



Nonstandard Local Discontinuous Galerkin Methods for Fully Nonlinear Second Order Elliptic and Parabolic Equations in High Dimensions

Xiaobing Feng¹ · Thomas Lewis²

Received: 28 December 2017 / Revised: 24 May 2018 / Accepted: 11 June 2018 / Published online: 26 June 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

This paper is concerned with developing accurate and efficient numerical methods for fully nonlinear second order elliptic and parabolic partial differential equations (PDEs) in multiple spatial dimensions. It presents a general framework for constructing high order local discontinuous Galerkin (LDG) methods for approximating viscosity solutions of these fully nonlinear PDEs. The proposed LDG methods are natural extensions of a narrow-stencil finite difference framework recently proposed by the authors for approximating viscosity solutions. The idea of the methodology is to use multiple approximations of first and second order derivatives as a way to resolve the potential low regularity of the underlying viscosity solution. Consistency and generalized monotonicity properties are proposed that ensure the numerical operator approximates the differential operator. The resulting algebraic system has several linear equations coupled with only one nonlinear equation that is monotone in many of its arguments. The structure can be explored to design nonlinear solvers. This paper also presents and analyzes numerical results for several numerical test problems in two dimensions which are used to gauge the accuracy and efficiency of the proposed LDG methods.

Keywords Fully nonlinear PDEs · Viscosity solutions · Discontinuous Galerkin methods

Mathematics Subject Classification 65N30 · 65M60 · 35J60 · 35K55

This paper is dedicated to Professor Bernardo Cockburn on the occasion of his sixtieth birthday.

The work of both authors was partially supported by the NSF Grant DMS-1620168..

✉ Xiaobing Feng
xfeng@math.utk.edu

Thomas Lewis
tllewis3@uncg.edu

¹ Department of Mathematics, The University of Tennessee, Knoxville, TN 37996, USA

² Department of Mathematics and Statistics, The University of North Carolina at Greensboro, Greensboro, NC 27410, USA

1 Introduction

In this paper we consider the following general fully nonlinear second order elliptic and parabolic PDEs in high dimensions:

$$F[u] := F(D^2u, \nabla u, u, x) = 0, \quad x \in \Omega \quad (1)$$

and

$$u_t + F(D^2u, \nabla u, u, x, t) = 0, \quad (x, t) \in \Omega_T := \Omega \times (0, T] \quad (2)$$

which are complemented by appropriate boundary and initial conditions for $\Omega \subset \mathbf{R}^d$ ($d = 2, 3$) a given bounded (possibly convex) domain. In particular, we are concerned with directly approximating $C^0(\overline{\Omega})$ (or bounded) solutions of fully nonlinear problems that correspond to the two prototypical fully nonlinear operators

$$F[u] = \det(D^2u) \quad \text{and} \quad F[u] = \inf_{\theta \in \Theta} (L_\theta u - f_\theta),$$

where L_θ is a second order linear elliptic operator with

$$L_\theta u := A^\theta : D^2u + b^\theta \cdot \nabla u + c^\theta u$$

for $A : B$ the Frobenius inner product for matrices $A, B \in \mathbf{R}^{d \times d}$. The first nonlinear operator defines the Monge–Ampère equation, [25], and the second nonlinear operator defines the Hamilton–Jacobi–Bellman equation, [15,16]. It should be noted that some parabolic counterparts of elliptic Monge–Ampère type equations may not have the form of (2) (cf. [21]). Fully nonlinear second order PDEs arise from many scientific and engineering fields [5]; they are a class of PDEs which are very difficult to analyze and even more challenging to approximate numerically.

Due to their fully nonlinear structures, fully nonlinear PDEs do not have variational (or weak) formulations in general. The weak solutions are often defined as *viscosity solutions* (see Sect. 2 for the definition). The non-variational structure prevents the applicability of standard Galerkin type methods such as finite element methods. On the other hand, to approximate very low regularity solutions of these PDEs, it is natural to use totally discontinuous piecewise polynomial functions (i.e., DG functions) due to their flexibility and the larger approximation spaces. As expected, such a method must be nonstandard (again) due to the fully nonlinear structure of these PDEs. Indeed, a class of nonstandard mixed interior penalty discontinuous Galerkin methods was developed by the authors in [11] that works well in both 1-D and high dimensions provided that the viscosity solutions belong to $C^0(\overline{\Omega}) \cap H^1(\Omega)$ and the polynomial degree is greater than or equal to 1. Their extensions to local discontinuous Galerkin (LDG) methods were done only in the 1-D case so far. There were several non-trivial barriers preventing the extensions in the high dimensional case.

The goal of this paper is to generalize the one-dimensional LDG framework and methods of [10] to approximate the PDEs (1) and (2) in high dimensions (i.e., $d \geq 2$). Specifically, we shall design and implement a class of local discontinuous Galerkin (LDG) methods which are based on a nonstandard mixed formulation of (1) and (2). Our interest in an LDG approach over the interior-penalty (IP) approach found in [11] is threefold. The first reason is due to the known increased potential for approximating gradients of regular solutions when compared with IPDG methods. The second motivation is due to the fact that the LDG approach will allow us to form two numerical gradients when discretizing fully nonlinear operators that formally involve the gradient of the viscosity solution. As already mentioned above, the formulation for the IPDG methods in [11] assumed the viscosity solutions were

in the space $C^0(\overline{\Omega}) \cap H^1(\Omega)$. By forming two numerical gradients, the LDG methods can naturally be formulated for viscosity solutions in the space $C^0(\overline{\Omega}) \setminus H^1(\Omega)$. Third, as will be seen in the following, the numerical derivatives associated with the LDG approach naturally generalize the corresponding difference quotients associated with a finite difference (FD) approach. Thus, we can potentially gain further insight into various FD methods for fully nonlinear problems by studying their LDG counterparts while also having a stronger theoretical foundation for the LDG methods proposed in this paper.

The main difficulty addressed in this paper is how to extend the one-dimensional framework of [10] to the high-dimensional setting. First, we will need to design a consistent way for forming multiple discrete gradient and Hessian approximations. To this end, we will utilize the conventions introduced in [12] where a finite element DG numerical calculus was developed based upon a discontinuous Galerkin methodology and choosing various fluxes to characterize various numerical derivative operators. To extend ideas to the high-dimensional setting we will discretize partial derivatives directly as a way to define various gradient approximations. We will need to introduce nonstandard trace operators that are consistent with the idea that each partial derivative is treated independently. Second, we will extend the framework to second order problems where the fully nonlinear differential operator also involves the gradient operator, as represented by the general problems (1) and (2). Third, on noting that the LDG formulation will introduce a large set of auxiliary equations, we will explore various solver techniques and the potential for variable reduction to reduce the computational cost.

We note that typically a DG formulation for a fully nonlinear problem is based upon a semi-Lagrangian approach or strong structure assumptions that guarantee a monotonicity property of the scheme (see [4,6,18,22,24,26,28] and the review article [5]). As such, the methods are limited to piecewise linear basis functions. Inspired by the work of Yan and Osher in [29], we seek to formulate DG methods that allow the use of high order polynomials and can achieve high-order accuracy. The methods proposed in this paper extend the narrow-stencil FD approach in [7,8,19] to high-order and to unstructured triangular meshes. As with the LDG methods for Hamilton–Jacobi equations in [29], the only analytic convergence result for the proposed LDG methods corresponds to choosing piecewise constant basis functions. In this special case, the proposed LDG methods reduce to the FD methods of [8]. Moreover, we are able to establish a link between the proposed LDG methods and the vanishing moment method of Feng and Neilan [14]. Heuristically such a link also helps to justify the proposed LDG methods for fully nonlinear second order PDEs in the same way the link to the vanishing viscosity method motivates the LDG methods of [29] for Hamilton–Jacobi equations.

The remainder of this paper is organized as follows. In Sect. 2 we introduce some background for the viscosity solution notion. In Sect. 3 we define key concepts of consistency and generalized monotonicity for numerical operators that will serve as the foundation of the proposed LDG framework. We also introduce the numerical operators that will be used in the design of our methods. The proposed LDG formulation for the nonlinear elliptic equation (1) is presented in Sect. 4. We use two main ideas in the formulation: the numerical viscosity borrowed from the discretization of first-order Hamilton–Jacobi equations and a novel concept of numerical moments. We also discuss various techniques for solving the resulting nonlinear (large) algebraic systems. In Sect. 5 we consider both explicit and implicit in time fully discrete LDG methods for the fully nonlinear parabolic equation (2) based on the method of lines approach. In Sect. 6 we present many numerical experiments for the proposed LDG methods. These numerical experiments verify the accuracy and demonstrate the efficiency of the new methods. The experiments also explore the role of the numerical moment in the formulation. Lastly, in Sect. 7, we provide some concluding remarks.

2 Preliminaries

We first recall the viscosity solution concept for fully nonlinear second order problems. For a bounded open domain $\Omega \subset \mathbf{R}^d$, let $B(\Omega)$, $USC(\Omega)$, and $LSC(\Omega)$ denote, respectively, the spaces of bounded, upper semi-continuous, and lower semi-continuous functions on Ω . For any $v \in B(\Omega)$, we define

$$v^*(x) := \limsup_{y \rightarrow x} v(y) \quad \text{and} \quad v_*(x) := \liminf_{y \rightarrow x} v(y).$$

Then, $v^* \in USC(\Omega)$ and $v_* \in LSC(\Omega)$, and they are called *the upper and lower semicontinuous envelopes* of v , respectively.

Given a function $F : \mathcal{S}^{d \times d} \times \mathbf{R}^d \times \mathbf{R} \times \overline{\Omega} \rightarrow \mathbf{R}$, where $\mathcal{S}^{d \times d}$ denotes the set of $d \times d$ symmetric real matrices, the general second order fully nonlinear PDE takes the form

$$F(D^2u, \nabla u, u, x) = 0 \quad \text{in } \overline{\Omega}. \tag{3}$$

Note that here we have used the convention of writing the boundary condition as a discontinuity of the PDE (cf. [1, p. 274]).

The following two definitions can be found in [1,2,17].

Definition 1 Equation (3) is said to be *elliptic* if for all $(\mathbf{q}, \lambda, x) \in \mathbf{R}^d \times \mathbf{R} \times \overline{\Omega}$ there holds

$$F(A, \mathbf{q}, \lambda, x) \leq F(B, \mathbf{q}, \lambda, x) \quad \forall A, B \in \mathcal{S}^{d \times d}, A \geq B, \tag{4}$$

where $A \geq B$ means that $A - B$ is a nonnegative definite matrix.

