $f_{i,k}$ and the linear operator $D_{tt}f = (\frac{f_{i,k-1} - 2f_{i,k} + f_{i,k+1}}{\Delta t^2})_{i,k}$) denotes the solution u to the linear system $-(D_{tt}u)_{i,k+1} - (D_{xx}u)_{i,k+1} = f_{i,k}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ with periodic spatial condition, Dirichlet initial condition $u_{i,1} = 0$, and Neumann terminal condition $u_{i,n_t+1} = u_{i,n_t}$.

4 **if** $\sum_{i=1}^{n_x} \sum_{k=2}^{n_t} |(D_t^- \phi^{\ell+1})_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi^{\ell+1})_{i,k}, (D_x^- \phi^{\ell+1})_{i,k}) - \epsilon (D_{xx} \phi^{\ell+1})_{i,k}| \leq \delta$ **then**
5 | Return $(\phi_{i,k}^{\ell+1})_{i,k}$.

6 **end**

7 Set $\bar{\phi}_{i,k}^{\ell+1} = 2\phi_{i,k}^{\ell+1} - \phi_{i,k}^{\ell}$ for all $i = 1, \dots, n_x$; $k = 2, \dots, n_t$.

8 Set $v_{i,k}^{\ell+1,0,+} = v_{i,k}^{\ell,+}$, $v_{i,k}^{\ell+1,0,-} = v_{i,k}^{\ell,-}$, $\rho_{i,k}^{\ell+1,0} = \rho_{i,k}^{\ell}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$.

9 **for** $m = 0, 1, \dots, N_{inner} - 1$ **do**

10 Update $v_{i,k}^+$ and $v_{i,k}^-$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ by

$$\begin{aligned} ((v_{i,k}^{\ell+1,m+1,+}, v_{i,k}^{\ell+1,m+1,-}))_{i,k} &= \arg \max_{v_{i,k}^+, v_{i,k}^- \forall i,k} \left\{ \mathcal{L}_{full}((\bar{\phi}_{i,k}^{\ell+1})_{i,k}, (\rho_{i,k}^{\ell+1,m})_{i,k}, (v_{i,k}^+)_{i,k}, (v_{i,k}^-)_{i,k}) \right. \\ &\quad \left. - \frac{1}{2\sigma} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \left((\rho_{i,k}^{\ell+1,m} (v_{i,k}^+ - v_{i,k}^{\ell+1,m,+}))^2 + (\rho_{i,k}^{\ell+1,m} (v_{i,k}^- - v_{i,k}^{\ell+1,m,-}))^2 \right) \right\} \\ &= \arg \min_{v_{i,k}^+, v_{i,k}^- \forall i,k} \left\{ \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \left(\hat{H}^*(x_i, t_{k+1}, v_{i,k}^+, v_{i,k}^-) + \frac{\rho_{i,k}^{\ell+1,m}}{2\sigma} \left(v_{i,k}^+ - v_{i,k}^{\ell+1,m,+} - \sigma \frac{(D_x^+ \bar{\phi}^{\ell+1})_{i,k+1}}{\rho_{i,k}^{\ell+1,m}} \right)^2 \right. \right. \\ &\quad \left. \left. + \frac{\rho_{i,k}^{\ell+1,m}}{2\sigma} \left(v_{i,k}^- - v_{i,k}^{\ell+1,m,-} - \sigma \frac{(D_x^- \bar{\phi}^{\ell+1})_{i,k+1}}{\rho_{i,k}^{\ell+1,m}} \right)^2 \right) \right\}. \end{aligned} \quad (29)$$

Update $\rho_{i,k}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ by

$$\begin{aligned} (\rho_{i,k}^{\ell+1,m+1})_{i,k} &= \arg \max_{\rho_{i,k} \forall i,k: \rho_{i,k} \geq 0} \mathcal{L}_{full}((\bar{\phi}_{i,k}^{\ell+1})_{i,k}, (\rho_{i,k})_{i,k}, (v_{i,k}^{\ell+1,m+1,+})_{i,k}, (v_{i,k}^{\ell+1,m+1,-})_{i,k}) \\ &\quad - \frac{1}{2\sigma} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} (\rho_{i,k} - \rho_{i,k}^{\ell+1,m})^2 \\ &= (\max\{\mu_{i,k}^{\ell+1,m+1}, 0\})_{i,k}, \end{aligned} \quad (30)$$

where $\mu_{i,k}^{\ell+1,m+1}$ is a number defined by

$$\begin{aligned} \mu_{i,k}^{\ell+1,m+1} &= \rho_{i,k}^{\ell+1,m} + \sigma \left((D_t^- \bar{\phi}^{\ell+1})_{i,k+1} + v_{i,k}^{\ell+1,m+1,+} (D_x^+ \bar{\phi}^{\ell+1})_{i,k+1} + \right. \\ &\quad \left. v_{i,k}^{\ell+1,m+1,-} (D_x^- \bar{\phi}^{\ell+1})_{i,k+1} - \hat{H}^*(x_i, t_{k+1}, v_{i,k}^{\ell+1,m+1,+}, v_{i,k}^{\ell+1,m+1,-}) - \epsilon (D_{xx} \bar{\phi}^{\ell+1})_{i,k+1} \right). \end{aligned}$$

11 **end**

12 Set $v_{i,k}^{\ell+1,+} = v_{i,k}^{\ell+1,N_{inner},+}$, $v_{i,k}^{\ell+1,-} = v_{i,k}^{\ell+1,N_{inner},-}$, $\rho_{i,k}^{\ell+1} = \rho_{i,k}^{\ell+1,N_{inner}}$ for all $i = 1, \dots, n_x$;
 $k = 1, \dots, n_t - 1$.

13 **end**

14 Return $(\phi_{i,k}^{N_{outer}})_{i,k}$.

Algorithm 4: The proposed algorithm for solving (15)

Inputs : Step size $\tau, \sigma > 0$, error tolerance $\delta > 0$, inner maximal iteration number N_{inner} and outer maximal iteration number N_{outer} .

Outputs: Solution to the saddle point problem (15).

1 For each $i = 1, \dots, n_x; j = 1, \dots, n_y$, initialize the functions by $\phi_{i,j}^0(t) = g(x_i, y_j)$ for all $t \in [0, T]$,

$$\rho_{i,j}^0 \equiv c, v_{i,j}^{0,+} = v_{i,j}^{0,-} = w_{i,j}^{0,+} = w_{i,j}^{0,-} \equiv 0.$$

2 **for** $\ell = 0, 1, \dots, N_{outer} - 1$ **do**

3 Update the functions $\phi_{i,j}: [0, T] \rightarrow \mathbb{R}$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$ by

$$\phi^{\ell+1} = \phi^\ell + \tau(-\partial_t^2 - D_{xx} - D_{yy})^{-1} \left(\rho^\ell + D_x^-(\rho^\ell v^{\ell,+}) + D_x^+(\rho^\ell v^{\ell,-}) + D_y^-(\rho^\ell w^{\ell,+}) + D_y^+(\rho^\ell w^{\ell,-}) + \epsilon(D_{xx} + D_{yy})\rho^\ell \right), \quad (32)$$

where $(-\partial_t^2 - D_{xx} - D_{yy})^{-1}f$ (here each $f_{i,j}$ is a function of t) denotes the solution u to the ODE system $-\ddot{u}_{i,j} - (D_{xx}u + D_{yy}u)_{i,j} = f_{i,j}$ in $[0, T]$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$ with periodic spatial condition, Dirichlet initial condition $u_{i,j}(0) = 0$, and Neumann terminal condition $\dot{u}_{i,j}(T) = 0$.

4 **if** $\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \|\dot{\phi}_{i,j}^{\ell+1}(t) +$

$$\hat{H}(x_i, y_j, t, (D_x^+ \phi^{\ell+1})_{i,j}(t), (D_x^- \phi^{\ell+1})_{i,j}(t), (D_y^+ \phi^{\ell+1})_{i,j}(t), (D_y^- \phi^{\ell+1})_{i,j}(t)) - \epsilon(D_{xx} \phi^{\ell+1} + D_{yy} \phi^{\ell+1})_{i,j}(t)\|_1 \leq \delta$$

5 **then**

Return $\phi^{\ell+1}$.

6 **end**

7 Set $\bar{\phi}_{i,j}^{\ell+1} = 2\phi_{i,j}^{\ell+1} - \phi_{i,j}^\ell$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$.

8 Set $v_{i,j}^{\ell+1,0,+} = v_{i,j}^{\ell,+}, v_{i,j}^{\ell+1,0,-} = v_{i,j}^{\ell,-}, w_{i,j}^{\ell+1,0,+} = w_{i,j}^{\ell,+}, w_{i,j}^{\ell+1,0,-} = w_{i,j}^{\ell,-}, \rho_{i,j}^{\ell+1,0} = \rho_{i,j}^\ell$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$.

9 **for** $m = 0, 1, \dots, N_{inner} - 1$ **do**

10 Update $v_{i,j}^+, v_{i,j}^-, w_{i,j}^+$ and $w_{i,j}^-$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$ by

$$\begin{aligned} & (v^{\ell+1,m+1,+}, v^{\ell+1,m+1,-}, w^{\ell+1,m+1,+}, w^{\ell+1,m+1,-}) \\ &= \arg \min_{v_{i,j}^+, v_{i,j}^-, w_{i,j}^+, w_{i,j}^-} \left\{ \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \int_0^T \hat{H}^*(x_i, t, v_{i,j}^+(t), v_{i,j}^-(t), w_{i,j}^+(t), w_{i,j}^-(t)) \right. \\ & \quad + \frac{\rho_{i,j}^{\ell+1,m}(t)}{2\sigma} \left((v_{i,j}^+(t) - v_{i,j}^{\ell+1,m,+}(t) - \sigma \frac{(D_x^+ \bar{\phi}^{\ell+1})_{i,j}(t)}{\rho_{i,j}^{\ell+1,m}(t)})^2 + (v_{i,j}^-(t) - v_{i,j}^{\ell+1,m,-}(t) - \sigma \frac{(D_x^- \bar{\phi}^{\ell+1})_{i,j}(t)}{\rho_{i,j}^{\ell+1,m}(t)})^2 \right. \\ & \quad \left. \left. + (w_{i,j}^+(t) - w_{i,j}^{\ell+1,m,+}(t) - \sigma \frac{(D_y^+ \bar{\phi}^{\ell+1})_{i,j}(t)}{\rho_{i,j}^{\ell+1,m}(t)})^2 + (w_{i,j}^-(t) - w_{i,j}^{\ell+1,m,-}(t) - \sigma \frac{(D_y^- \bar{\phi}^{\ell+1})_{i,j}(t)}{\rho_{i,j}^{\ell+1,m}(t)})^2 \right) dt \right\}. \end{aligned} \quad (33)$$