Equation (3) is said to be *proper elliptic* if for all $(\mathbf{q}, x) \in \mathbf{R}^d \times \overline{\Omega}$ there holds

$$F(A, \mathbf{q}, a, x) \leq F(B, \mathbf{q}, b, x) \quad \forall A, B \in \mathcal{S}^{d \times d}, A \geq B, a, b \in \mathbf{R}, a \leq b. \tag{5}$$

We note that when F is differentiable, ellipticity can also be defined by requiring that the matrix $\frac{\partial F}{\partial A}$ is negative semi-definite (cf. [17, p. 441]).

Definition 2 A function $u \in B(\Omega)$ is called a viscosity subsolution (resp. supersolution) of (3) if, for all $\varphi \in C^2(\overline{\Omega})$, if $u^* - \varphi$ (resp. $u_* - \varphi$) has a local maximum (resp. minimum) at $x_0 \in \overline{\Omega}$, then we have

$$F_*(D^2\varphi(x_0), \nabla\varphi(x_0), u^*(x_0), x_0) \leq 0$$

(resp. $F^*(D^2\varphi(x_0), \nabla\varphi(x_0), u_*(x_0), x_0) \geq 0$). The function u is said to be a viscosity solution of (3) if it is simultaneously a viscosity subsolution and a viscosity supersolution of (3).

Remark 1 It can be proved that it is sufficient only to consider $\varphi \in \mathcal{P}_2$, the space of all *quadratic polynomials*, in Definition 2 (see [2, page 20]).

Definition 3 Problem (3) is said to satisfy a *comparison principle* if the following statement holds. For any upper semi-continuous function u and lower semi-continuous function v on $\overline{\Omega}$, if u is a viscosity subsolution and v is a viscosity supersolution of (3), then $u \leq v$ on $\overline{\Omega}$.

We remark that if F and u are continuous, then the upper and lower $*$ indices can be removed in Definition 2. The definition of ellipticity implies that the differential operator F must be non-increasing in its first argument in order to be elliptic. It turns out that ellipticity and a comparison principle provide sufficient conditions for Eq. (3) to fulfill a maximum principle (cf. [2,17]). It is clear from the above definition that viscosity solutions in general

do not satisfy the underlying PDEs in a tangible sense, and the concept of viscosity solutions is *nonvariational*. Such a solution is not defined through integration by parts against arbitrary test functions; hence, it does not satisfy an integral identity. The non-variational nature of viscosity solutions is the main obstacle that prevents the direct construction of Galerkin-type methods.

3 A Generalized Monotone Nonstandard LDG Framework

Our methodology for directly approximating viscosity solutions of second-order fully nonlinear PDEs is based on several motivational ideas which we explain below. Since integration by parts cannot be performed on Eq. (1), *the first key idea* is to introduce the auxiliary variables $P := D^2u$ and $q := \nabla u$ and rewrite the original fully nonlinear PDE as a system of PDEs:

$$F(p, q, u, x) = 0, \tag{6a}$$

$$q - \nabla u = 0, \tag{6b}$$

$$P - \nabla q = 0. \tag{6c}$$

To address the fact that ∇u and D^2u may not exist for a viscosity solution $u \in C^0(\overline{\Omega})$, *the second key idea* is to formally replace $q := \nabla u$ by two possible values of ∇u , namely, the left and right (possibly infinite) limits, and $P := \nabla q$ by two possible values for each possible q , namely, the left and right (possibly infinite) limits. Thus, we have the auxiliary variables $q^-, q^+ : \Omega \rightarrow \mathbf{R}^d$ and $P^{--}, P^{-+}, P^{+-}, P^{++} : \Omega \rightarrow \mathbf{R}^{d \times d}$ such that

$$[q^-(x)]_i = \lim_{\sigma \rightarrow 0^+} [\nabla u(x - \sigma \mathbf{e}_i)]_i, \tag{7a}$$

$$[q^+(x)]_i = \lim_{\sigma \rightarrow 0^+} [\nabla u(x + \sigma \mathbf{e}_i)]_i, \tag{7b}$$

$$[P^{--}(x)]_{ij} = \lim_{\sigma \rightarrow 0^+} [\nabla q^-(x - \sigma \mathbf{e}_j)]_{ij}, \tag{7c}$$

$$[P^{-+}(x)]_{ij} = \lim_{\sigma \rightarrow 0^+} [\nabla q^-(x + \sigma \mathbf{e}_j)]_{ij}, \tag{7d}$$

$$[P^{+-}(x)]_{ij} = \lim_{\sigma \rightarrow 0^+} [\nabla q^+(x - \sigma \mathbf{e}_j)]_{ij}, \tag{7e}$$

$$[P^{++}(x)]_{ij} = \lim_{\sigma \rightarrow 0^+} [\nabla q^+(x + \sigma \mathbf{e}_j)]_{ij} \tag{7f}$$

for all $i, j \in \{1, 2, \dots, d\}$, where \mathbf{e}_i denotes the i th canonical basis vector for \mathbf{R}^d . *The third key idea* is to replace (6a) by

$$\widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, u, x) = 0, \tag{8}$$

where \widehat{F} , which is called a *numerical operator*, should be some well-chosen approximation to F that incorporates the multiple gradient and Hessian variables.

The next step is to address the key issue about what criterion or properties “good” numerical operators \widehat{F} should satisfy. A large part of our framework revolves around describing sufficient conditions on the choice of numerical operators, as reflected in the following definitions that generalize the one-dimensional definitions given in [10].

Definition 4 (i) A function $\widehat{F} : (\mathbf{R}^{d \times d})^4 \times (\mathbf{R}^d)^2 \times \mathbf{R} \times \Omega \rightarrow \mathbf{R}$ is called a *numerical operator*.

(ii) Let $P \in \mathbf{R}^{d \times d}$, $q \in \mathbf{R}^d$, $v \in \mathbf{R}$, and $x \in \overline{\Omega}$. A numerical operator \widehat{F} is said to be *consistent* (with the differential operator F) if \widehat{F} satisfies

$$\liminf_{\substack{P^{\mu\nu} \rightarrow P; \mu, \nu = -, + \\ q^\pm \rightarrow q, \lambda \rightarrow v, \xi \rightarrow x}} \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, \lambda, \xi) \geq F_*(P, q, v, x), \tag{9}$$

$$\limsup_{\substack{P^{\mu\nu} \rightarrow P; \mu, \nu = -, + \\ q^\pm \rightarrow q, \lambda \rightarrow v, \xi \rightarrow x}} \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, \lambda, \xi) \leq F_*(P, q, v, x), \tag{10}$$

where F_* and F^* denote, respectively, the lower and the upper semi-continuous envelopes of F . Thus, we have

$$F_*(P, q, v, x) := \liminf_{\substack{\widetilde{P} \rightarrow P, \widetilde{q} \rightarrow q, \\ \widetilde{v} \rightarrow v, \widetilde{x} \rightarrow x}} F(\widetilde{P}, \widetilde{q}, \widetilde{v}, \widetilde{x}),$$

$$F^*(P, q, v, x) := \limsup_{\substack{\widetilde{P} \rightarrow P, \widetilde{q} \rightarrow q, \\ \widetilde{v} \rightarrow v, \widetilde{x} \rightarrow x}} F(\widetilde{P}, \widetilde{q}, \widetilde{v}, \widetilde{x}),$$

where $\widetilde{P} \in \mathbf{R}^{d \times d}$, $\widetilde{q} \in \mathbf{R}^d$, $\widetilde{v} \in \mathbf{R}$, and $\widetilde{x} \in \Omega$. Note that when F and \widehat{F} are continuous, the above definition can be simplified to

$$\widehat{F}(P, P, P, P, q, q, v, x) = F(P, q, v, x). \tag{11}$$

(iii) A numerical operator \widehat{F} is said to be *g-monotone* if for all $x \in \Omega$, there holds $\widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, v, x)$ is monotone increasing in P^{++} , P^{--} , q^- , and v and monotone decreasing in P^{+-} , P^{-+} , and q^+ . More precisely, the numerical operator \widehat{F} is g-monotone if for all $P^{\mu\nu} \in \mathbf{R}^{d \times d}$ and $q^\mu \in \mathbf{R}^d$, $\mu, \nu \in \{+, -\}$, for all $v \in \mathbf{R}$, and for all $x \in \Omega$, there holds

$$\begin{aligned} \widehat{F}(A, P^{+-}, P^{-+}, P^{--}, q^+, q^-, v, x) &\leq \widehat{F}(B, P^{+-}, P^{-+}, P^{--}, q^+, q^-, v, x), \\ \widehat{F}(P^{++}, A, P^{-+}, P^{--}, q^+, q^-, v, x) &\geq \widehat{F}(P^{++}, B, P^{-+}, P^{--}, q^+, q^-, v, x), \\ \widehat{F}(P^{++}, P^{+-}, A, P^{--}, q^+, q^-, v, x) &\geq \widehat{F}(P^{++}, P^{+-}, B, P^{--}, q^+, q^-, v, x), \\ \widehat{F}(P^{++}, P^{+-}, P^{-+}, A, q^+, q^-, v, x) &\leq \widehat{F}(P^{++}, P^{+-}, P^{-+}, B, q^+, q^-, v, x), \end{aligned}$$

for all $A, B \in S^{d \times d}$ such that $A \leq B$, where $A \leq B$ means that $B - A$ has all nonnegative components,

$$\begin{aligned} \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, a, q^-, v, x) &\geq \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, b, q^-, v, x), \\ \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, a, v, x) &\leq \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, b, v, x), \end{aligned}$$

for all $a, b \in \mathbf{R}^d$ such that $a_i \leq b_i$ for all $i = 1, 2, \dots, d$, and

$$\widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, a, x) \leq \widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, b, x)$$

for all $a, b \in \mathbf{R}$ such that $a \leq b$.

The condition can be summarized by $\widehat{F}(\uparrow, \downarrow, \downarrow, \uparrow, \downarrow, \uparrow, \uparrow, x)$, where the monotonicity with respect to the matrix entries is enforced component-wise.

The final concern for the framework is how to design numerical operators that are both consistent and g-monotone. Inspired by Lax–Friedrichs numerical Hamiltonians used for

Hamilton–Jacobi equations [27], we propose the following Lax–Friedrichs-like numerical operator:

$$\widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, \lambda, \xi) := F\left(\frac{P^{-+} + P^{+-}}{2}, \frac{q^- + q^+}{2}, \lambda, \xi\right) - \beta \cdot (q^- - q^+) + \alpha : (P^{++} - P^{+-} - P^{-+} + P^{--}), \tag{12}$$

where $\alpha \in \mathbf{R}^{d \times d}$ is an undetermined positive semi-definite matrix and $\beta \in \mathbf{R}^d$ is an undetermined nonnegative vector. $A : B$ stands for the Frobenius inner product for matrices $A, B \in \mathbf{R}^{d \times d}$. The second to last term $\beta \cdot (q^- - q^+)$ is referred to as the numerical viscosity and is directly borrowed from Lax–Friedrichs numerical Hamiltonians, and the last term $\alpha : (P^{++} - P^{+-} - P^{-+} + P^{--})$ is referred to as the *numerical moment*. It is trivial to verify that \widehat{F} is consistent with F when F is continuous. By choosing α and β correctly, we can also ensure g -monotonicity. In practice, we typically choose $\beta = b\mathbf{1}$ and $\alpha = a_1 I + a_2 \mathbf{1}$ for sufficiently large positive constants a_1, a_2 , and b , where $\mathbf{1}$ is the vector/matrix with all entries equal to one and I is the identity matrix. We note that the g -monotonicity condition can be realized for a_2 sufficiently large and $a_1 = 0$. By also choosing a_1 large, we can additionally enforce the g -monotonicity condition using the partial order based on SPD matrices.

- Remark 2** (a) Due to the definition of ellipticity for F , the g -monotonicity constraints on \widehat{F} with respect to P_{ii}^{-+} and P_{ii}^{+-} are natural. Consistency is used to pass to a single matrix argument and ellipticity is used to guarantee the correct monotonicity with respect to the partial ordering induced by SPD matrices.
- (b) By choosing the numerical viscosity and the numerical moment correctly, the numerical operator \widehat{F} will behave like a strongly elliptic operator even if the PDE operator F is a degenerate elliptic operator. The consistency assumption then guarantees that the numerical operator is still a reasonable approximation for the PDE operator.
- (c) When F is differentiable, while it may not be possible to globally bound $\frac{\partial F}{\partial v_u}$ and $\frac{\partial F}{\partial D^2 u}$, it may be sufficient to choose values for β and α such that the g -monotonicity property is preserved over each iteration of the nonlinear solver for a given initial guess. The same remark holds if F is locally Lipschitz. Thus, the values depend upon the solution and not just the operator F . Similarly, the values for β and α could be chosen locally as done with adaptively defined numerical viscosities for Hamilton–Jacobi equations.

4 Formulation of Nonstandard LDG Methods for Elliptic PDEs

We now formulate our nonstandard LDG methods for approximating viscosity solutions of fully nonlinear elliptic PDEs which are based on the mixed formulation (7) and (8). We also provide a detailed explanation of how to treat the boundary traces in the formulation. Lastly we use the DG formulation to better understand the numerical viscosity and numerical moment appearing in our Lax–Friedrichs-like numerical operator and explore two algorithms for solving the resulting nonlinear algebraic systems.

4.1 DG Notation

To formulate our LDG methods, we need to introduce some notation and conventions which are standard and can be found in [12]. Let Ω be a polygonal domain and \mathcal{T}_h denote a locally quasi-uniform and shape-regular partition of Ω with $h = \max_{K \in \mathcal{T}_h} (\text{diam}K)$. We introduce the broken H^1 -space and broken C^0 -space

$$H^1(\mathcal{T}_h) := \prod_{K \in \mathcal{T}_h} H^1(K), \quad C^0(\mathcal{T}_h) := \prod_{K \in \mathcal{T}_h} C^0(\bar{K})$$

and the broken L^2 -inner product

$$(v, w)_{\mathcal{T}_h} := \sum_{K \in \mathcal{T}_h} \int_K v w \, dx \quad \forall v, w \in L^2(\mathcal{T}_h).$$

Let \mathcal{E}_h^I denote the set of all interior faces/edges of \mathcal{T}_h , \mathcal{E}_h^B denote the set of all boundary faces/edges of \mathcal{T}_h , and $\mathcal{E}_h := \mathcal{E}_h^I \cup \mathcal{E}_h^B$. Then, for a set $\mathcal{S}_h \subset \mathcal{E}_h$, we define the broken L^2 -inner product over \mathcal{S}_h by

$$(v, w)_{\mathcal{S}_h} := \sum_{e \in \mathcal{S}_h} \int_e v w \, ds \quad \forall v, w \in L^2(\mathcal{S}_h).$$

For a fixed integer $r \geq 0$, we define the standard DG finite element space $V^h \subset H^1(\mathcal{T}_h) \subset L^2(\Omega)$ by

$$V^h := \prod_{K \in \mathcal{T}_h} \mathcal{P}_r(K),$$

where $\mathcal{P}_r(K)$ denotes the set of all polynomials on K with degree not exceeding r .

For $K, K' \in \mathcal{T}_h$, let $e = \partial K \cap \partial K' \in \mathcal{E}_h^I$. Without a loss of generality, we assume that the global labeling number of K is smaller than that of K' and define the following (standard) jump and average notations:

$$[v] := v|_K - v|_{K'}, \quad \{v\} := \frac{v|_K + v|_{K'}}{2} \tag{13}$$

for any $v \in H^m(\mathcal{T}_h)$. We also define $n_e := n_K = -n_{K'}$ as the normal vector to e . When $e \in \mathcal{E}_h^B$, n_e denotes the unit outward normal for the underlying boundary simplex. We note that the function values defined on \mathcal{E}_h^B will be handled in a nonstandard way in our LDG methods by allowing the boundary function values to depend on the degree of the polynomial basis r . However, when $r \geq 1$, the boundary function values can be treated in a more standard way as in [12].

4.2 Formulation of LDG Methods

We now present an element-wise formulation for our LDG methods. First we introduce some local definitions. For any $e \in \mathcal{E}_h^I$ with $e = \partial K \cap \partial K'$ for some $K, K' \in \mathcal{T}_h$ and for any $v \in V^h$, let $v(x^I)$ denote the value of $v(x)$ on ∂K from the interior of the element K and $v(x^E)$ denote the value of $v(x)$ on ∂K from the interior of the element K' . Using these limit definitions, we then define the local boundary flux operators: $T^+, T^- : \mathcal{P}_r(K) \rightarrow (\prod_{e \subset \partial K} \mathcal{P}_r(e))^d$ by

$$T_i^-(v_h)(x) := \begin{cases} v_h(x^I), & \text{if } n_i(x) > 0, \\ v_h(x^E), & \text{if } n_i(x) < 0, \\ \{v_h(x)\}, & \text{if } n_i(x) = 0, \end{cases} \tag{14a}$$

$$T_i^+(v_h)(x) := \begin{cases} v_h(x^E), & \text{if } n_i(x) > 0, \\ v_h(x^I), & \text{if } n_i(x) < 0, \\ \{v_h(x)\}, & \text{if } n_i(x) = 0 \end{cases} \tag{14b}$$

for all $i \in \{1, 2, \dots, d\}$, $x \in e$, and $v_h \in V^h$. The definition of $T_i^\pm(v)$ for $v \in V^h$ on each $e \in \mathcal{E}_h^B$ will be delayed to Sect. 4.3. Observe that, for $e \in \mathcal{E}_h^I$, we can also rewrite the labelling-dependent trace operators as

$$T_i^\pm(v_h) = \{v_h\} \mp \frac{1}{2} \operatorname{sgn}(n_e^{(i)})[v_h] \quad \text{where} \quad \operatorname{sgn}(y) = \begin{cases} 1 & \text{if } y > 0, \\ -1 & \text{if } y < 0, \\ 0 & \text{if } y = 0 \end{cases} \quad (15)$$

for all $y \in \mathbf{R}$, where $n_e^{(i)}$ denotes the i -th component of n_e (the unit outward normal to e). Note that the trace operators are nonstandard in that their values depend on the individual components of the edge normal n_e . The standard definition assigns a single-value (called a numerical flux) based on the edge normal vector as a whole.

We are now ready to formulate our LDG methods for system (7)–(8). First, we approximate the (fully) nonlinear Eq. (8) by its broken L^2 -projection into V^h , namely,

$$a_0(u_h, q_h^+, q_h^-, P_h^{++}, P_h^{+-}, P_h^{-+}, P_h^{--}; \phi_{0h}) = 0 \quad \forall \phi_{0h} \in V^h, \quad (16)$$

where

$$\begin{aligned} & a_0(u, q^+, q^-, P^{++}, P^{+-}, P^{-+}, P^{--}; \phi_0) \\ &= (\widehat{F}(P^{++}, P^{+-}, P^{-+}, P^{--}, q^+, q^-, u, \cdot), \phi_0)_{\mathcal{T}_h}. \end{aligned}$$

Next, we discretize the six linear equations in (7) locally with respect to each component using the integration by parts formula:

$$\int_S v_{x_i} \varphi \, dx = \int_{\partial S} v \varphi n_i \, ds - \int_S v \varphi_{x_i} \, dx \quad \forall \varphi \in C^1(S) \quad (17)$$

for $i = 1, 2, \dots, d$. Thus, the above formula yields an integral characterization for the partial derivative v_{x_i} on the set S for all $v \in H^1(S)$. Using the preceding identity, we define our gradient approximations $q_h^\mu \in (V^h)^d$, $\mu \in \{+, -\}$, by

$$\int_K q_i^\mu \phi_i^\mu \, dx + \int_K u (\phi_i^\mu)_{x_i} \, dx = \int_{\partial K} T_i^\mu(u) n_i \phi_i^\mu(x^I) \, ds \quad \forall \phi_i^\mu \in V^h \quad (18)$$

for $i = 1, 2, \dots, d$, $\mu = +, -$.

Similarly, we define our Hessian approximations $P_h^{\mu v} \in (V^h)^{d \times d}$, $\mu, v \in \{+, -\}$, by

$$\int_K P_{i,j}^{\mu v} \psi_{i,j}^{\mu v} \, dx + \int_K q_i^\mu (\psi_{i,j}^{\mu v})_{x_j} \, dx = \int_{\partial K} T_j^v(q_i^\mu) n_j \psi_{i,j}^{\mu v}(x^I) \, ds \quad (19)$$

for all $\psi_{i,j}^{\mu v} \in V^h$ and $i, j = 1, 2, \dots, d$, $\mu, v = +, -$.

Thus, in order to approximate the viscosity solution u for the fully nonlinear PDE (1) paired with a Dirichlet boundary condition

$$u = g \quad \text{on } \partial\Omega \quad (20)$$

for a given function $g \in C^0(\partial\Omega)$, we seek functions $u_h \in V^h$; $q_h^+, q_h^- \in (V^h)^d$; and $P_h^{++}, P_h^{+-}, P_h^{-+}, P_h^{--} \in (V^h)^{d \times d}$ such that Eq. (16) holds as well as Eqs. (18) and (19) for all $K \in \mathcal{T}_h$, where u_h forms the approximation for u . We note that the implementation of the Dirichlet boundary condition into the definition of the boundary flux/trace operator in (18) and (19) will be described in Sect. 4.3.