Update $\rho_{i,j}$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y$ by

$$\rho^{\ell+1,m+1} = (\max\{\mu_{i,j}^{\ell+1,m+1}, 0\})_{i,j}, \quad (34)$$

where $\mu_{i,j}^{\ell+1,m+1}$ is a function defined by

$$\begin{aligned} \mu_{i,j}^{\ell+1,m+1} &= \rho_{i,j}^{\ell+1,m} + \sigma \left(\dot{\bar{\phi}}_{i,j}^{\ell+1}(t) + v_{i,j}^{\ell+1,m+1,+}(t)(D_x^+ \bar{\phi}^{\ell+1})_{i,j}(t) + v_{i,j}^{\ell+1,m+1,-}(t)(D_x^- \bar{\phi}^{\ell+1})_{i,j}(t) \right. \\ & \quad + w_{i,j}^{\ell+1,m+1,+}(t)(D_y^+ \bar{\phi}^{\ell+1})_{i,j}(t) + w_{i,j}^{\ell+1,m+1,-}(t)(D_y^- \bar{\phi}^{\ell+1})_{i,j}(t) - \hat{H}^*(x_i, y_j, t, v_{i,j}^{\ell+1,m+1,+}(t), \\ & \quad \left. v_{i,j}^{\ell+1,m+1,-}(t), w_{i,j}^{\ell+1,m+1,+}(t), w_{i,j}^{\ell+1,m+1,-}(t)) - \epsilon(D_{xx} \bar{\phi}^{\ell+1} + D_{yy} \bar{\phi}^{\ell+1})_{i,j}(t) \right). \end{aligned}$$

11 **end**

12 Set $v^{\ell+1,+} = v^{\ell+1,N_{inner},+}, v^{\ell+1,-} = v^{\ell+1,N_{inner},-}, w^{\ell+1,+} = w^{\ell+1,N_{inner},+}, w^{\ell+1,-} = w^{\ell+1,N_{inner},-}, \rho^{\ell+1} = \rho^{\ell+1,N_{inner}}$.

13 **end**

14 Return $\phi^{N_{outer}}$.

A.5 Two-dimensional fully-discrete method

To solve the fully-discrete equation (16), we propose solving the saddle point problem (17), whose first order optimality condition is

$$\left\{ \begin{array}{l} (D_t^- \phi)_{i,j,k} + \hat{H}(x_i, y_j, t_k, (D_x^+ \phi)_{i,j,k}, (D_x^- \phi)_{i,j,k}, (D_y^+ \phi)_{i,j,k}, (D_y^- \phi)_{i,j,k}) \leq \epsilon(D_{xx}\phi)_{i,j,k} + \epsilon(D_{yy}\phi)_{i,j,k}, \\ \quad i = 1, \dots, n_x; j = 1, \dots, n_y; k = 2, \dots, n_t, \\ (D_t^+ \rho)_{i,j,k} + (D_x^-(\rho v^+))_{i,j,k} + (D_x^+(\rho v^-))_{i,j,k} + (D_y^-(\rho w^+))_{i,j,k} + (D_y^+(\rho w^-))_{i,j,k} + \epsilon(D_{xx}\rho)_{i,j,k} + \epsilon(D_{yy}\rho)_{i,j,k} = 0, \\ \quad i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1, \\ (v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-) = \nabla_{(p_1^+, p_1^-, p_2^+, p_2^-)} \hat{H}(x_i, y_j, t_{k+1}, (D_x^+ \phi)_{i,j,k+1}, (D_x^- \phi)_{i,j,k+1}, (D_y^+ \phi)_{i,j,k+1}, (D_y^- \phi)_{i,j,k+1}), \\ \quad i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1, \\ \phi_{i,j,1} = g(x_i, y_j), \quad \rho_{i,j,n_t} = c, \quad i = 1, \dots, n_x; j = 1, \dots, n_y, \\ \rho_{i,k-1} \left((D_t^- \phi)_{i,j,k} + \hat{H}(x_i, y_j, t_k, (D_x^+ \phi)_{i,j,k}, (D_x^- \phi)_{i,j,k}, (D_y^+ \phi)_{i,j,k}, (D_y^- \phi)_{i,j,k}) - \epsilon(D_{xx}\phi)_{i,j,k} - \epsilon(D_{yy}\phi)_{i,j,k} \right) = 0, \\ \quad i = 1, \dots, n_x; j = 1, \dots, n_y; k = 2, \dots, n_t. \end{array} \right. \quad (35)$$

The proposed algorithm for solving (17) is shown in Algorithm 5. In the algorithm, we use \mathcal{L}_{full} to denote the objective function in (13). When updating ϕ , the term $(D_t^+ \rho)_{i,j,n_t-1}$ denotes $\frac{c - \rho_{i,j,n_t-1}}{\Delta t}$, which is consistent with the terminal condition for ρ in (35). For simplicity, we use α to denote $(\alpha_{i,j,k})_{i,j,k}$ for any quantity α . This algorithm is similar to the one-dimensional case, so we omitted some details. We choose the quadratic function for the penalty term of all functions besides ϕ . For ϕ , our choice of penalty term for preconditioning is $\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{i=2}^{n_t} \left(((D_t^- \phi)_{i,j,k} - (D_t^- \phi^\ell)_{i,j,k})^2 + ((D_x^+ \phi)_{i,j,k} - (D_x^+ \phi^\ell)_{i,j,k})^2 + ((D_y^+ \phi)_{i,j,k} - (D_y^+ \phi^\ell)_{i,j,k})^2 \right)$.

B An equivalent formulation

Through the substitution of the variable $m = \rho v$, we obtain an alternative expression for equation (4) as follows:

$$\min_{\phi(x,0)=g(x)} \max_{\substack{\rho, m \\ \rho \geq 0}} \int_0^T \int_{\Omega} \rho(x,t) \left(\frac{\partial \phi(x,t)}{\partial t} - \epsilon \Delta_x \phi(x,t) \right) + m(x,t) \nabla_x \phi(x,t) - \rho(x,t) H^* \left(x, t, \frac{m(x,t)}{\rho(x,t)} \right) dx dt - c \int \phi(x,T) dx. \quad (39)$$

In contrast to the saddle point formulation (4), this revised equation conforms to the conventional PDHG-applicable saddle point formulation (as detailed in [20]), which thereby ensures convergence for the corresponding algorithm. Analogously, the semi-discrete and fully discrete counterparts discussed in Sections B.2 and B.4 also follow this structure, thereby endowing their corresponding algorithms with theoretical convergence guarantees. For more details, we refer readers to the following remark and also Remark B.3.

Remark B.1. *The convergence of the PDHG algorithm for equation (39), as well as its associated semi-discrete and fully-discrete counterparts in the specialized context of separable and shifted 1-homogeneous Hamiltonians described in Appendices B.2 and B.4, can be established based on established theory [20]. These specific saddle point problems adhere to the conventional format outlined in [20], characterized by bilinear interactions between primal and dual variables, while other terms exhibit convexity concerning ϕ or concavity concerning (ρ, m) . To be more precise, in scenarios where H exhibits separability and shifted 1-homogeneity with respect to p , the primal and dual optimization problems in equation (47) can be reformulated as linear programming problems involving (ρ, m) or ϕ . Additionally, explicit updating formulas for both primal and dual variables can be derived in this specific case (for more information, refer to Appendices B.2 and B.4).*

These explicit formulas, which remove the inner loop in Algorithm 1, are attainable due to the simplification of the term $\rho(x,t) H^ \left(x, t, \frac{m(x,t)}{\rho(x,t)} \right)$ (and its corresponding nonlinear terms in both semi-discrete and fully-discrete scenarios), resulting in linear objective functions and linear constraints. Nevertheless, in more general cases where the intricate nonlinear term cannot be simplified, there is no feasible approach to derive an explicit updating formula for (ρ, m) . In such instances, Algorithm 1 and its semi-discrete and fully-discrete variations are more suitable.*

Algorithm 5: The proposed algorithm for solving (17)

Inputs : Stepsize $\tau, \sigma > 0$, error tolerance $\delta > 0$, inner maximal iteration number N_{inner} and outer maximal iteration number N_{outer} .

Outputs: Solution to the saddle point problem (17).

1 For each $i = 1, \dots, n_x; j = 1, \dots, n_y$, initialize the matrices by $\phi_{i,j,k}^0 = g(x_i, y_j)$ for $k = 1, \dots, n_t$,

$\rho_{i,j,k}^0 = c, v_{i,j,k}^{0,+} = v_{i,j,k}^{0,-} = w_{i,j,k}^{0,+} = w_{i,j,k}^{0,-} = 0$ for $k = 1, \dots, n_t - 1$.

2 **for** $\ell = 0, 1, \dots, N_{outer} - 1$ **do**

3 Update the tensor element $\phi_{i,j,k}$ for $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t$ by

$$\begin{aligned} \phi^{\ell+1} = \phi^\ell + \tau(-D_{tt} - D_{xx} - D_{yy})^{-1} & \left(D_t^+ \rho^\ell + D_x^-(\rho^\ell v^{\ell,+}) + D_x^+(\rho^\ell v^{\ell,-}) \right. \\ & \left. + D_y^-(\rho^\ell w^{\ell,+}) + D_y^+(\rho^\ell w^{\ell,-}) + \epsilon D_{xx} \rho^\ell + \epsilon D_{yy} \rho^\ell \right), \end{aligned} \quad (36)$$

where $(-D_{tt} - D_{xx} - D_{yy})^{-1}f$ (for a tensor f with elements $f_{i,j,k}$ and the linear operator $D_{tt}f = (\frac{f_{i,j,k-1} - 2f_{i,j,k} + f_{i,j,k+1}}{\Delta t^2})_{i,j,k}$) denotes the solution u to the linear system $-(D_{tt}u)_{i,j,k+1} - (D_{xx}u + D_{yy}u)_{i,j,k+1} = f_{i,j,k}$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1$ with periodic spatial condition, Dirichlet initial condition $u_{i,j,1} = 0$, and Neumann terminal condition $u_{i,j,n_t+1} = u_{i,j,n_t}$.

4 **if** $\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=2}^{n_t} |(D_t^- \phi^{\ell+1})_{i,j,k} +$

$\hat{H}(x_i, y_j, t_k, (D_x^+ \phi^{\ell+1})_{i,j,k}, (D_x^- \phi^{\ell+1})_{i,j,k}, (D_y^+ \phi^{\ell+1})_{i,j,k}, (D_y^- \phi^{\ell+1})_{i,j,k}) - \epsilon(D_{xx} \phi^{\ell+1} + D_{yy} \phi^{\ell+1})_{i,j,k}| \leq \delta$ **then**

5 | Return $\phi^{\ell+1}$.

6 **end**

7 Set $\bar{\phi}^{\ell+1} = 2\phi^{\ell+1} - \phi^\ell$.

8 Set $v^{\ell+1,0,+} = v^{\ell,+}, v^{\ell+1,0,-} = v^{\ell,-}, w^{\ell+1,0,+} = w^{\ell,+}, w^{\ell+1,0,-} = w^{\ell,-}, \rho^{\ell+1,0} = \rho^\ell$.

9 **for** $m = 0, 1, \dots, N_{inner} - 1$ **do**

10 Update $v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1$ by

$$\begin{aligned} & (v^{\ell+1,m+1,+}, v^{\ell+1,m+1,-}, w^{\ell+1,m+1,+}, w^{\ell+1,m+1,-}) \\ & = \arg \min_{v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-} \left\{ \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t-1} \left(\hat{H}^*(x_i, y_j, t_{k+1}, v_{i,j,k}^+, v_{i,j,k}^-, w_{i,j,k}^+, w_{i,j,k}^-) \right. \right. \\ & \quad \left. \left. + \frac{\rho_{i,j,k}^{\ell+1,m}}{2\sigma} \left((v_{i,j,k}^+ - v_{i,j,k}^{\ell+1,m,+} - \sigma \frac{(D_x^+ \bar{\phi}^{\ell+1})_{i,j,k+1}}{\rho_{i,j,k}^{\ell+1,m}})^2 + (v_{i,j,k}^- - v_{i,j,k}^{\ell+1,m,-} - \sigma \frac{(D_x^- \bar{\phi}^{\ell+1})_{i,j,k+1}}{\rho_{i,j,k}^{\ell+1,m}})^2 \right) \right. \right. \\ & \quad \left. \left. + (w_{i,j,k}^+ - w_{i,j,k}^{\ell+1,m,+} - \sigma \frac{(D_y^+ \bar{\phi}^{\ell+1})_{i,j,k+1}}{\rho_{i,j,k}^{\ell+1,m}})^2 + (w_{i,j,k}^- - w_{i,j,k}^{\ell+1,m,-} - \sigma \frac{(D_y^- \bar{\phi}^{\ell+1})_{i,j,k+1}}{\rho_{i,j,k}^{\ell+1,m}})^2 \right) \right\}. \end{aligned} \quad (37)$$

11 Update $\rho_{i,j,k}$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1$ by

$$\rho_{i,j,k}^{\ell+1,m+1} = \max\{\mu_{i,j,k}^{\ell+1,m+1}, 0\}, \quad (38)$$

where $\mu_{i,j,k}^{\ell+1,m+1}$ is a number defined by

$$\begin{aligned} \mu_{i,j,k}^{\ell+1,m+1} = \rho_{i,j,k}^{\ell+1,m} + \sigma & \left((D_t^- \bar{\phi})_{i,j,k+1}^{\ell+1} + v_{i,j,k}^{\ell+1,m+1,+} (D_x^+ \bar{\phi}^{\ell+1})_{i,j,k+1} + v_{i,j,k}^{\ell+1,m+1,-} (D_x^- \bar{\phi})_{i,j,k+1}^{\ell+1} \right. \\ & \left. + w_{i,j,k}^{\ell+1,m+1,+} (D_y^+ \bar{\phi}^{\ell+1})_{i,j,k+1} + w_{i,j,k}^{\ell+1,m+1,-} (D_y^- \bar{\phi})_{i,j,k+1}^{\ell+1} - \hat{H}^*(x_i, y_j, t_{k+1}, \right. \\ & \left. v_{i,j,k}^{\ell+1,m+1,+}, v_{i,j,k}^{\ell+1,m+1,-}, w_{i,j,k}^{\ell+1,m+1,+}, w_{i,j,k}^{\ell+1,m+1,-}) - \epsilon(D_{xx} \bar{\phi} + D_{yy} \bar{\phi})_{i,j,k+1}^{\ell+1} \right). \end{aligned}$$

12 **end**

13 Set $v^{\ell+1,+} = v^{\ell+1,N_{inner},+}, v^{\ell+1,-} = v^{\ell+1,N_{inner},-}, w^{\ell+1,+} = w^{\ell+1,N_{inner},+},$

$w^{\ell+1,-} = w^{\ell+1,N_{inner},-}, \rho^{\ell+1} = \rho^{\ell+1,N_{inner}}$.

14 **end**

15 Return $\phi^{N_{outer}}$.

We will now provide a summary of the relevant semi-discrete and fully-discrete formulations, along with their corresponding algorithms, for both one-dimensional and two-dimensional cases in the upcoming sections.

B.1 One-dimensional semi-discrete and fully-discrete methods

For the one-dimensional case, we first discretize the spatial domain $[a, b]$ and get the following semi-discrete saddle point problem

$$\min_{\substack{\phi_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\substack{\rho_i, m_i \forall i \\ \rho_i \geq 0}} \int_0^T \sum_{i=1}^{n_x} \left(\rho_i(t) \dot{\phi}_i(t) + m_i(t) (D_x^+ \phi)_i(t) - \epsilon \rho_i(t) (D_{xx} \phi)_i(t) \right) - \hat{L}((x_i)_i, t, (\rho_i(t))_i, (m_i(t))_i) dt - c \sum_{i=1}^{n_x} \phi_i(T), \quad (40)$$

where $\hat{L}: \mathbb{R}^{n_x} \times [0, +\infty) \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ is defined by

$$\hat{L}((x_i)_i, t, (\rho_i)_i, (m_i)_i) = \max_{d_i \forall i} \sum_{i=1}^{n_x} (m_i d_i - \rho_i \hat{H}(x_i, t, d_i, d_{i-1})).$$

The derivation of this formula is given as follows

$$\begin{aligned} & \min_{\phi \text{ satisfying (10)}} -c \sum_{i=1}^{n_x} \phi_i(T) \\ &= \min_{\substack{\phi_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\rho_i \forall i} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) + \hat{H}(x_i, t, (D_x^+ \phi)_i(t), (D_x^- \phi)_i(t)) - \epsilon (D_{xx} \phi)_i(t) \right) dt - c \sum_{i=1}^{n_x} \phi_i(T) \\ &= \min_{\substack{\phi_i, d_i \forall i \\ \phi_i(0)=g(x_i) \\ d_i(t) = (D_x^+ \phi)_i(t)}} \max_{\rho_i \forall i} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) + \hat{H}(x_i, t, d_i(t), d_{i-1}(t)) - \epsilon (D_{xx} \phi)_i(t) \right) dt - c \sum_{i=1}^{n_x} \phi_i(T) \\ &= \min_{\substack{\phi_i, d_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\rho_i, m_i \forall i} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) + \hat{H}(x_i, t, d_i(t), d_{i-1}(t)) - \epsilon (D_{xx} \phi)_i(t) \right) + m_i(t) ((D_x^+ \phi)_i(t) - d_i(t)) dt - c \sum_{i=1}^{n_x} \phi_i(T) \\ &= \min_{\substack{\phi_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\rho_i, m_i \forall i} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) - \epsilon (D_{xx} \phi)_i(t) \right) + m_i(t) (D_x^+ \phi)_i(t) dt - c \sum_{i=1}^{n_x} \phi_i(T) \\ &\quad - \int_0^T \max_{d_i(t)} \sum_{i=1}^{n_x} (m_i(t) d_i(t) - \rho_i(t) \hat{H}(x_i, t, d_i(t), d_{i-1}(t))) dt \\ &= \min_{\substack{\phi_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\rho_i, m_i \forall i} \int_0^T \sum_{i=1}^{n_x} \rho_i(t) \left(\dot{\phi}_i(t) - \epsilon (D_{xx} \phi)_i(t) \right) + m_i(t) (D_x^+ \phi)_i(t) dt - c \sum_{i=1}^{n_x} \phi_i(T) - \int_0^T \hat{L}((x_i(t))_i, t, (\rho_i(t))_i, (m_i(t))_i) dt. \end{aligned} \quad (41)$$

As we explained in Remark 2.1 and Appendix A.2, when the corresponding mean-field control problem has a solution including a positive density ρ , we can add the constraint $\rho \geq 0$ to the saddle point problem, from which we obtain (40).

Remark B.2. When $\hat{H}(x, t, p^+, p^-)$ takes on a “separable” form, denoted by $\hat{H}(x, t, p^+, p^-) = \hat{H}_1(x, t, p^+) + \hat{H}_2(x, t, p^-)$, the following equality holds:

$$\begin{aligned} & \max_{d \in \mathbb{R}^{n_x}} \sum_{i=1}^{n_x} (m_i d_i - \rho_i \hat{H}(x_i, t, d_i, d_{i-1})) = \max_{d \in \mathbb{R}^{n_x}} \sum_{i=1}^{n_x} (m_i d_i - \rho_i (\hat{H}_1(x_i, t, d_i) + \hat{H}_2(x_i, t, d_{i-1}))) \\ &= \max_{d \in \mathbb{R}^{n_x}} \sum_{i=1}^{n_x} m_i d_i - (\rho_i \hat{H}_1(x_i, t, d_i) + \rho_{i+1} \hat{H}_2(x_{i+1}, t, d_i)) = \sum_{i=1}^{n_x} (\rho_i \hat{H}_{1,i,t} + \rho_{i+1} \hat{H}_{2,i+1,t})^*(m_i), \end{aligned} \quad (42)$$

where $\hat{H}_{1,i,t}$ and $\hat{H}_{2,i,t}$ denotes the functions $p \mapsto \hat{H}_1(x_i, t, p)$ and $p \mapsto \hat{H}_2(x_i, t, p)$, respectively. The corresponding saddle point problem becomes

$$\min_{\substack{\phi_i \forall i \\ \phi_i(0)=g(x_i)}} \max_{\substack{\rho_i, m_i \forall i \\ \rho_i \geq 0}} \int_0^T \sum_{i=1}^{n_x} \left(\rho_i(t) \dot{\phi}_i(t) + m_i(t) (D_x^+ \phi)_i(t) - \epsilon \rho_i(t) (D_{xx} \phi)_i(t) - (\rho_i(t) \hat{H}_{1,i,t} + \rho_{i+1}(t) \hat{H}_{2,i+1,t})^*(m_i(t)) \right) dt - c \sum_{i=1}^{n_x} \phi_i(T). \quad (43)$$

For instance, in the context of applying the Engquist-Osher scheme [79, 38, 39], the function $\hat{H}_{1,i,t}$ corresponds to the non-increasing part of $H(x,t,\cdot)$, while $\hat{H}_{2,i,t}$ corresponds to the non-decreasing part of $H(x,t,\cdot)$. Under this framework, the link between (11) and (43) involves the transformation of variables: $\rho_i v_i^+ = \min\{m_i, 0\}$ and $\rho_i v_i^- = \max\{m_{i-1}, 0\}$. Furthermore, if H exhibits 1-homogeneity and convexity with respect to p , the function $(\rho_i(t)\hat{H}_{1,i,t} + \rho_{i+1}(t)\hat{H}_{2,i+1,t})^*$ becomes associated with the indicator function, which notably simplifies the computation process. For further details, please refer to Appendix B.2.