By summing the definitions of q_h^\pm and $P_h^{\mu,v}$ over \mathcal{T}_h and using (15), we obtain the following global (labeling-dependent) formulations for the proposed LDG methods:

$$(q_i^\mu, \varphi_i^\mu)_{\mathcal{T}_h} + a_i^\mu(u_h, \varphi_i^\mu) = 0 \quad \forall \varphi_i^\mu \in V^h, \tag{21a}$$

$$(P_{ij}^{\mu\nu}, \psi_{ij}^{\mu\nu})_{\mathcal{T}_h} + a_j^\nu(q_i^\mu, \psi_{ij}^{\mu\nu}) = 0 \quad \forall \psi_{ij}^{\mu\nu} \in V^h \tag{21b}$$

for $i, j = 1, 2, \dots, d$ and $\mu, \nu = -, +$, where

$$a_i^\pm(v, \phi) := (v, \phi_{x_i})_{\mathcal{T}_h} - \left\{ \{v\} \mp \frac{1}{2} \operatorname{sgn}(n_e^{(i)})[v], [\phi]n_e^{(i)} \right\}_{\mathcal{E}_h^I} - (T_i^\pm(v), \phi n_i)_{\mathcal{E}_h^B} \tag{22}$$

for all $v, \phi \in V^h$. Then, the proposed LDG methods correspond to solving the global formulation (16) and (21).

Remark 3 Since the approximations are piecewise totally discontinuous polynomials, the sided limits in (7) only need to be enforced along the faces/edges. By [12], we know that the proposed auxiliary variables provide proper meanings for the limits in (7) since the various derivative approximations coincide with the L^2 projections of distributional derivatives onto V^h with variable strengths on the interior faces/edges depending on the choices of the traces, where the traces are chosen such that the sided limits in (7) are consistent.

4.3 Numerical Boundary Fluxes

In this section, we extend the definition for the boundary flux operators, given by (14), to the set \mathcal{E}_h^B . To this end, we will introduce a set of constraint equations that express all exterior limits in terms of interior limits and known data. The Dirichlet boundary data will serve as an exterior constraint on the sought-after numerical solution. We will consider two cases based on whether the order of the DG space V^h is zero or nonzero, i.e., $r = 0$ or $r \geq 1$. When $r \geq 1$, we will enforce a “continuity” assumption across the boundary $\partial\Omega$, and when $r = 0$, we will prescribe an alternative approach that will more closely resemble the introduction of “ghost values” commonly used in FD methods.

Prior to introducing the constraint equations, we specify a convention to be used for all boundary faces/edges. Let $K \in \mathcal{T}_h$ be a boundary simplex, and let $e \in \mathcal{E}_h^B$ such that $e \subset \partial K$. Suppose $v_h \in V^h$ such that v_h is supported on K . Then, we define $v_h(x) := v_h(x^I)$ for all $x \in e$.

We first consider $r \geq 1$, in which case we make the “continuity” assumption

$$v_h(x^E) = v_h(x) \tag{23}$$

for all $x \in e$ and $v_h \in V^h$ such that $e \in \mathcal{E}_h^B$. Since problem (1) and (20) does not provide a Neumann boundary data, we simply treat $q_i^\pm(x)$ as an unknown for all $i = 1, 2, \dots, d$ and $x \in e$ with $e \in \mathcal{E}_h^B$. Alternatively, when defining the boundary flux values for u_h , we use the Dirichlet boundary condition given by (20). Thus, for $r \geq 1$, we wish to impose

$$u_h(x) = g(x)$$

for all $x \in \partial\Omega$. However, g may not be a polynomial of degree r . Thus, we enforce this condition weakly by imposing the following constraint equations:

$$\sum_{i=1}^d (u_h(x), \varphi_h(x)n_i)_{\mathcal{E}_h^B} = \sum_{i=1}^d (g(x), \varphi_h(x)n_i)_{\mathcal{E}_h^B} \quad \forall \varphi_h \in V^h, \tag{24}$$

where n denotes the unit outward normal vector along $\partial\Omega$. Observe that when a boundary simplex has more than one face/edge in \mathcal{E}_h^B , we are treating all of the boundary simplex’s faces/edges in \mathcal{E}_h^B as a single $(d-1)$ -dimensional surface.

We now consider the case $r = 0$. Extending the definition for the boundary flux operators, given by (14), to the set \mathcal{E}_h^B is less straightforward in this case. We can see this by observing the fact that when fixing the interior limit of a boundary value on a boundary simplex, we actually fix the function value on the entire simplex. Thus, strictly enforcing a Dirichlet boundary condition for u_h may result in a boundary layer with respect to the overall approximation error when measured in low-order norms such as the L^∞ - or L^2 -norm. Our goal is to prescribe boundary flux values in a way that results in a potential boundary layer that corresponds to only high-order error, i.e., boundary layers that only appear when measuring the approximation error in the $W^{1,\infty}$ - or H^1 -semi-norms, when defined.

In order to motivate our choice of boundary flux values when $r = 0$, we observe that, for this special case, the DG gradient approximations q_h^\pm are actually equivalent to the forward and backward difference quotients used in FD methods for interior simplexes when \mathcal{T}_h is a Cartesian partition labelled with the natural ordering (see [12]). By extending the equivalence of the proposed LDG methods and the FD methods defined in [9, 19] to the boundary of the domain, we can derive the necessary boundary flux values for u_h and q_h^\pm on \mathcal{E}_h^B . To this end, we will need to develop a methodology for extending the solution u to the exterior of the domain Ω . We now define a way to do such an extension that is consistent with the interpretation of the auxiliary variables and consistent with the FD strategy of introducing “ghost values” for a grid function, where the underlying grid will be defined by the midpoints of the Cartesian partition \mathcal{T}_h .

We first describe the extension for the approximation function u_h . Given the Dirichlet boundary data for the viscosity solution u , it is natural to assume that the approximation function u_h has a constant extension beyond each individual boundary face/edge. Thus, we wish to define the exterior boundary fluxes using the Dirichlet boundary condition by setting $u(x^E) = g(x)$ for all $x \in \partial K \cap \partial\Omega$. However, a given boundary simplex may have multiple faces/edges in \mathcal{E}_h^B . Therefore, we introduce a “ghost simplex” exterior to each individual face/edge in \mathcal{E}_h^B , and we define the exterior value as g_e , where

$$\sum_{i=1}^d \langle g_e, n_e^{(i)} \rangle_e = \sum_{i=1}^d \langle g, n_e^{(i)} \rangle_e \quad \forall e \in \mathcal{E}_h^B. \tag{25}$$

Then, we define

$$u_h(x^E)|_e := g_e \quad \forall e \in \mathcal{E}_h^B. \tag{26}$$

Observe that, for $r = 0$, we only apply the Dirichlet boundary condition to the exterior function limits. Furthermore, we define the exterior function limits to be edge-dependent. Since the function value is constant on each simplex K , we do not extend the Dirichlet boundary condition to the interior of the domain by strongly enforcing (20). Instead, we treat the value of u_h on K as an unknown whenever K is a boundary simplex. We use the edge-dependent definition to mimic the use of ghost values when $r = 0$, which are introduced for each coordinate direction when using a FD methodology. When \mathcal{T}_h is a Cartesian partition, our methodology does in fact result in the introduction of a fixed exterior boundary flux value for each individual coordinate direction. The result of the methodology will be a more weighted approximation on a boundary simplex based upon the boundary condition along each boundary face/edge independently and on the PDE for the interior of the simplex.

Next, we describe how we assign boundary values for q_h^\pm for $r = 0$. Since we do not have Neumann boundary data, we will have to enforce auxiliary boundary conditions. Assuming \mathcal{T}_h is a Cartesian partition labelled with the natural ordering, throughout the interior of the domain there holds

$$q_i^-|_K = q_i^+|_{K_i^-}, \quad q_i^+|_K = q_i^-|_{K_i^+} \tag{27}$$

for all $i = 1, 2, \dots, d$ and all interior simplexes $K \in \mathcal{T}_h$ due to the equivalence with FD forward and backward difference quotients, where K_i^- denotes the neighboring simplex in the negative i -th Cartesian direction and K_i^+ denotes the neighboring simplex in the positive i -th Cartesian direction. Extending (27) to the boundary yields

$$q_i^-(x^E) = q_i^+(x^I), \quad \text{if } n_e^{(i)} < 0, \tag{28a}$$

$$q_i^+(x^E) = q_i^-(x^I), \quad \text{if } n_e^{(i)} > 0 \tag{28b}$$

for $x \in e$, where both $q_i^+(x^I)$ and $q_i^-(x^I)$ are treated as unknowns. We will assume such a relationship holds along the boundary for all triangulations. We also note that the relationship is arbitrary if $n_i^{(i)} = 0$.

Observe that the above extension does not define exterior limits for q_i^+ if $n_e^{(i)} < 0$ or q_i^- if $n_e^{(i)} > 0$. In order to define the remaining exterior limit values, we impose the following auxiliary constraint equations:

$$\sum_{i=1}^d \left\langle q_i^-(x^I) - q_i^-(x^E), n_e^{(i)} \right\rangle_e = 0 \quad \forall e \in \mathcal{E}_h^B, \tag{29a}$$

$$\sum_{i=1}^d \left\langle q_i^+(x^I) - q_i^+(x^E), n_e^{(i)} \right\rangle_e = 0 \quad \forall e \in \mathcal{E}_h^B. \tag{29b}$$

The above constraint equations are consistent with discretizing the higher order auxiliary constraint for all ghost-values of q_h^\pm :

$$\sum_{k=1}^d (q_k^\pm)_{x_k}(x) = 0 \quad \forall x \in \Omega^c.$$

The philosophy for such an auxiliary assumption can be found in [13]. We note that the constraint equations (29) are also trivially satisfied when defining the exterior values for $r \geq 1$ due to our “continuity” assumption. Assuming that \mathcal{T}_h is either a uniform Cartesian partition or a d -triangular partition where each simplex has at most one face/edge in \mathcal{E}_h^B , we can see that all exterior limits on the boundary of the domain have now been expressed in terms of unknown interior limits that correspond to degrees of freedom for the discretization.