Next, we apply implicit time discretization and employ $(D_t^- \phi)_{i,j,k}$ to estimate $\dot{\phi}_{i,j}(t)$. Consequently, the saddle point formulation (58) transforms into the following expression (It's important to observe that the index k for ρ , m^1 , and m^2 ranges from 1 to $n_t - 1$, while for ϕ , it spans from 1 to n_t).

$$\min_{\substack{\phi_{i,k} \forall i,k \\ \phi_{i,1}=g(x_i)}} \max_{\substack{\rho_{i,k}, m_{i,k} \forall i,k \\ \rho_{i,k} \geq 0}} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \left(\rho_{i,k} (D_t^- \phi)_{i,k+1} + m_{i,k} (D_x^+ \phi)_{i,k+1} - \epsilon \rho_{i,k} (D_{xx} \phi)_{i,k+1} \right) - \hat{L}((x_i)_i, (t_k)_k, (\rho_{i,k})_{i,k}, (m_{i,k})_{i,k}) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \phi_{i,n_t}, \quad (44)$$

where we abuse the notation \hat{L} and define $\hat{L}: \mathbb{R}^{n_x} \times \mathbb{R}^{n_t} \times \mathbb{R}^{n_x \times (n_t-1)} \times \mathbb{R}^{n_x \times (n_t-1)} \rightarrow \mathbb{R}$ by

$$\hat{L}((x_i)_i, (t_k)_k, (\rho_{i,k})_{i,k}, (m_{i,k})_{i,k}) = \max_{d_{i,k} \in \mathbb{R} \forall i,k} \sum_{k=1}^{n_t-1} \sum_{i=1}^{n_x} (m_{i,k} d_{i,k} - \rho_{i,k} \hat{H}(x_i, t_{k+1}, d_{i,k}, d_{i-1,k})).$$

The discussion in Remark B.2 also applies to this fully-discrete formula with straightforward modifications.

While it is possible to decouple the formula for \hat{L} with respect to k , this decoupling cannot be achieved with respect to i . This discrepancy constitutes a significant limitation of this formulation when compared to (13), wherein the nonlinear term within the objective function can be effortlessly decoupled concerning both i and k . This property facilitates the parallel computation of the updates for ρ , v^+ , v^- in Algorithm 3. However, in certain special scenarios (such as when H is separable and shifted 1-homogeneous with respect to p , as discussed in the subsequent section), the function \hat{L} takes on a particular structure, enabling the derivation of an explicit joint updating formula for (ρ, m) .

B.2 One-dimensional special case: 1-homogeneous Hamiltonian

In this section, we consider the one-dimensional HJ PDE whose Hamiltonian H is in the form of

$$H(x, t, p) = \gamma(x, t)|p| + f(x, t), \quad (45)$$

with the assumption that $\gamma(x, t) > 0$ for any $x \in \mathbb{R}$, $t \in [0, T]$. We apply the Engquist-Osher scheme [79, 38, 39], in which we have $\hat{H}(x, t, p_+, p_-) = \hat{H}_-(x, t, p_+) + \hat{H}_+(x, t, p_-) + f(x, t)$, where $\hat{H}_-(x, t, p) = \gamma(x, t) \max\{-p, 0\}$ and $\hat{H}_+(x, t, p) = \gamma(x, t) \max\{p, 0\}$. (Notably, these functions \hat{H}_+ and \hat{H}_- draw parallels to the ReLU activation function within the domain of neural networks.) In this scenario, we arrive at the following formula for the function \hat{L} in (44):

$$\begin{aligned} \hat{L}((x_i)_i, (t_k)_k, (\rho_{i,k})_{i,k}, (m_{i,k})_{i,k}) &= \max_{d_{i,k} \in \mathbb{R} \forall i,k} \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} (m_{i,k} d_{i,k} - \rho_{i,k} \hat{H}(x_i, t_{k+1}, d_{i,k}, d_{i-1,k})) \\ &= \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} (-\rho_{i,k} f(x_i, t_{k+1}) + I_{[-\rho_{i,k}\gamma(x_i, t_{k+1}), \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1})]}(m_{i,k})), \end{aligned} \quad (46)$$

where I_C represents the indicator function of set C , which is 0 when the variable is within C , and $+\infty$ otherwise. Then, the saddle point problem becomes

$$\min_{\substack{\phi_{i,k} \forall i,k \\ \phi_{i,1}=g(x_i)}} \max_{\substack{\rho_{i,k}, m_{i,k} \forall i,k \\ \rho_{i,k} \geq 0}} \left\{ \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} \left(\rho_{i,k} (D_t^- \phi)_{i,k+1} + m_{i,k} (D_x^+ \phi)_{i,k+1} - \epsilon \rho_{i,k} (D_{xx} \phi)_{i,k+1} - \rho_{i,k} f(x_i, t_{k+1}) \right) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \phi_{i,n_t} : \right. \\ \left. -\rho_{i,k}\gamma(x_i, t_{k+1}) \leq m_{i,k} \leq \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1}), \forall i = 1, \dots, n_x; k = 1, \dots, n_t - 1 \right\}. \quad (47)$$

As elucidated in Remark B.1, the objective function within the saddle point problem (47) adheres to the conventional form within the theory in [20]. Furthermore, the sole nonlinear term in this context is the indicator function, resulting in explicit formulas for updating ϕ , ρ , and m . It's important to note that this approach can be extended to encompass any Hamiltonian that is convex and shifted 1-homogeneous with respect to p .

We employ the PDHG method to tackle (47). A comprehensive outline of the algorithm is encapsulated in Algorithm 6. In the algorithm, we conveniently employ the notation \mathcal{L}_{full} to represent the objective function in (47). During the update of ϕ , the component $(D_t^+ \rho)_{i,n_t-1}$ represents $\frac{c-\rho_{i,n_t-1}}{\Delta t}$, aligning with the terminal condition for ρ . As we possess an explicit formula for the joint update of (ρ, m) , there is no need for an inner loop. This method is supported by a theoretical guarantee, as explained in Remark B.1, and this is further detailed in the following remark.

Remark B.3. *According to the theory in [20], the algorithm converges when the chosen step sizes τ and σ adhere to the condition $\tau\sigma\|K\|^2 < 1$, where K represents the bilinear operator. In our case, we have*

$$\begin{aligned} |K((\rho, m), \phi)| &= \left| \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} (\rho_{i,k}(D_t^- \phi)_{i,k+1} + m_{i,k}(D_x^+ \phi)_{i,k+1} - \epsilon \rho_{i,k}(D_{xx} \phi)_{i,k+1}) \right| \\ &\leq \|\rho\| \|D_t^- \phi\| + \|m\| \|D_x^+ \phi\| + \frac{2\epsilon}{\Delta x} \|\rho\| \|D_x^+ \phi\| \\ &\leq 2 \left(1 + \frac{2\epsilon}{\Delta x} \right) \|(\rho, m)\| \| (D_t^- \phi, D_x^+ \phi) \|. \end{aligned}$$

Consequently, we derive $\|K\| \leq 2 \left(1 + \frac{2\epsilon}{\Delta x} \right)$, allowing for the selection of step sizes τ and σ that satisfy $\tau\sigma < \left(2 + \frac{4\epsilon}{\Delta x} \right)^{-2}$.

Remark B.4. *As elaborated in Remark B.2, the connection between (13) and (44) involves the variable transformations $\rho_{i,k} v_{i,k}^+ = \min\{m_{i,k}, 0\}$ and $v_{i,k}^- = \max\{m_{i-1,k}, 0\}$. Algorithm 6 is closely related to Algorithm 3, though there exist some distinctions.*

The penalties pertaining to ρ in Algorithms 6 and 3 are identical. The primary focus is directed towards the penalty related to m , which is $\sum_{i,k} (m_{i,k} - m_{i,k}^\ell)^2$. If both $m_{i,k}$ and $m_{i,k}^\ell$ are positive, the (i, k) -th term in the summation becomes $(\rho_{i,k} v_{i,k}^- - \rho_{i,k}^\ell v_{i,k}^{\ell,-})^2$, closely resembling the penalty term $(\rho_{i,k}^\ell v_{i,k}^- - \rho_{i,k} v_{i,k}^{\ell,-})^2$ in Algorithm 3. Similarly, when both $m_{i,k}$ and $m_{i,k}^\ell$ are negative, $(m_{i,k} - m_{i,k}^\ell)^2$ becomes $(\rho_{i,k} v_{i,k}^+ - \rho_{i,k}^\ell v_{i,k}^{\ell,+})^2$, analogous to the penalty term for v^+ in Algorithm 3.

Therefore, if the sign of m remains constant between the previous and current steps, Algorithms 6 and 3 exhibit similarities.

The update procedure for ϕ remains as previously described. We will now detail the process of deriving updates for ρ and m .