We end this section by explicitly specifying the resulting exterior limit definitions for q_h^+ and q_h^- when approximating a two-dimensional problem with piecewise constant basis functions. The explicit definitions for one-dimensional problems can be found in [10]. Let $q_i^\pm := (q_h^\pm)_i$. Then, using the strategy given above, we have

$$\begin{aligned} q_1^+(x^E) &= q_1^-(x^I), & q_2^+(x^E) &= q_2^-(x^I), \\ q_1^-(x^E) &= q_1^-(x^I), & q_2^-(x^E) &= q_2^-(x^I), \end{aligned}$$

if $n_1(x) < 0$ and $n_2(x) < 0$,

$$\begin{aligned} q_1^+(x^E) &= q_1^-(x^I), & q_2^+(x^E) &= q_2^+(x^I) + q_1^+(x^I) - q_1^+(x^E), \\ q_1^-(x^E) &= q_1^-(x^I) + q_2^-(x^I) - q_2^-(x^E), & q_2^-(x^E) &= q_2^+(x^I) \end{aligned}$$

if $n_1(x) < 0$ and $n_2(x) \geq 0$,

$$\begin{aligned} q_1^-(x^E) &= q_1^+(x^I), & q_2^-(x^E) &= q_2^-(x^I) + q_1^-(x^I) - q_1^-(x^E), \\ q_1^+(x^E) &= q_1^+(x^I) + q_2^+(x^I) - q_2^+(x^E), & q_2^+(x^E) &= q_2^-(x^I) \end{aligned}$$

if $n_1(x) \geq 0$ and $n_2(x) < 0$, and

$$\begin{aligned} q_1^-(x^E) &= q_1^+(x^I), & q_2^-(x^E) &= q_2^+(x^I), \\ q_1^+(x^E) &= q_1^+(x^I), & q_2^+(x^E) &= q_2^+(x^I) \end{aligned}$$

if $n_1(x) \geq 0$ and $n_2(x) \geq 0$ for all $x \in \partial\Omega \cap e$ for some $e \in \mathcal{E}_h^B$.

Remark 4 (a) When $r = 0$, our approximation space consists of totally discontinuous piecewise constant functions. We have prescribed a way to assign all exterior boundary flux values for our approximation functions, and, by convention, we treat all interior boundary flux values as unknowns.

(b) The above constraint equations occur naturally in the boundary edge terms for the bilinear form (22) for each auxiliary variable. We use this observation to enforce our boundary conditions for u_h and q_h^\pm in the numerical tests found in Sect. 6.

4.4 The Numerical Viscosity and Numerical Moment

In this section, we take a closer look at the numerical viscosity and the numerical moment used in the definition of the Lax–Friedrichs-like numerical operator (12). We divide the analysis into two cases, $r = 0$ and $r \geq 1$. When $r = 0$, we will recover vanishing FD approximations of the Laplacian operator and the biharmonic operator. When $r \geq 1$, we will recover interior jump/stabilization terms.

First we consider the case $r = 0$ in the definition of V^h . Suppose that \mathcal{T}_h is a uniform Cartesian partition labelled using the natural ordering. Let K be an interior simplex, x_K denote its midpoint, and χ_K denote the characteristic function on K . Then, by [12], we have

$$\begin{aligned} -\beta \cdot (q_h^+ - q_h^-, \chi_K)_{\mathcal{T}_h} &= - \sum_{i=1}^d \beta_i (\delta_{x_i, h_i}^+ u_h(x_K) - \delta_{x_i, h_i}^- u_h(x_K)) \\ &= \sum_{i=1}^d \beta_i h_i \delta_{x_i, h_i}^2 u_h(x_K), \end{aligned}$$

where δ_{x_i, h_i}^+ denotes the forward difference quotient operator, δ_{x_i, h_i}^- denotes the backward difference quotient operator, and δ_{x_i, h_i}^2 denotes the standard second order central difference quotient operator for approximating pure second derivatives. Also, by [12], we have

$$\begin{aligned} \alpha &: (P_{ij}^{++} - P_{ij}^{+-} - P_{ij}^{-+} + P_{ij}^{--}, \chi_K)_{\mathcal{T}_h} \\ &= \sum_{i,j=1}^d \alpha_{ij} (\delta_{x_i, h_i}^+ \delta_{x_j, h_j}^+ u_h(x_K) - \delta_{x_i, h_i}^+ \delta_{x_j, h_j}^- u_h(x_K) \\ &\quad - \delta_{x_i, h_i}^- \delta_{x_j, h_j}^+ u_h(x_K) + \delta_{x_i, h_i}^- \delta_{x_j, h_j}^- u_h(x_K)) \\ &= \sum_{i,j=1}^d \alpha_{i,j} h_i h_j \delta_{x_i, h_i}^2 \delta_{x_j, h_j}^2 u_h(x_K). \end{aligned}$$

Thus, for $\beta = \mathbf{1}$ and $\alpha = \mathbf{1}$, we recover scaled approximations for the Laplace and biharmonic operator. Consequently, the Lax–Friedrichs-like numerical operator is a direct realization of the vanishing moment method (cf. [13,14]) combined with the vanishing viscosity method from Hamilton–Jacobi equations (cf. [3]).

A similar consequence of the relationship with FD when $r = 0$ and \mathcal{T}_h corresponds to a uniform Cartesian grid labelled using the natural ordering is that

$$(P_h^{\pm\mp})_{ii} = \frac{1}{h_i} (q_h^- - q_h^+)_i \quad \text{for } i = 1, 2, \dots, d.$$

Thus, if \widehat{F} is defined by (12), then \widehat{F} may implicitly be monotone increasing with respect to q_h^+ and monotone decreasing with respect to q_h^- for $\beta = \mathbf{0}$ as long as h_i is sufficiently small and $\alpha_{ii} > 0$ for all $i = 1, 2, \dots, d$. In other words, the numerical moment can implicitly enforce the g-monotonicity requirements for q_h^\pm . We exploit this observation in Sect. 6 by choosing $\beta = \mathbf{0}$ in our numerical tests. Heuristically, we expect the corresponding FD schemes to be limited to 1st order accuracy when the numerical viscosity is present (as with Lax–Friedrichs schemes for Hamilton–Jacobi equations), whereas the corresponding FD schemes may be capable of 2nd order accuracy when only the numerical moment is present. Such an observation is supported by the numerical tests found later in Sect. 6 as well as the numerical tests of the FD methods found in [19].

We now consider the case $r \geq 1$ in the definition of V^h . Let $i \in \{1, 2, \dots, d\}$. Observe that by the boundary conditions from Sect. 4.3, we have

$$(q_i^+ - q_i^-, \phi)_{\mathcal{T}_h} = a_i^+(u_h, \phi) - a_i^-(u_h, \phi) = \left\langle [u_h], [\phi] |n_e^{(i)}| \right\rangle_{\mathcal{E}_h^I}.$$

Thus,

$$-\beta \cdot (q_h^- - q_h^+, \phi)_{\mathcal{T}_h} = \sum_{i=1}^d \beta_i \left\langle [u_h], [\phi] |n_e^{(i)}| \right\rangle_{\mathcal{E}_h^I}. \tag{30}$$

Similarly, for $i, j \in \{1, 2, \dots, d\}$,

$$\begin{aligned} &(P_{i,j}^{++} - P_{i,j}^{+-} - P_{i,j}^{-+} + P_{i,j}^{--}, \phi)_{\mathcal{T}_h} \\ &= a_j^+(q_i^+, \phi) - a_j^-(q_i^+, \phi) - a_j^+(q_i^-, \phi) + a_j^-(q_i^-, \phi) \\ &= \left\langle [q_i^+], [\phi] |n_e^{(j)}| \right\rangle_{\mathcal{E}_h^I} - \left\langle [q_i^-], [\phi] |n_e^{(j)}| \right\rangle_{\mathcal{E}_h^I}. \end{aligned}$$

Thus,

$$\alpha : (P_{i,j}^{++} - P_{i,j}^{+-} - P_{i,j}^{-+} + P_{i,j}^{--}, \phi)_{\mathcal{T}_h} = \sum_{i,j=1}^d \alpha_{i,j} \left\langle [q_i^+ - q_i^-], [\phi] |n_e^{(j)}| \right\rangle_{\mathcal{E}_h^I}. \tag{31}$$

From above, we can see that

$$\begin{aligned}
 & a_0(u_h, q_h^-, q_h^+, P_h^{--}, P_h^{-+}, P_h^{+-}, P_h^{++}; \phi_h) \\
 &= (F(P_h, q_h, u_h, \cdot), \phi_h)_{\mathcal{T}_h} + \sum_{i=1}^d \beta_i \left([u_h], [\phi_h] |n_e^{(i)}| \right)_{\mathcal{E}_h^I} \\
 &+ \sum_{i,j=1}^d \alpha_{i,j} \left([q_i^+ - q_i^-], [\phi_h] |n_e^{(j)}| \right)_{\mathcal{E}_h^I}, \tag{32}
 \end{aligned}$$

where

$$P_h = \frac{P_h^{+-} + P_h^{-+}}{2}, \quad q_h = \frac{q_h^+ + q_h^-}{2},$$

and q_h^+, q_h^- are both approximations for ∇u . Thus, adding a numerical moment and a numerical viscosity amounts to the addition of interior jump/stabilization terms to an L^2 -projection of the fully nonlinear PDE operator into V^h . We do note that the jump/stabilization terms that arise due to the numerical moment penalize the differences in q_h^+ and q_h^- . Thus, the numerical moment is not analogous to a high order penalization term that penalizes jumps in a single approximation for ∇u , as sometimes used in interior penalty methods. Instead, the numerical moment penalizes the difference in two optimal DG approximations for ∇u (cf. [12]). We remark that this new jump term is the distinguishing characteristic of the proposed LDG methods since it was not possible to obtain an analogous result for the IPDG framework proposed in [11].

4.5 Solvers

We now discuss different strategies for solving the nonlinear system of equations that results from the proposed LDG discretization for the elliptic problem. The underlying goal for the methodology presented in this paper is to discretize the fully nonlinear PDE problem in a way that removes much of the burden of approximating viscosity solutions from the design of the solver. Thus, our primary focus is at the discretization level. However, some of the properties of the methodology are more apparent from the solver perspective.

Most tests show that it is sufficient to simply use a Newton solver on the full system of Eqs. (16) and (21). Observe that only (16) is nonlinear, the equation is purely algebraic, and \widehat{F} is monotone in seven of its arguments. The auxiliary equations (21) are all linear. The numerical operator presented in this paper is symmetric in both the mixed approximations P_h^{-+} and P_h^{+-} and the non-mixed approximations P_h^{--} and P_h^{++} . Thus, we can reduce the size of the system of equations by averaging the two pairs of auxiliary variables in the above formulation without changing the methodology.

Due to the size of the mixed formulation, we first present a splitting algorithm that provides an alternative to a straightforward Newton solver for the entire system of equations. By using a splitting algorithm, the resulting algorithm will iteratively solve an entirely local, nonlinear equation that has strong monotonicity properties in the d unknown arguments, and the solution of the equation can be mapped to an updated approximation for u_h . Tests show that the solver is particularly useful for nonlinear problems that have a unique viscosity solution only defined in a restrictive function class. For instance, viscosity solutions of the Monge–Ampère equation are unique in the class of convex functions. However, the proposed solver is not as efficient as the second solver we present that takes advantage of the above

nonstandard discretization technique. In order to improve the speed of the solver, fast Poisson solvers for the DWDG method (cf. [20]) need to be developed.

Our second solver strategy is a natural generalization of the FD methodology for numerical PDEs. Constructing and applying the DG derivative operators requires sparse matrix multiplication and addition as well as inverting the local mass matrices. Thus, all auxiliary equations in the mixed formulation can be solved for a given function u_h . Substituting these operators directly into the numerical operator results in a single nonlinear variational problem for u_h that can be solved iteratively.

4.5.1 An Inverse-Poisson Fixed-Point Solver

We now describe the above mentioned splitting algorithm that takes into account the special structure of the nonlinear algebraic system that results from our nonstandard LDG discretization methods for elliptic PDEs and parabolic PDEs when using implicit time-stepping. The algorithm is strongly based upon using a particular numerical moment.

Algorithm 1

1. Pick an initial guess for u_h .
2. Form initial guesses for q_h^+ , q_h^- , P_h^{++} , P_h^{+-} , P_h^{-+} , and P_h^{--} using equations (21).
3. Set

$$G_i := F\left(\frac{P_h^{-+} + P_h^{+-}}{2}, \frac{q_h^- + q_h^+}{2}, u_h, x\right) + \gamma(P_h^{++} - P_h^{+-} - P_h^{-+} + P_h^{--})_{ii} - \beta_i(q_h^- - q_h^+)_i$$

for a fixed constant $\gamma > 0$, and solve

$$(G_i, \varphi_i)_{\mathcal{T}_h} = 0 \quad \forall \varphi_i \in V^h$$

for $\frac{1}{2}(P_h^{-+} + P_h^{+-})_{ii}$ for all $i = 1, 2, \dots, d$. For sufficiently large γ and a differentiable operator F , the above set of equations has a negative definite Jacobian.

4. Find u_h , q_h^+ , and q_h^- by solving the linear system of equations formed by (21a) and the trace of averaging (21b) for $\mu = -, \nu = +$ and $\mu = +, \nu = -$. Observe that this is equivalent to solving Poisson’s equation with source data given by the trace of $\frac{1}{2}(P_h^{-+} + P_h^{+-})$. Alternatively, apply the DWDG method using the trace of $\frac{1}{2}(P_h^{-+} + P_h^{+-})$ as the source data to find u_h .
5. Solve (21b) for P_h^{++} , P_h^{+-} , P_h^{-+} , and P_h^{--} . If the alternative approach in step 4 was used, also solve (21a) for q_h^+ and q_h^- .
6. Repeat Steps 3–5 until the change in $\frac{1}{2}(P_h^{-+} + P_h^{+-})$ is sufficiently small.

We now make a couple of comments about the proposed solver.

Remark 5 (a) The proposed algorithm is well-posed since it is based on the DWDG method which results in a symmetric positive definite discretization of Poisson’s equation (cf. [20]).

- (b) The nonlinear equation in Step 3 is entirely local with respect to the unknown variable.
- (c) Clearly a fixed point for the solver corresponds to a discrete solution of the original PDE problem. In Sect. 6 and in [10] the numerical tests indicate that the proposed solver is less dependent upon the initial guess. The algorithm can also be used to form a preconditioned initial guess for other nonlinear solvers that may be faster but require a “better” initial guess.

4.5.2 A Direct Approach for a Reduced System

In this section, we propose a solver technique that is analogous to the approach used in FD methods. Observe that if $(u_h, q_h^+, q_h^-, P_h^{++}, P_h^{+-}, P_h^{-+}, P_h^{--})$ is a solution to (16) and (21), then there exists linear operators ∇_h^\pm and $D_h^{\mu\nu}$ such that $q_h^\pm = \nabla_h^\pm u_h$ and $P_h^{\mu\nu} = D_h^{\mu\nu} u_h$ for all $\mu, \nu \in \{+, -\}$, where the linear operators are locally defined by (18) and (19).

Using these numerical derivative operators, the second solver is given by:

Algorithm 2

1. Given \mathcal{T}_h and V^h , compute the operators ∇_h^\pm and $D_h^{\mu\nu}$.
2. Solve for $u_h \in V^h$ the single nonlinear equation

$$\left(\widehat{F}(D_h^{++}u_h, D_h^{+-}u_h, D_h^{-+}u_h, D_h^{--}u_h, \nabla_h^+u_h, \nabla_h^-u_h, u_h, \cdot), \varphi_h \right)_{\mathcal{T}_h} = 0 \quad \forall \varphi_h \in V^h.$$

We note that a reduced formulation can also be used where we simply create the following new differential operators:

$$\overline{D}_h^2 := \frac{D_h^{--} + D_h^{++}}{2}, \quad \widetilde{D}_h^2 := \frac{D_h^{-+} + D_h^{+-}}{2}, \quad \nabla_h := \frac{\nabla_h^+ + \nabla_h^-}{2}.$$

The Lax–Friedrichs-like numerical operator can be written as

$$\begin{aligned} & \widehat{F}(\overline{D}_h^2 u_h, \widetilde{D}_h^2 u_h, \nabla_h^+ u_h, \nabla_h^- u_h, u_h, x) \\ &= F(\widetilde{D}_h^2 u_h, \nabla_h u_h, u_h, x) + 2\alpha : (\overline{D}_h^2 u_h - \widetilde{D}_h^2 u_h) - 2\beta \cdot (\nabla_h^+ u_h - \nabla_h^- u_h). \end{aligned} \quad (33)$$

For all of the tests below where a Newton solver is used for the full system of equations in the mixed formulation, analogous results were obtained using Algorithm 2 with the reduced numerical operators. As expected, for two-dimensional problems we observed significant speed-up in the performance of the solver.

Remark 6 The methodology of Algorithm 2 follows directly from the FD methodology where derivatives in a PDE are simply replaced by numerical derivatives of the approximation for the solution u to form the discretization of the PDE problem. For nonlinear problems, we replace the nonlinear PDE operator by a numerical operator. In our LDG setting, we use the LDG methodology to define the various numerical derivatives.

5 An Extension for Parabolic Problems

We now develop fully discrete methods for approximating the parabolic equation (2) complemented by the following boundary condition and initial condition:

$$u(x, t) = g(x), \quad (x, t) \in \Omega_T := \Omega \times (0, T], \quad (34a)$$

$$u(x, 0) = u_0(x), \quad x \in \Omega \quad (34b)$$

using an LDG spatial-discretization paired with the method of lines approach for the time discretization. Taking advantage of the elliptic formulation in Sect. 4, we will propose the following implicit and explicit time-discretizations: forward Euler, backward Euler, trapezoidal, and Runge-Kutta (RK). The time-discretization used in application should be selected according to the potential optimal order $r + 1$ of the LDG spatial-discretization for sufficiently regular viscosity solutions.

We first present the semi-discrete discretization of the (fully) nonlinear equation (2) by discretizing the spatial dimension. Replacing the PDE operator F with a numerical operator \widehat{F} in (2), applying a spatial discretization using the above LDG framework for elliptic equations, and using the L^2 -projection operator $\mathcal{P}_h : L^2(\mathcal{T}_h) \rightarrow V^h$ defined by

$$(\mathcal{P}_h v, \phi_h)_{\mathcal{T}_h} = (v, \phi_h)_{\mathcal{T}_h} \quad \forall \phi_h \in V^h \tag{35}$$

for all $v \in L^2(\mathcal{T}_h)$, we have the following semi-discrete equation

$$(u_h)_t = -\mathcal{P}_h \left(\widehat{F}(P_h^{++}, P_h^{+-}, P_h^{-+}, P_h^{--}, q_h^+, q_h^-, u_h, x, t) \right), \tag{36}$$

where, given u_h at time t , corresponding values for q_h^\pm and $P_h^{\mu\nu}$, $\mu, \nu \in \{+, -\}$, can be found by solving the local equations (18) and (19).

Our full-discretization of the initial-boundary value problem (2), (34a), and (34b) is defined by applying an ODE solver to the semi-discrete (variational) form given in (36). To partition the time domain, we fix an integer $M > 0$ and let $\Delta t = \frac{T}{M}$. Then, we define $t_k := k \Delta t$ for a real number k with $0 \leq k \leq M$. Notationally, $u_h^k \in V^h$ and $q_h^{\pm, k} \in (V^h)^d$ will be an approximation for $u(\cdot, t_k)$ and $\nabla u(\cdot, t_k)$, respectively, for all $0 \leq k \leq M$. For both implicit and explicit schemes, we define the initial value, u_h^0 , by

$$u_h^0 = \mathcal{P}_h u_0. \tag{37}$$

To simplify the appearance of the methods and to make them more transparent for use with a given ODE solver, we use a subscript k to denote the fact that the boundary values are being naturally enforced in (18) and (19) using the boundary condition (34a) evaluated at time t_k , $0 \leq k \leq M$. Thus,

$$\begin{aligned} ((q_{h,k}^\pm)_i, \phi_i^\pm)_{\mathcal{T}_h} &= \left\langle T_i^\pm(u_{h,k}), [\phi_i^\pm] n_e^{(i)} \right\rangle_{\mathcal{E}_h^I} + \left\langle T_i^\pm(u_{h,k}), \phi_i^\pm(x^I) n_i \right\rangle_{\mathcal{E}_h^B} \\ &\quad - (u_{h,k}, (\phi_i^\pm)_{x_i})_{\mathcal{T}_h} \quad \forall \phi_i^\pm \in V^h \end{aligned} \tag{38}$$

for $i = 1, 2, \dots, d$, where we evaluate the boundary flux values using the convention

$$\sum_{i=1}^d (u_{h,k}, \varphi_h n_i)_{\mathcal{E}_h^B} = \sum_{i=1}^d (g(\cdot, t_k), \varphi_h n_i)_{\mathcal{E}_h^B} \quad \forall \varphi_h \in V^h$$

when $r \geq 1$ and

$$\sum_{i=1}^d (u_{h,k}(x^E), n_e^{(i)})_e = \sum_{i=1}^d (g(\cdot, t_k), n_e^{(i)})_e$$

when $r = 0$. Similarly,

$$\begin{aligned} ((P_{h,k}^{\mu\nu})_{ij}, \psi_{ij}^{\mu\nu})_{\mathcal{T}_h} &= \left\langle T_j^\nu((q_{h,k}^\mu)_i), [\psi_{ij}^{\mu\nu}] n_e^{(j)} \right\rangle_{\mathcal{E}_h^I} + \left\langle T_j^\nu(q_{h,k}^\mu)_i, \psi_{ij}^{\mu\nu}(x^I) n_j \right\rangle_{\mathcal{E}_h^B} \\ &\quad - ((q_{h,k}^\mu)_i, (\psi_{ij}^{\mu\nu})_{x_j})_{\mathcal{T}_h} \quad \forall \psi_{ij}^{\mu\nu} \in V^h \end{aligned} \tag{39}$$

for $i, j \in \{1, 2, \dots, d\}$, $\mu, \nu \in \{+, -\}$, where we assume $(q_{h,k}^\pm(x^E))_i = (q_{h,k}^\pm(x))_i$ when $r \geq 1$ or

$$\sum_{i=1}^d \left\langle (q_{h,k}^\pm(x^I))_i - (q_{h,k}^\pm(x^E))_i, n_e^{(i)} \right\rangle_e = 0$$

and

$$\begin{aligned} (q_{h,k}^-(x^E))_i &= (q_{h,k}^+(x^J))_i, & \text{if } n_e^{(i)} < 0, \\ (q_{h,k}^+(x^E))_i &= (q_{h,k}^-(x^J))_i, & \text{if } n_e^{(i)} > 0 \end{aligned}$$

for all $e \in \mathcal{E}_h^B$, using (28) and (29), when $r = 0$. Note, for $k = 0$, we replace $g(\cdot, t_k)$ with $u_0(\cdot)$ in the above constraint equations if u_0 has an L^2 trace. Otherwise, we replace $g(\cdot, t_k)$ with the trace of $\mathcal{P}_h u_0$.