B.2.1 The detailed derivation of updating ρ and m in Algorithm 6

When H adheres to the conditions specified in (45), the updates for ρ and m require solving the subsequent problem:

$$\begin{aligned} \min_{\substack{\rho_{i,k}, m_{i,k} \forall i,k \\ \rho_{i,k} \geq 0}} \left\{ - \sum_{i=1}^{n_x} \sum_{k=1}^{n_t-1} (\rho_{i,k}(D_t^- \bar{\phi}^{\ell+1})_{i,k+1} + m_{i,k}(D_x^+ \bar{\phi}^{\ell+1})_{i,k+1} - \epsilon \rho_{i,k}(D_{xx} \bar{\phi}^{\ell+1})_{i,k+1} + \rho_{i,k} f(x_i, t_{k+1})) + \frac{1}{2\sigma} \sum_{k=1}^{n_t-1} \sum_{i=1}^{n_x} (\rho_{i,k} - \rho_{i,k}^\ell)^2 \right. \\ \left. + \frac{1}{2\sigma} \sum_{k=1}^{n_t-1} \sum_{i=1}^{n_x} (m_{i,k} - m_{i,k}^\ell)^2 : -\rho_{i,k} \gamma(x_i, t_{k+1}) \leq m_{i,k} \leq \rho_{i+1,k} \gamma(x_{i+1}, t_{k+1}), \forall i = 1, \dots, n_x; k = 1, \dots, n_t - 1 \right\}. \end{aligned} \quad (49)$$

Define z and α as in Algorithm 6, i.e., $z_{i,k}^\ell = m_{i,k}^\ell + \sigma(D_x^+ \bar{\phi}^{\ell+1})_{i,k+1}$, and $\alpha_{i,k}^\ell = \rho_{i,k}^\ell + \sigma((D_t^- \bar{\phi}^{\ell+1})_{i,k+1} + f(x_i, t_{k+1}) - \epsilon(D_{xx} \bar{\phi}^{\ell+1})_{i,k+1})$. Then, we can rewrite (49) as

$$\min_{\substack{\rho_{i,k}, m_{i,k} \forall i,k \\ \rho_{i,k} \geq 0}} \left\{ \frac{1}{2\sigma} \sum_{k=1}^{n_t-1} \sum_{i=1}^{n_x} ((\rho_{i,k} - \alpha_{i,k}^\ell)^2 + (m_{i,k} - z_{i,k}^\ell)^2) : -\rho_{i,k} \gamma(x_i, t_{k+1}) \leq m_{i,k} \leq \rho_{i+1,k} \gamma(x_{i+1}, t_{k+1}), \forall i = 1, \dots, n_x; k = 1, \dots, n_t - 1 \right\}. \quad (50)$$

Algorithm 6: The proposed algorithm for solving (47)

Inputs : Stepsize $\tau, \sigma > 0$, error tolerance $\delta > 0$, and the maximal iteration number N .

Outputs: Solution to the saddle point problem (47).

1 For each $i = 1, \dots, n_x$, initialize the matrices by $\phi_{i,k}^0 = g(x_i)$ for $k = 1, \dots, n_t$, $\rho_{i,k}^0 = c$, $m_{i,k}^0 = 0$ for $k = 1, \dots, n_t - 1$.

2 **for** $\ell = 0, 1, \dots, N - 1$ **do**

3 Update the matrix $\phi_{i,k}$ for $i = 1, \dots, n_x$; $k = 1, \dots, n_t$ by

$$\begin{aligned} (\phi_{i,k}^{\ell+1})_{i,k} &= \arg \min_{\substack{\phi_{i,k} \forall i,k \\ \phi_{i,1} = g(x_i)}} \mathcal{L}_{full}((\phi_{i,k})_{i,k}, (\rho_{i,k}^\ell)_{i,k}, (v_{i,k}^{\ell,+})_{i,k}, (v_{i,k}^{\ell,-})_{i,k}) \\ &\quad + \frac{1}{2\tau} \sum_{i=1}^{n_x} \sum_{k=2}^{n_t} \left(((D_t^- \phi)_{i,k} - (D_t^- \phi^\ell)_{i,k})^2 + ((D_x^+ \phi)_{i,k} - (D_x^+ \phi^\ell)_{i,k})^2 \right) \quad (48) \\ &= (\phi_{i,k}^\ell)_{i,k} + \tau (-D_{tt} - D_{xx})^{-1} \left(D_t^+ \rho^\ell + D_x^- m^\ell + \epsilon D_{xx} \rho^\ell \right), \end{aligned}$$

where $(-D_{tt} - D_{xx})^{-1}(f_{i,k})_{i,k}$ (for a matrix f with elements $f_{i,k}$ and the linear operator

$D_{tt}f = (\frac{f_{i,k-1} - 2f_{i,k} + f_{i,k+1}}{\Delta t^2})_{i,k}$) denotes the solution u to the linear system

$-(D_{tt}u)_{i,k+1} - (D_{xx}u)_{i,k+1} = f_{i,k}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ with periodic spatial condition, Dirichlet initial condition $u_{i,1} = 0$, and Neumann terminal condition $u_{i,n_t+1} = u_{i,n_t}$.

4 **if** $\sum_{i=1}^{n_x} \sum_{k=2}^{n_t} |(D_t^- \phi^{\ell+1})_{i,k} + \hat{H}(x_i, t_k, (D_x^+ \phi^{\ell+1})_{i,k}, (D_x^- \phi^{\ell+1})_{i,k}) - \epsilon (D_{xx} \phi^{\ell+1})_{i,k}| \leq \delta$ **then**

5 | Return $(\phi_{i,k}^{\ell+1})_{i,k}$.

6 **end**

7 Set $\bar{\phi}_{i,k}^{\ell+1} = 2\phi_{i,k}^{\ell+1} - \phi_{i,k}^\ell$ for all $i = 1, \dots, n_x$; $k = 2, \dots, n_t$.

8 Update $\rho_{i,k}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ by

$$\begin{aligned} \rho_{i,k}^{\ell+1} &= \arg \min_{y \in \mathcal{A}_{i,k}^\ell} \left\{ (y - \alpha_{i,k}^\ell)^2 + \delta_{\{z_{i,k}^\ell \leq 0, -y\gamma(x_i, t_{k+1}) \geq z_{i,k}^\ell\}} (y\gamma(x_i, t_{k+1}) + z_{i,k}^\ell)^2 \right. \\ &\quad \left. + \delta_{\{z_{i-1,k}^\ell > 0, y\gamma(x_i, t_{k+1}) \leq z_{i-1,k}^\ell\}} (y\gamma(x_i, t_{k+1}) - z_{i-1,k}^\ell)^2 \right\}, \end{aligned}$$

where $z_{i,k}^\ell = m_{i,k}^\ell + \sigma (D_x^+ \bar{\phi}^{\ell+1})_{i,k+1}$,

$\alpha_{i,k}^\ell = \rho_{i,k}^\ell + \sigma ((D_t^- \bar{\phi}^{\ell+1})_{i,k+1} + f(x_i, t_{k+1}) - \epsilon (D_{xx} \bar{\phi}^{\ell+1})_{i,k+1})$, and δ_C is a function which takes the value 1 if the constraint in C is satisfied, and 0 otherwise. This optimization problem can be solved by comparing the function values of several candidates in $\mathcal{A}_{i,k}^\ell$ defined in (57).

For details, see Section B.2.1.

9 Update $m_{i,k}$ for all $i = 1, \dots, n_x$; $k = 1, \dots, n_t - 1$ by

$$m_{i,k}^{\ell+1} = \max \left\{ \min \left\{ z_{i,k}^\ell, \rho_{i+1,k}^{\ell+1} \gamma(x_{i+1}, t_{k+1}) \right\}, -\rho_{i,k}^{\ell+1} \gamma(x_i, t_{k+1}) \right\},$$

where $z_{i,k}^\ell$ is defined in the last step.

10 **end**

11 Return $(\phi_{i,k}^N)_{i,k}$.

For a fixed ρ , the minimization for m is separable and in the following form

$$m_{i,k}^{\ell+1} = \arg \min_{y \in \mathbb{R}} \{(y - z_{i,k}^\ell)^2 : y \in [-\rho_{i,k}\gamma(x_i, t_{k+1}), \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1})]\}, \quad (51)$$

whose solution is given by the truncation of $z_{i,k}^\ell$ in the interval $[-\rho_{i,k}\gamma(x_i, t_{k+1}), \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1})]$:

$$m_{i,k}^{\ell+1} = \max\{\min\{z_{i,k}^\ell, \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1})\}, -\rho_{i,k}\gamma(x_i, t_{k+1})\}. \quad (52)$$

Note that the right hand side of (52) can be seen as a function of $\rho_{i,k}$ and $\rho_{i+1,k}$. After some computation, we get

- If $z_{i,k}^\ell > 0$, we have $m_{i,k}^{\ell+1} = \min\{z_{i,k}^\ell, \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1})\}$, and hence $m_{i,k}^{\ell+1}$ only depends on $\rho_{i+1,k}$ in the following form (denoting this function by $G_{i,k}^+$)

$$m_{i,k}^{\ell+1} = G_{i,k}^+(\rho_{i+1,k}) := \begin{cases} \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1}), & \text{if } \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1}) \leq z_{i,k}^\ell, \\ z_{i,k}^\ell, & \text{if } \rho_{i+1,k}\gamma(x_{i+1}, t_{k+1}) > z_{i,k}^\ell. \end{cases} \quad (53)$$

- If $z_{i,k}^\ell \leq 0$, we have $m_{i,k}^{\ell+1} = \max\{z_{i,k}^\ell, -\rho_{i,k}\gamma(x_i, t_{k+1})\}$, and hence $m_{i,k}^{\ell+1}$ only depends on $\rho_{i,k}$ in the following form (denoting this function by $G_{i,k}^-$)

$$m_{i,k}^{\ell+1} = G_{i,k}^-(\rho_{i,k}) := \begin{cases} -\rho_{i,k}\gamma(x_i, t_{k+1}), & \text{if } -\rho_{i,k}\gamma(x_i, t_{k+1}) \geq z_{i,k}^\ell, \\ z_{i,k}^\ell, & \text{if } -\rho_{i,k}\gamma(x_i, t_{k+1}) < z_{i,k}^\ell. \end{cases} \quad (54)$$

In other words, depending on the sign of $z_{i,k}^\ell$, $m_{i,k}^{\ell+1}$ is a piecewise linear (ReLU-shape) function which depends either on $\rho_{i,k}$ or on $\rho_{i+1,k}$.

We plug these formulas into (50) to get a minimization problem of ρ only:

$$\begin{aligned} \rho_{i,k}^{\ell+1} &= \arg \min_{y \geq 0} (y - \alpha_{i,k}^\ell)^2 + \delta_{\{z_{i,k}^\ell \leq 0\}} (G_{i,k}^-(y) - z_{i,k}^\ell)^2 + \delta_{\{z_{i-1,k}^\ell > 0\}} (G_{i-1,k}^+(y) - z_{i-1,k}^\ell)^2 \\ &= \arg \min_{y \geq 0} (y - \alpha_{i,k}^\ell)^2 + \delta_{\{z_{i,k}^\ell \leq 0, -y\gamma(x_i, t_{k+1}) \geq z_{i,k}^\ell\}} \left(y\gamma(x_i, t_{k+1}) + z_{i,k}^\ell \right)^2 \\ &\quad + \delta_{\{z_{i-1,k}^\ell > 0, y\gamma(x_{i-1}, t_{k+1}) \leq z_{i-1,k}^\ell\}} \left(y\gamma(x_{i-1}, t_{k+1}) - z_{i-1,k}^\ell \right)^2. \end{aligned} \quad (55)$$

Consequently, the objective function takes on the form of a piecewise quadratic function. With each δ -function capable of taking only values of 1 or 0, there exist four potential combinations. We consider the minimization of each potential combination as a candidate. These candidates encompass $\alpha_{i,k}^\ell$, $\frac{\alpha_{i,k}^\ell - \gamma(x_i, t_{k+1})z_{i,k}^\ell}{\gamma(x_i, t_{k+1})^2 + 1}$, $\frac{\alpha_{i,k}^\ell + \gamma(x_{i-1}, t_{k+1})z_{i-1,k}^\ell}{\gamma(x_{i-1}, t_{k+1})^2 + 1}$, and $\frac{\alpha_{i,k}^\ell - \gamma(x_i, t_{k+1})z_{i,k}^\ell + \gamma(x_{i-1}, t_{k+1})z_{i-1,k}^\ell}{\gamma(x_i, t_{k+1})^2 + \gamma(x_{i-1}, t_{k+1})^2 + 1}$. We also incorporate $-\frac{z_{i,k}^\ell}{\gamma(x_i, t_{k+1})}$ and $\frac{z_{i-1,k}^\ell}{\gamma(x_{i-1}, t_{k+1})}$ into the candidates, which correspond to the points at which the δ -functions transition between values. Furthermore, considering the constraint $\rho \geq 0$, we truncate all these candidates to the interval $[0, +\infty)$ and additionally include 0 as another candidate. In essence, we update ρ employing the subsequent formula:

$$\begin{aligned} \rho_{i,k}^{\ell+1} &= \arg \min_{y \in \mathcal{A}_{i,k}^\ell} (y - \alpha_{i,k}^\ell)^2 + \delta_{\{z_{i,k}^\ell \leq 0, -y\gamma(x_i, t_{k+1}) \geq z_{i,k}^\ell\}} \left(y\gamma(x_i, t_{k+1}) + z_{i,k}^\ell \right)^2 \\ &\quad + \delta_{\{z_{i-1,k}^\ell > 0, y\gamma(x_{i-1}, t_{k+1}) \leq z_{i-1,k}^\ell\}} \left(y\gamma(x_{i-1}, t_{k+1}) - z_{i-1,k}^\ell \right)^2, \end{aligned} \quad (56)$$

where $\mathcal{A}_{i,k}^\ell$ designates the set encompassing all the candidates, defined by

$$\mathcal{A}_{i,k}^\ell = \left\{ 0, \left(\alpha_{i,k}^\ell \right)_+, \left(\frac{\alpha_{i,k}^\ell - \gamma(x_i, t_{k+1})z_{i,k}^\ell}{\gamma(x_i, t_{k+1})^2 + 1} \right)_+, \left(\frac{\alpha_{i,k}^\ell + \gamma(x_{i-1}, t_{k+1})z_{i-1,k}^\ell}{\gamma(x_{i-1}, t_{k+1})^2 + 1} \right)_+, \left(\frac{\alpha_{i,k}^\ell - \gamma(x_i, t_{k+1})z_{i,k}^\ell + \gamma(x_{i-1}, t_{k+1})z_{i-1,k}^\ell}{\gamma(x_i, t_{k+1})^2 + \gamma(x_{i-1}, t_{k+1})^2 + 1} \right)_+, \left(-\frac{z_{i,k}^\ell}{\gamma(x_i, t_{k+1})} \right)_+, \left(\frac{z_{i-1,k}^\ell}{\gamma(x_{i-1}, t_{k+1})} \right)_+ \right\}. \quad (57)$$

Here, y_+ denotes the positive part of the number y , denoted as $y_+ := \max\{y, 0\}$.

B.3 Two-dimensional semi-discrete and fully-discrete method

Within this section, we present the semi-discrete and fully-discrete counterparts of the two-dimensional version of (39). The derivations closely mirror those in the one-dimensional cases, and as such, we have omitted certain particulars. Initially, we proceed with spatial domain discretization, leading to the following formulation:

$$\begin{aligned} \min_{\substack{\phi_{i,j} \forall i,j \\ \phi_{i,j}(0)=g(x_i,y_j)}} \max_{\substack{\rho_{i,j}, m_{i,j}^1, m_{i,j}^2 \\ \rho_{i,j} \geq 0}} \int_0^T \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(\rho_{i,j}(t) \dot{\phi}_{i,j}(t) + m_{i,j}^1(t) (D_x^+ \phi)_{i,j}(t) + m_{i,j}^2(t) (D_y^+ \phi)_{i,j}(t) - \epsilon \rho_{i,j}(t) (D_{xx} \phi + D_{yy} \phi)_{i,j}(t) \right) \\ - \hat{L} \left((x_i)_{i,j}, t, (\rho_{i,j}(t))_{i,j}, (m_{i,j}^1(t))_{i,j}, (m_{i,j}^2(t))_{i,j} \right) dt - c \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j}(T). \end{aligned} \quad (58)$$

where $\hat{L}: \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times [0, +\infty) \times \mathbb{R}^{n_x \times n_y} \times \mathbb{R}^{n_x \times n_y} \times \mathbb{R}^{n_x \times n_y} \rightarrow \mathbb{R}$ is defined by

$$\hat{L} \left((x_i)_i, (y_j)_j, t, (\rho_{i,j})_{i,j}, (m_{i,j}^1)_{i,j}, (m_{i,j}^2)_{i,j} \right) = \max_{d_{i,j}^1, d_{i,j}^2 \forall i,j} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(m_{i,j}^1 d_{i,j}^1 + m_{i,j}^2 d_{i,j}^2 - \rho_{i,j} \hat{H}(x_i, y_j, t, d_{i,j}^1, d_{i-1,j}^1, d_{i,j}^2, d_{i,j-1}^2) \right).$$

The derivation of this formula is similar as the one-dimensional case, and hence omitted here.

Remark B.5. *In the case when $\hat{H}(x, y, t, p^{1,+}, p^{1,-}, p^{2,+}, p^{2,-})$ is “separable”, i.e., it can be written as $\hat{H}_1^1(x, y, t, p^{1,+}) + \hat{H}_2^1(x, y, t, p^{1,-}) + \hat{H}_1^2(x, y, t, p^{2,+}) + \hat{H}_2^2(x, y, t, p^{2,-})$, we have*

$$\begin{aligned} & \max_{d_{i,j}^1, d_{i,j}^2 \forall i,j} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(m_{i,j}^1 d_{i,j}^1 + m_{i,j}^2 d_{i,j}^2 - \rho_{i,j} \hat{H}(x_i, y_j, t, d_{i,j}^1, d_{i-1,j}^1, d_{i,j}^2, d_{i,j-1}^2) \right) \\ &= \max_{d_{i,j}^1, d_{i,j}^2 \forall i,j} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(m_{i,j}^1 d_{i,j}^1 + m_{i,j}^2 d_{i,j}^2 - \rho_{i,j} \hat{H}_1^1(x_i, y_j, t, d_{i,j}^1) - \rho_{i+1,j} \hat{H}_2^1(x_{i+1}, y_j, t, d_{i,j}^1) - \rho_{i,j} \hat{H}_1^2(x_i, y_j, t, d_{i,j}^2) - \rho_{i,j+1} \hat{H}_2^2(x_i, y_{j+1}, t, d_{i,j}^2) \right) \\ &= \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left((\rho_{i,j} \hat{H}_{1,i,j,t}^1 + \rho_{i+1,j} \hat{H}_{2,i+1,j,t}^1)^* (m_{i,j}^1) + (\rho_{i,j} \hat{H}_{1,i,j,t}^2 + \rho_{i,j+1} \hat{H}_{2,i,j+1,t}^2)^* (m_{i,j}^2) \right), \end{aligned} \quad (59)$$

where $\hat{H}_{a,i,j,t}^b$ denotes the functions $p \mapsto \hat{H}_a^b(x_i, y_j, t, p)$. The corresponding saddle point problem is

$$\begin{aligned} \min_{\substack{\phi_{i,j} \forall i,j \\ \phi_{i,j}(0)=g(x_i,y_j)}} \max_{\substack{\rho_{i,j}, m_{i,j}^1, m_{i,j}^2 \\ \rho_{i,j} \geq 0}} \int_0^T \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(\rho_{i,j}(t) \dot{\phi}_{i,j}(t) + m_{i,j}^1(t) (D_x^+ \phi)_{i,j}(t) + m_{i,j}^2(t) (D_y^+ \phi)_{i,j}(t) - \epsilon \rho_{i,j}(t) (D_{xx} \phi + D_{yy} \phi)_{i,j}(t) \right) \\ - \left((\rho_{i,j}(t) \hat{H}_{1,i,j,t}^1 + \rho_{i+1,j}(t) \hat{H}_{2,i+1,j,t}^1)^* (m_{i,j}^1(t)) + (\rho_{i,j}(t) \hat{H}_{1,i,j,t}^2 + \rho_{i,j+1}(t) \hat{H}_{2,i,j+1,t}^2)^* (m_{i,j}^2(t)) \right) dt - c \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j}(T). \end{aligned} \quad (60)$$

Now, we perform the implicit time discretization and use $(D_t^- \phi)_{i,j,k}$ to approximate $\dot{\phi}_{i,j}(t)$. Then, the saddle point formula (58) becomes (note that the index k for ρ , m^1 , m^2 are from 1 to $n_t - 1$, while the range of k for ϕ is from 1 to n_t)

$$\begin{aligned} \min_{\substack{\phi_{i,j,k} \forall i,j,k \\ \phi_{i,j,1}=g(x_i,y_j)}} \max_{\substack{\rho_{i,j,k}, m_{i,j,k}^1, m_{i,j,k}^2 \\ \rho_{i,j,k} \geq 0}} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t-1} \left(\rho_{i,j,k} (D_t^- \phi)_{i,j,k+1} + m_{i,j,k}^1 (D_x^+ \phi)_{i,j,k+1} + m_{i,j,k}^2 (D_y^+ \phi)_{i,j,k+1} \right) \\ - \epsilon \rho_{i,j,k} (D_{xx} \phi + D_{yy} \phi)_{i,j,k+1} - \hat{L} \left((x_i)_i, (y_j)_j, (t_k)_k, (\rho_{i,j,k})_{i,j,k}, (m_{i,j,k}^1)_{i,j,k}, (m_{i,j,k}^2)_{i,j,k} \right) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j,n_t}, \end{aligned} \quad (61)$$

where we abuse the notation \hat{L} and let $\hat{L}: \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_t} \times (\mathbb{R}^{n_x \times (n_t-1)})^3 \rightarrow \mathbb{R}$ be defined by

$$\begin{aligned} \hat{L} \left((x_i)_i, (y_j)_j, (t_k)_k, (\rho_{i,j,k})_{i,j,k}, (m_{i,j,k}^1)_{i,j,k}, (m_{i,j,k}^2)_{i,j,k} \right) = \max_{d_{i,j,k}^1, d_{i,j,k}^2 \forall i,j,k} \sum_{k=1}^{n_t-1} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \left(m_{i,j,k}^1 d_{i,j,k}^1 + m_{i,j,k}^2 d_{i,j,k}^2 \right) \\ - \rho_{i,j,k} \hat{H}(x_i, y_j, t_{k+1}, d_{i,j,k}^1, d_{i-1,j,k}^1, d_{i,j,k}^2, d_{i,j-1,k}^2). \end{aligned}$$

Similar to the one-dimensional case, this formula is particularly well-suited for Hamiltonians that possess specific structures, such as being separable and shifted 1-homogeneous with respect to p , as we will elaborate on in the subsequent section.