We also simplify the presentation of the fully-discrete methods by introducing the operator notation

$$\widehat{F}^k[v] := \widehat{F} \left(D_{h,k}^{++} v, D_{h,k}^{+-} v, D_{h,k}^{-+} v, D_{h,k}^{--} v, \nabla_{h,k}^+ v, \nabla_{h,k}^- v, v, x, k \Delta t \right) \tag{40}$$

for all $v \in V^h$, where we are introducing linear operators $\nabla_{h,k}^\pm$ and $D_{h,k}^{\mu\nu}$ such that $q_{h,k}^\pm = \nabla_{h,k}^\pm u_h$ and $P_{h,k}^{\mu\nu} = D_{h,k}^{\mu\nu} u_h$ for all $\mu, \nu \in \{+, -\}$, where the linear operators are locally defined by replacing $u_{h,k}$ with an arbitrary function $v_h \in V^h$ in (38) and (39). Then, the semi-discrete equation can be rewritten compactly as

$$(u_h)_t(x, t_k) = -\mathcal{P}_h \widehat{F}^k[u_h(x, t_k)] \quad \forall 0 \leq k \leq M, x \in \Omega. \tag{41}$$

Lastly, we define a modified projection operator $\mathcal{P}_{h,k} : L^2(\mathcal{I}_h) \rightarrow V^h$ that will be used to enforce the boundary conditions for explicit methods using a penalty technique due to Nitsche in [23]. Thus, we define $\mathcal{P}_{h,k}$ by

$$\begin{aligned} (\mathcal{P}_{h,k} v, \varphi_h)_{\mathcal{I}_h} + \delta \sum_{i=1}^d \left(\mathcal{P}_{h,k} v, \varphi_h n_i \right)_{\mathcal{E}_h^B} \\ = (v, \varphi_h)_{\mathcal{I}_h} + \delta \sum_{i=1}^d \left(g(\cdot, t_k), \varphi_h n_i \right)_{\mathcal{E}_h^B} \quad \forall \varphi_h \in V^h \end{aligned} \tag{42}$$

for all $v \in L^2(\mathcal{I}_h)$, where δ is a nonnegative penalty constant and $0 \leq k \leq M$. We note that, for $\delta = 0$, $\mathcal{P}_{h,k} = \mathcal{P}_h$, yielding the broken L^2 -projection operator.

Using the above conventions, we can define fully discrete methods for approximating problem (2), (34a), and (34b) based on approximating (41) using the forward Euler method, backward Euler method, or the trapezoidal method. Thus, we have respectively

$$u_h^{n+1} = \mathcal{P}_{h,n+1} \left(u_h^n - \Delta t \widehat{F}^n [u_h^n] \right), \tag{43}$$

$$u_h^{n+1} + \Delta t \mathcal{P}_h \widehat{F}^{n+1} [u_h^{n+1}] = u_h^n, \tag{44}$$

and

$$u_h^{n+1} + \frac{\Delta t}{2} \mathcal{P}_h \widehat{F}^{n+1} [u_h^{n+1}] = u_h^n - \frac{\Delta t}{2} \mathcal{P}_h \widehat{F}^n [u_h^n] \tag{45}$$

for $n = 0, 1, \dots, M - 1$, where $u_h^0 := \mathcal{P}_h u_0$ and, for (44) and (45), we also have, by (40), the implied auxiliary linear equations

$$\begin{aligned} q_h^{\mu,n} &= \nabla_{h,n}^\mu u_h^n & \forall \mu \in \{+, -\}, \\ P_h^{\mu\nu,n} &= D_{h,n}^{\mu\nu} u_h^n & \forall \mu, \nu \in \{+, -\}. \end{aligned}$$

Remark 7 Using an implicit method, such as the backward Euler and the trapezoidal method, results in approximating a fully nonlinear elliptic PDE at each time step using the LDG

methods for elliptic PDEs formulated in Sect. 4. Due to the time integration, the nonlinear solver has a natural initial guess for each time-step given by the approximation at the previous time step.

Finally, we formulate the Runge-Kutta (RK) methods for approximating (41). Let s be a positive integer, $A \in \mathbf{R}^{s \times s}$, and $b, c \in \mathbf{R}^s$ such that

$$\sum_{\ell=1}^s a_{k,\ell} = c_k$$

for each $k = 1, 2, \dots, s$. Then, a generic s -stage RK method for approximating (41) is defined by

$$u_h^{n+1} = \mathcal{P}_{h,n+1} \left(u_h^n - \Delta t \sum_{\ell=1}^s b_\ell \widehat{F}^{n+c_\ell} [\xi_h^{n,\ell}] \right), \quad n = 0, 1, \dots, N - 1, \tag{46}$$

where

$$\xi_h^{n,\ell} = \mathcal{P}_{h,n+c_k} \left(u_h^n - \Delta t \sum_{k=1}^s a_{k,\ell} \widehat{F}^{n+c_k} [\xi_h^{n,k}] \right), \quad n = 0, 1, \dots, N - 1,$$

and $u_h^0 = \mathcal{P}_h u_0$. We note that (46) corresponds to an explicit method when A is strictly lower diagonal and an implicit method otherwise.

Remark 8 $\xi_h^{n,\ell}$ in (46) can be viewed as an approximation for $u_h^{n+c_\ell}$. Since the boundary condition at t_{n+1} is enforced by \widehat{F}^{n+1} , we can set $\delta = 0$ in (42) if $c_s = 1$.

6 Numerical Experiments

In this section, we present a series of numerical tests to demonstrate the utility of the proposed LDG methods for fully nonlinear PDE problems of type (1) and (2) with two spatial dimensions. For elliptic problems, both Monge–Ampère and Hamilton–Jacobi–Bellman types of equations will be tested. We also perform a test using the (semi-linear) infinite-Laplacian equation with a known low-regularity solution. The tests use spatial meshes composed of uniform rectangles. To solve the resulting nonlinear algebraic systems, we use either the Matlab built-in nonlinear solver *fsolve* or Algorithm 1, where *fsolve* is used to perform Step 3 of Algorithm 1. For the elliptic problems, we choose the initial guess as the zero function. For the parabolic test problem, we choose the initial guess as the approximation formed at the previous time step and use the backward Euler method. We also choose the approximation at time $t = 0$ to be given by the L^2 -projection of the initial condition into V^h .

For our numerical tests, errors will be measured in the L^∞ norm and the L^2 norm. All recorded data corresponds to tests without a numerical viscosity, i.e., $\beta = \mathbf{0}$. Similar results hold when the numerical viscosity is present. For elliptic problems and parabolic problems where the error is not dominated by the time discretization, the test problems in [10] indicate the spatial errors are of order $\mathcal{O}(h^s)$ for most problems, where $s = \min\{r + 1, k\}$ for the viscosity solution $u \in H^k(\Omega)$. In this paper, the computed convergence rates are a little more sporadic. On average, the schemes appear to exhibit an optimal rate of convergence in both norms. We note that the actual convergence rates have not yet been analyzed, and they may also depend on the regularity of the differential operator F and the severity of its nonlinearity in addition to the regularity of the viscosity solution u .

Table 1 Rates of convergence for Example 1 using $r = 0, \alpha = 24I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
1.41e-01	3.73e-01		8.31e-02	
8.84e-02	2.42e-01	0.92	5.10e-02	1.04
5.89e-02	1.64e-01	0.95	3.31e-02	1.06
4.42e-02	1.24e-01	0.97	2.44e-02	1.07

Table 2 Rates of convergence for Example 1 using $r = 1, \alpha = 24I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
1.41e-01	2.47e-02		1.73e-03	
1.18e-01	1.36e-02	3.25	1.61e-03	0.39
1.01e-01	1.03e-02	1.81	1.12e-03	2.31
7.86e-02	8.04e-03	0.99	5.82e-04	2.62

Table 3 Rates of convergence for Example 1 using $r = 2, \alpha = 24I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
7.07e-01	6.39e-02		4.45e-03	
4.71e-01	2.32e-02	2.50	1.30e-03	3.03
3.54e-01	1.09e-02	2.63	5.45e-04	3.02

Example 1 Consider the Monge–Ampère problem

$$\begin{aligned}
 -\det D^2u &= -u_{xx}u_{yy} + u_{xy}u_{yx} = f && \text{in } \Omega, \\
 u &= g && \text{on } \partial\Omega,
 \end{aligned}$$

where $f = -(1+x^2+y^2)e^{x^2+y^2}$, $\Omega = (0, 1) \times (0, 1)$, and g is chosen such that the viscosity solution is given by $u(x, y) = e^{\frac{x^2+y^2}{2}}$.

Notice that the problem has two possible solutions as seen in [13], where the viscosity solution is convex and the "concave" solution is the viscosity solution of the problem $F[u] = \det D^2u$. Also, this problem is degenerate for the class of functions that are both concave and convex. Results for approximating with $r = 0, 1, 2$ can be found in Tables 1, 2 and 3, respectively, where we observe optimal convergence rates. Plots for some of the various approximations can be found in Figs. 1 and 2.