B.4 Two-dimensional special case: separable and shifted 1-homogeneous Hamiltonian

In this section, we consider the two-dimensional HJ PDE whose Hamiltonian is in the form of

$$\hat{H}(x, y, t, p) = \gamma(x, y, t) \|p\|_1 + f(x, y, t) = \gamma(x, y, t) (|p_1| + |p_2|) + f(x, y, t), \quad (62)$$

with the assumption that $\gamma(x, y, t) > 0$ for any $(x, y) \in \Omega$, $t \in [0, T]$. Although we consider this specific case, the derivation can be generalized to a Hamiltonian which is separable (see Remark B.5), convex, and 1-homogeneous with respect to p .

We adopt the Engquist-Osher scheme [79, 38, 39], where we set $\hat{H}(x, y, t, p^{1,+}, p^{1,-}, p^{2,+}, p^{2,-})$ to be $\hat{H}_-(x, y, t, p^{1,+}) + \hat{H}_+(x, y, t, p^{1,-}) + \hat{H}_-(x, y, t, p^{2,+}) + \hat{H}_+(x, y, t, p^{2,-}) + f(x, y, t)$, with $\hat{H}_-(x, y, t, p) = \gamma(x, y, t) \max\{-p, 0\}$ and $\hat{H}_+(x, y, t, p) = \gamma(x, y, t) \max\{p, 0\}$. In this context, the computation takes the following form (where $\rho_{i,j}$, $m_{i,j}^1$, $m_{i,j}^2$, $d_{i,j}^1$, $d_{i,j}^2$ are used instead of $\rho_{i,j,k}$, $m_{i,j,k}^1$, $m_{i,j,k}^2$, $d_{i,j,k}^1$, $d_{i,j,k}^2$ for simplified notation):

$$\begin{aligned} & \max_{d^1, d^2 \in \mathbb{R}^{n_x \times n_y}} \sum_{i,j} m_{i,j}^1 d_{i,j}^1 + m_{i,j}^2 d_{i,j}^2 - \rho_{i,j} \hat{H}(x_i, y_j, t_{k+1}, d_{i,j}^1, d_{i-1,j}^1, d_{i,j}^2, d_{i,j-1}^2) \\ &= \max_{d^1, d^2 \in \mathbb{R}^{n_x \times n_y}} \sum_{i,j} \left(m_{i,j}^1 d_{i,j}^1 + m_{i,j}^2 d_{i,j}^2 - \rho_{i,j} \hat{H}_-(x_i, y_j, t_{k+1}, d_{i,j}^1) - \rho_{i+1,j} \hat{H}_+(x_{i+1}, y_j, t_{k+1}, d_{i,j}^1) - \rho_{i,j} \hat{H}_-(x_i, y_j, t_{k+1}, d_{i,j}^2) \right. \\ & \quad \left. - \rho_{i,j+1} \hat{H}_+(x_i, y_{j+1}, t_{k+1}, d_{i,j}^2) - f(x_i, y_j, t_{k+1}) \right) \\ &= - \sum_{i,j} f(x_i, y_j, t_{k+1}) + I_{[-\rho_{i,j} \gamma(x_i, y_j, t_{k+1}), \rho_{i+1,j} \gamma(x_{i+1}, y_j, t_{k+1})]}(m_{i,j}^1) + I_{[-\rho_{i,j} \gamma(x_i, y_j, t_{k+1}), \rho_{i,j+1} \gamma(x_i, y_{j+1}, t_{k+1})]}(m_{i,j}^2). \end{aligned} \quad (63)$$

Then, the saddle point problem (61) becomes

$$\begin{aligned} & \min_{\phi} \max_{\substack{\rho, m^1, m^2 \\ \rho_{i,j,k} \geq 0}} \left\{ \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t-1} \left(\rho_{i,j,k} (D_t^- \phi)_{i,j,k+1} + m_{i,j,k}^1 (D_x^+ \phi)_{i,j,k+1} + m_{i,j,k}^2 (D_y^+ \phi)_{i,j,k+1} - \epsilon \rho_{i,j,k} (D_{xx} \phi + D_{yy} \phi)_{i,j,k+1} \right. \right. \\ & \quad \left. \left. - \rho_{i,j,k} f(x_i, y_j, t_k) \right) - \frac{c}{\Delta t} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \phi_{i,j,n_t} : -\rho_{i,j,k} \gamma(x_i, y_j, t_{k+1}) \leq m_{i,j,k}^1 \leq \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1}), \right. \\ & \quad \left. -\rho_{i,j,k} \gamma(x_i, y_j, t_{k+1}) \leq m_{i,j,k}^2 \leq \rho_{i,j+1,k} \gamma(x_i, y_{j+1}, t_{k+1}), \forall i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1 \right\}. \end{aligned} \quad (64)$$

The update procedure for ϕ remains as previously described. We will now detail the process of deriving updates for ρ and m .

B.4.1 The detailed derivation of updating ρ and m in Algorithm 7

Define z and α as in Algorithm 7. With a similar computation as in Section B.4.1, we get explicit formulas for $m_{1,i,j}^{\ell+1}$ and $m_{2,i,j}^{\ell+1}$ using ρ :

$$\begin{aligned} m_{i,j,k}^{1,\ell+1} &= \max \left\{ \min \left\{ z_{i,j,k}^{1,\ell}, \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1}) \right\}, -\rho_{i,j,k} \gamma(x_i, y_j, t_{k+1}) \right\}, \\ m_{i,j,k}^{2,\ell+1} &= \max \left\{ \min \left\{ z_{i,j,k}^{2,\ell}, \rho_{i,j+1,k} \gamma(x_i, y_{j+1}, t_{k+1}) \right\}, -\rho_{i,j,k} \gamma(x_i, y_j, t_{k+1}) \right\}. \end{aligned} \quad (66)$$

In other words, we get

- If $z_{i,j,k}^{1,\ell} > 0$, $m_{i,j,k}^{1,\ell+1} = \min\{z_{i,j,k}^{1,\ell}, \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1})\}$, and hence $m_{i,j,k}^{1,\ell+1}$ only depends on $\rho_{i+1,j,k}$ in the following form (denoting this function by $G_{i,j,k}^{1,+}$)

$$m_{i,j,k}^{1,\ell+1} = G_{i,j,k}^{1,+}(\rho_{i+1,j,k}) := \begin{cases} \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1}), & \text{if } \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1}) \leq z_{i,j,k}^{1,\ell}, \\ z_{i,j,k}^{1,\ell}, & \text{if } \rho_{i+1,j,k} \gamma(x_{i+1}, y_j, t_{k+1}) > z_{i,j,k}^{1,\ell}. \end{cases} \quad (67)$$

Algorithm 7: The proposed algorithm for solving (64)

Inputs : Stepsize $\tau, \sigma > 0$, error tolerance $\delta > 0$, and the maximal iteration number N .

Outputs: Solution to the saddle point problem (64).

1 For each $i = 1, \dots, n_x; j = 1, \dots, n_y$, initialize the matrices by $\phi_{i,j,k}^0 = g(x_i, y_j)$ for $k = 1, \dots, n_t$,
 $\rho_{i,j,k}^0 = c, m_{i,j,k}^{1,0} = m_{i,j,k}^{2,0} = 0$ for $k = 1, \dots, n_t - 1$.

2 **for** $\ell = 0, 1, \dots, N - 1$ **do**

3 Update the matrix $\phi_{i,k}$ for $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t$ by

$$\phi^{\ell+1} = \phi^\ell + \tau(-D_{tt} - D_{xx} - D_{yy})^{-1} \left(D_t^+ \rho^\ell + D_x^- m^{1,\ell} + D_y^- m^{2,\ell} + \epsilon D_{xx} \rho^\ell + \epsilon D_{yy} \rho^\ell \right), \quad (65)$$

where $(-D_{tt} - D_{xx} - D_{yy})^{-1} f$ (for a tensor f with elements $f_{i,j,k}$ and the linear operator $D_{tt} f = (\frac{f_{i,j,k-1} - 2f_{i,j,k} + f_{i,j,k+1}}{\Delta t^2})_{i,j,k}$) denotes the solution u to the linear system $-(D_{tt} u)_{i,j,k+1} - (D_{xx} u + D_{yy} u)_{i,j,k+1} = f_{i,j,k}$ for all $i = 1, \dots, n_x; j = 1, \dots, n_y; k = 1, \dots, n_t - 1$ with periodic spatial condition, Dirichlet initial condition $u_{i,j,1} = 0$, and Neumann terminal condition $u_{i,j,n_t+1} = u_{i,j,n_t}$.

4 **if** $\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \sum_{k=2}^{n_t} |(D_t^- \phi^{\ell+1})_{i,j,k} + \hat{H}(x_i, y_j, t_k, (D_x^+ \phi^{\ell+1})_{i,j,k}, (D_x^- \phi^{\ell+1})_{i,j,k}, (D_y^+ \phi^{\ell+1})_{i,j,k}, (D_y^- \phi^{\ell+1})_{i,j,k}) - \epsilon(D_{xx} \phi^{\ell+1} + D_{yy} \phi^{\ell+1})_{i,j,k}| \leq \delta$ **then**
 5 | Return $\phi^{\ell+1}$.

6 **end**

7 Set $\bar{\phi}^{\ell+1} = 2\phi^{\ell+1} - \phi^\ell$.

8 Update ρ by

$$\begin{aligned} \rho_{i,k}^{\ell+1} = \arg \min_{y \in \mathcal{A}_{i,j,k}^\ell} & (y - \alpha_{i,j,k}^\ell)^2 + \delta_{\{z_{i,j,k}^{1,\ell} \leq 0, -y\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{1,\ell}\}} (y\gamma(x_i, y_j, t_{k+1}) + z_{i,j,k}^{1,\ell})^2 \\ & + \delta_{\{z_{i-1,j,k}^{1,\ell} > 0, y\gamma(x_{i-1}, y_j, t_{k+1}) \leq z_{i-1,j,k}^{1,\ell}\}} (y\gamma(x_{i-1}, y_j, t_{k+1}) - z_{i-1,j,k}^{1,\ell})^2 \\ & + \delta_{\{z_{i,j,k}^{2,\ell} \leq 0, -y\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{2,\ell}\}} (y\gamma(x_i, y_j, t_{k+1}) + z_{i,j,k}^{2,\ell})^2 \\ & + \delta_{\{z_{i,j-1,k}^{2,\ell} > 0, y\gamma(x_i, y_{j-1}, t_{k+1}) \leq z_{i,j-1,k}^{2,\ell}\}} (y\gamma(x_i, y_{j-1}, t_{k+1}) - z_{i,j-1,k}^{2,\ell})^2, \end{aligned}$$

where $z_{i,j,k}^{1,\ell} = m_{i,j,k}^{1,\ell} + \sigma(D_x^+ \bar{\phi}^{\ell+1})_{i,j,k+1}$, $z_{i,j,k}^{2,\ell} = m_{i,j,k}^{2,\ell} + \sigma(D_y^+ \bar{\phi}^{\ell+1})_{i,j,k+1}$, $\alpha_{i,j,k}^\ell = \rho_{i,j,k}^\ell + \sigma((D_t^- \bar{\phi}^{\ell+1})_{i,j,k+1} + f(x_i, y_j, t_{k+1}) - \epsilon(D_{xx} \bar{\phi}^{\ell+1} + D_{yy} \bar{\phi}^{\ell+1})_{i,j,k+1})$, and δ_C denotes a function which takes the value 1 if the constraint in C is satisfied, and 0 otherwise.

This optimization problem can be solved by comparing the function values of several candidates in $\mathcal{A}_{i,j,k}^\ell$ defined in (72). For details, see Section B.4.1.

9 Update $m_{i,j,k}^1$ and $m_{i,j,k}^2$ for all i, j, k ($k < n_t$) by

$$\begin{aligned} m_{i,j,k}^{1,\ell+1} &= \max \left\{ \min \left\{ z_{i,j,k}^{1,\ell}, \rho_{i+1,j,k}^{\ell+1} \gamma(x_{i+1}, y_j, t_{k+1}) \right\}, -\rho_{i,j,k}^{\ell+1} \gamma(x_i, y_j, t_{k+1}) \right\}, \\ m_{i,j,k}^{2,\ell+1} &= \max \left\{ \min \left\{ z_{i,j,k}^{2,\ell}, \rho_{i,j+1,k}^{\ell+1} \gamma(x_i, y_{j+1}, t_{k+1}) \right\}, -\rho_{i,j,k}^{\ell+1} \gamma(x_i, y_j, t_{k+1}) \right\}. \end{aligned}$$

where $z_{i,j,k}^{1,\ell}$ and $z_{i,j,k}^{2,\ell}$ are defined in the last step.

10 **end**

11 Return ϕ^N .

- If $z_{i,j,k}^{1,\ell} \leq 0$, we have $m_{i,j,k}^{1,\ell+1} = \max\{z_{i,j,k}^{1,\ell}, -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1})\}$, and hence $m_{i,j,k}^{1,\ell+1}$ only depends on $\rho_{i,j,k}$ in the following form (denoting this function by $G_{i,j,k}^{1,-}$)

$$m_{i,j,k}^{1,\ell+1} = G_{i,j,k}^{1,-}(\rho_{i,j,k}) := \begin{cases} -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}), & \text{if } -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{1,\ell}, \\ z_{i,j,k}^{1,\ell}, & \text{if } -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}) < z_{i,j,k}^{1,\ell}. \end{cases} \quad (68)$$

- If $z_{i,j,k}^{2,\ell} > 0$, we have $m_{i,j,k}^{2,\ell+1} = \min\{z_{i,j,k}^{2,\ell}, \rho_{i,j+1,k}\gamma(x_i, y_{j+1}, t_{k+1})\}$, and hence $m_{i,j,k}^{2,\ell+1}$ only depends on $\rho_{i,j+1,k}$ in the following form (denoting this function by $G_{i,j,k}^{2,+}$)

$$m_{i,j,k}^{2,\ell+1} = G_{i,j,k}^{2,+}(\rho_{i,j+1,k}) := \begin{cases} \rho_{i,j+1,k}\gamma(x_i, y_{j+1}, t_{k+1}), & \text{if } \rho_{i,j+1,k}\gamma(x_i, y_{j+1}, t_{k+1}) \leq z_{i,j,k}^{2,\ell}, \\ z_{i,j,k}^{2,\ell}, & \text{if } \rho_{i,j+1,k}\gamma(x_i, y_{j+1}, t_{k+1}) > z_{i,j,k}^{2,\ell}. \end{cases} \quad (69)$$

- If $z_{i,j,k}^{2,\ell} \leq 0$, we have $m_{i,j,k}^{2,\ell+1} = \max\{z_{i,j,k}^{2,\ell}, -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1})\}$, and hence $m_{i,j,k}^{2,\ell+1}$ only depends on $\rho_{i,j,k}$ in the following form (denoting this function by $G_{i,j,k}^{2,-}$)

$$m_{i,j,k}^{2,\ell+1} = G_{i,j,k}^{2,-}(\rho_{i,j,k}) := \begin{cases} -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}), & \text{if } -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{2,\ell}, \\ z_{i,j,k}^{2,\ell}, & \text{if } -\rho_{i,j,k}\gamma(x_i, y_j, t_{k+1}) < z_{i,j,k}^{2,\ell}. \end{cases} \quad (70)$$

Therefore, depending on the sign of $z_{i,j,k}^{1,\ell}$, the minimizer $m_{i,j,k}^{1,\ell+1}$ is a piecewise linear function depending on either $\rho_{i,j,k}$ or $\rho_{i+1,j,k}$. Similarly, depending on the sign of $z_{i,j,k}^{2,\ell}$, the minimizer $m_{i,j,k}^{2,\ell+1}$ is a piecewise linear function depending on either $\rho_{i,j,k}$ or $\rho_{i,j+1,k}$.

Then, we plug the minimizers m^1 and m^2 into the original updating scheme to get a minimization problem of ρ only:

$$\begin{aligned} \rho_{i,j,k}^{\ell+1} &= \arg \min_{y \geq 0} (y - \alpha_{i,j,k}^\ell)^2 + \delta_{\{z_{i,j,k}^{1,\ell} \leq 0\}} (G_{i,j,k}^{1,-}(y) - z_{i,j,k}^{1,\ell})^2 + \delta_{\{z_{i-1,j,k}^{1,\ell} > 0\}} (G_{i-1,j,k}^{1,+}(y) - z_{i-1,j,k}^{1,\ell})^2 \\ &\quad + \delta_{\{z_{i,j,k}^{2,\ell} \leq 0\}} (G_{i,j,k}^{2,-}(y) - z_{i,j,k}^{2,\ell})^2 + \delta_{\{z_{i,j-1,k}^{2,\ell} > 0\}} (G_{i,j-1,k}^{2,+}(y) - z_{i,j-1,k}^{2,\ell})^2 \\ &= \arg \min_{y \geq 0} (y - \alpha_{i,j,k}^\ell)^2 + \delta_{\{z_{i,j,k}^{1,\ell} \leq 0, -y\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{1,\ell}\}} (y\gamma(x_i, y_j, t_{k+1}) + z_{i,j,k}^{1,\ell})^2 \\ &\quad + \delta_{\{z_{i-1,j,k}^{1,\ell} > 0, y\gamma(x_{i-1}, y_j, t_{k+1}) \leq z_{i-1,j,k}^{1,\ell}\}} (y\gamma(x_{i-1}, y_j, t_{k+1}) - z_{i-1,j,k}^{1,\ell})^2 \\ &\quad + \delta_{\{z_{i,j,k}^{2,\ell} \leq 0, -y\gamma(x_i, y_j, t_{k+1}) \geq z_{i,j,k}^{2,\ell}\}} (y\gamma(x_i, y_j, t_{k+1}) + z_{i,j,k}^{2,\ell})^2 \\ &\quad + \delta_{\{z_{i,j-1,k}^{2,\ell} > 0, y\gamma(x_i, y_{j-1}, t_{k+1}) \leq z_{i,j-1,k}^{2,\ell}\}} (y\gamma(x_i, y_{j-1}, t_{k+1}) - z_{i,j-1,k}^{2,\ell})^2. \end{aligned} \quad (71)$$

Given that each δ -function can adopt values of either 0 or 1, there exist a total of 16 distinct combinations, each corresponding to a unique minimizer. We integrate the positive parts of these minimizers into our pool of candidates, applying truncation due to the non-negativity of ρ . These 16 candidates comprise the following ensemble:

$$\mathcal{A}_1 = \left\{ \left(\frac{\alpha_{i,j,k}^\ell - \sum_{(v_0, v_1) \in \mathcal{C}} v_0 v_1}{1 + \sum_{(v_0, v_1) \in \mathcal{C}} v_0^2} \right)_+ : \mathcal{C} \subseteq \left\{ (\gamma(x_i, y_j, t_{k+1}), z_{i,j,k}^{1,\ell}), (\gamma(x_{i-1}, y_j, t_{k+1}), -z_{i-1,j,k}^{1,\ell}), \right. \right. \\ \left. \left. (\gamma(x_i, y_j, t_{k+1}), z_{i,j,k}^{2,\ell}), (\gamma(x_i, y_{j-1}, t_{k+1}), -z_{i,j-1,k}^{2,\ell}) \right\} \right\}.$$

We also include the boundary points as in Section B.2.1 to get the following set of all candidates

$$\mathcal{A}_{i,j,k}^\ell = \left\{ 0, \left(-\frac{z_{i,j,k}^{1,\ell}}{\gamma(x_i, y_j, t_{k+1})} \right)_+, \left(\frac{z_{i-1,j,k}^{1,\ell}}{\gamma(x_{i-1}, y_j, t_{k+1})} \right)_+, \left(-\frac{z_{i,j,k}^{2,\ell}}{\gamma(x_i, y_j, t_{k+1})} \right)_+, \left(\frac{z_{i,j-1,k}^{2,\ell}}{\gamma(x_i, y_{j-1}, t_{k+1})} \right)_+ \right\} \cup \mathcal{A}_1. \quad (72)$$

Then, the minimizer $\rho_{i,j,k}^{\ell+1}$ is chosen from these candidates by comparing their objective function values.