We now demonstrate that for this test problem the numerical moment appears to assist with resolving the issue of uniqueness only in a restrictive function class. We approximate Example 1 using the numerical moment with $\alpha = -121$, $N_x = N_y = 24$, $r = 0$, and initial guess given by the zero function. The result is recorded in Fig. 3. Thus, we can see that for a negative semi-definite choice for α , we recover an approximation for the non-convex solution of the Monge–Ampère problem as recorded in [13].

Example 2 Consider the Monge–Ampère problem

$$\begin{aligned}
 -\det D^2u &= -u_{xx}u_{yy} + u_{xy}u_{yx} = 0 && \text{in } \Omega, \\
 u &= g && \text{on } \partial\Omega,
 \end{aligned}$$

where $\Omega = (-1, 1) \times (-1, 1)$ and g is chosen such that the viscosity solution is given by $u(x, y) = |x| \in H^1(\Omega)$.

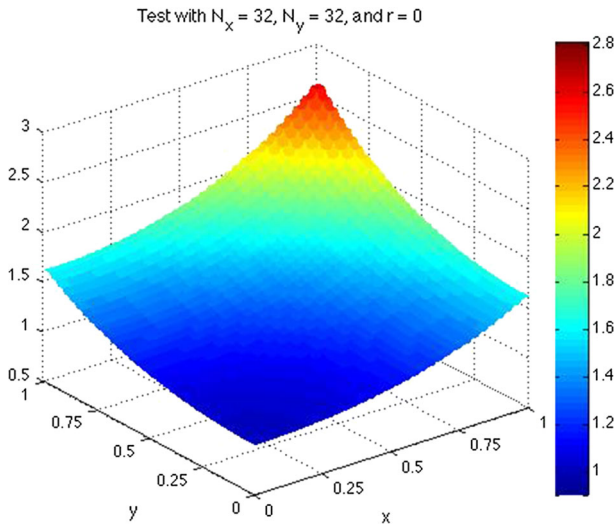


Fig. 1 Computed solution for Example 1 using $r = 0, \alpha = 24I, h = 4.419e-02,$ and *fsolve* with initial guess $u_h^{(0)} = 0$

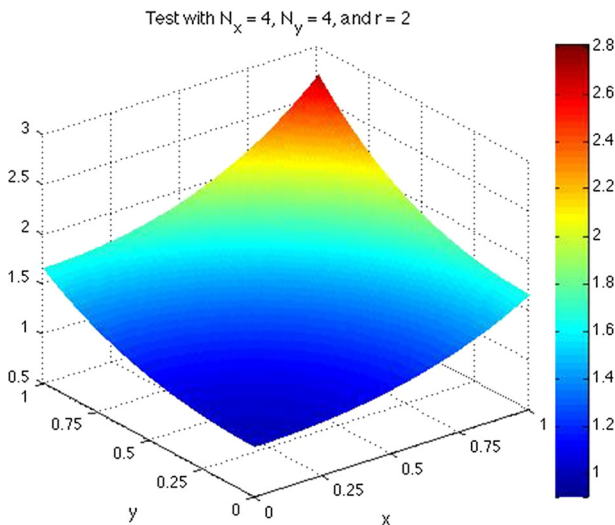


Fig. 2 Computed solution for Example 1 using $r = 2, \alpha = 24I, h = 3.536e-01,$ and *fsolve* with initial guess $u_h^{(0)} = 0$

Observe that the PDE is actually degenerate when acting on the solution u . Furthermore, due to the low regularity of u , we expect the rate of convergence to be bound by one. Using both piecewise constant and piecewise linear basis functions, we can see that the rate of convergence is bound by the theoretical bound in Tables 4 and 5. Plots for some of the approximations can be found in Fig. 4 for $r = 0$ and Fig. 5 for $r = 1$. We remark that for $r = 0$, all three solver approaches discussed in Sect. 4.5 gave analogous results. However,

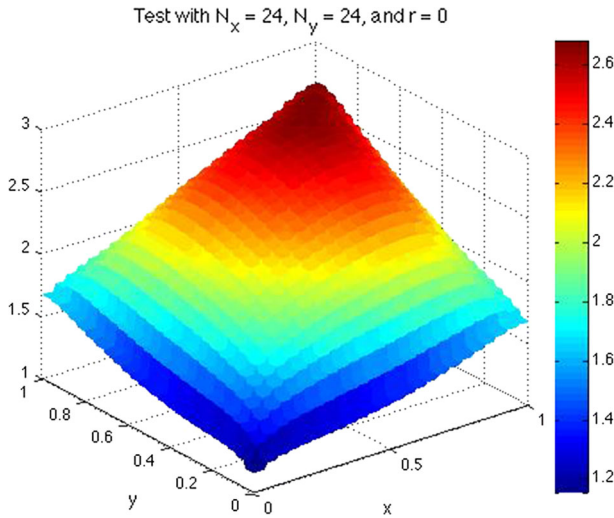


Fig. 3 Computed solution for Example 1 using $r = 0$, $\alpha = -121$, $h = 5.893e-02$, and *fsolve* with initial guess $u_h^{(0)} = 0$

Table 4 Rates of convergence for Example 2 using $r = 0$, $\alpha = I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h_x	L^∞ norm	Order	L^2 norm	Order
1.33e-01	1.87e-01		1.70e-01	
8.00e-02	1.30e-01	0.71	1.22e-01	0.65
5.71e-02	1.02e-01	0.72	9.77e-02	0.66
4.44e-02	8.51e-02	0.74	8.23e-02	0.68
3.64e-02	7.33e-02	0.74	7.16e-02	0.69

Table 5 Rates of convergence for Example 2 using $r = 1$, $\alpha = I$, $h_y = 1/3$ fixed, and Algorithm 1 with initial guess $u_h^{(0)} = 0$

h_x	L^∞ norm	Order	L^2 norm	Order
2.50e-01	3.86e-02		3.42e-02	
1.25e-01	2.08e-02	0.89	1.85e-02	0.88
8.33e-02	1.38e-02	1.02	1.24e-02	0.99

for $r = 1$, the direct formulation appears to have small residual wells that can trap the solver. Thus, for this test, the non-Newton solver given by Algorithm 1 appears to be better suited.

Another benefit of the numerical moment is that it can help regularize a problem that may not be well-conditioned for a Newton solver due to a singular or poorly scaled Jacobian. Note that $\frac{\partial F}{\partial D^2 u} = 0$ almost everywhere in Ω for the viscosity solution u due to the fact that $D^2 u(x, y) = 0$ for all $x \neq 0$. This leads to a singular or badly scaled matrix when using a Newton algorithm to solve the problem without the presence of a numerical moment. By adding a numerical moment, the resulting system of equations may be better suited for Newton algorithms since $\frac{\partial \widehat{F}}{\partial P_h^{\pm \mp}} = \frac{\partial F}{\partial P_h^{\pm \mp}} - \alpha$ may be nonsingular even when $P_h^{\pm \mp} \approx 0$. For the next numerical test, we let $\alpha = \gamma \mathbf{1}$ for various positive values of γ to see how the numerical moment affects both the accuracy and the performance of the Newton solver

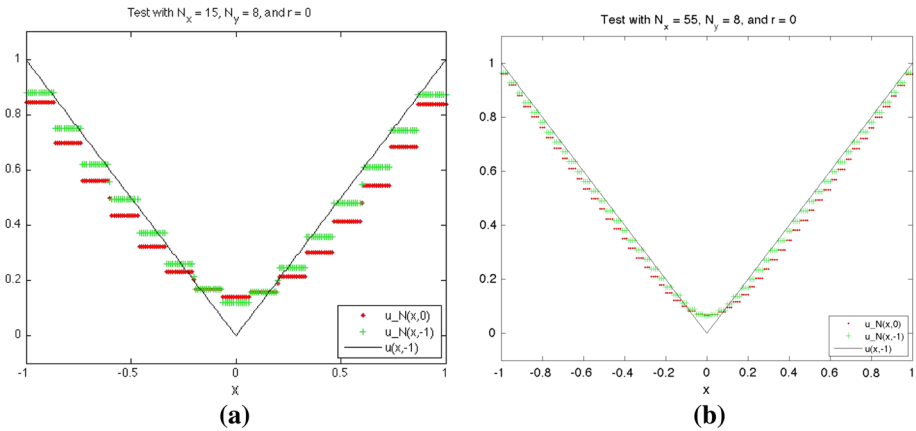


Fig. 4 Computed solutions for Example 2 using $r = 0$, $\alpha = I$, $h_y = 1.250e-01$, and *fsolve* with initial guess $u_h^{(0)} = 0$. **a** $h_x = 6.667e-02$. **b** $h_x = 1.818e-02$

fsolve. The choice for the numerical moment is especially interesting upon noting that α is in fact a singular matrix. However, with a numerical moment, the perturbation in $\frac{\partial \widehat{F}}{\partial P_h^{\pm \mp}}$ caused by $P_h^{\pm \mp}$ may be enough to eliminate the singularity since the approximation may now have some curvature. We let the initial guess be given by the zero function, fix the mesh $N_x = N_y = 20$, and let $r = 0$. We can see from Table 6 that for γ small, *fsolve* converges slowly, if at all. For $\gamma = 0$, *fsolve* does not converge within 100 iterations even for a very good initial guess. However, increasing γ does appear to aid *fsolve* in its ability to find a root with only a small penalty in the approximation error. For $r \geq 1$, we again note that Algorithm 1 provides a much better suited solver due to the degeneracy of the problem. However, the crux of Algorithm 1 reduces to a choice of $\gamma > 0$ with $\alpha = \gamma I$ instead of $\alpha = \gamma \mathbf{1}$. Similar results, as seen in Table 6, hold for $\alpha = \gamma I$.

Example 3 Consider the stationary Hamilton–Jacobi–Bellman problem

$$\begin{aligned} \min \{-\Delta u, -\Delta u/2\} &= f && \text{in } \Omega, \\ u &= g && \text{on } \partial\Omega, \end{aligned}$$

where $\Omega = (0, \pi) \times (-\pi/2, \pi/2)$,

$$f(x, y) = \begin{cases} 2 \cos(x) \sin(y), & \text{if } (x, y) \in S, \\ \cos(x) \sin(y), & \text{otherwise,} \end{cases}$$

$S = (0, \pi/2] \times (-\pi/2, 0] \cup (\pi/2, \pi] \times (0, \pi/2)$, and g is chosen such that the viscosity solution is given by $u(x, y) = \cos(x) \sin(y)$.

We can see that the optimal coefficient for Δu varies over four patches in the domain. Results for approximating with $r = 0, 1, 2$ can be seen in Tables 7, 8 and 9, respectively, where we observe optimal convergence rates for $r = 0, 1$ and near optimal convergence rates for $r = 2$. Plots for $r = 0$ and $r = 1$ can be found in Figs. 6 and 7.

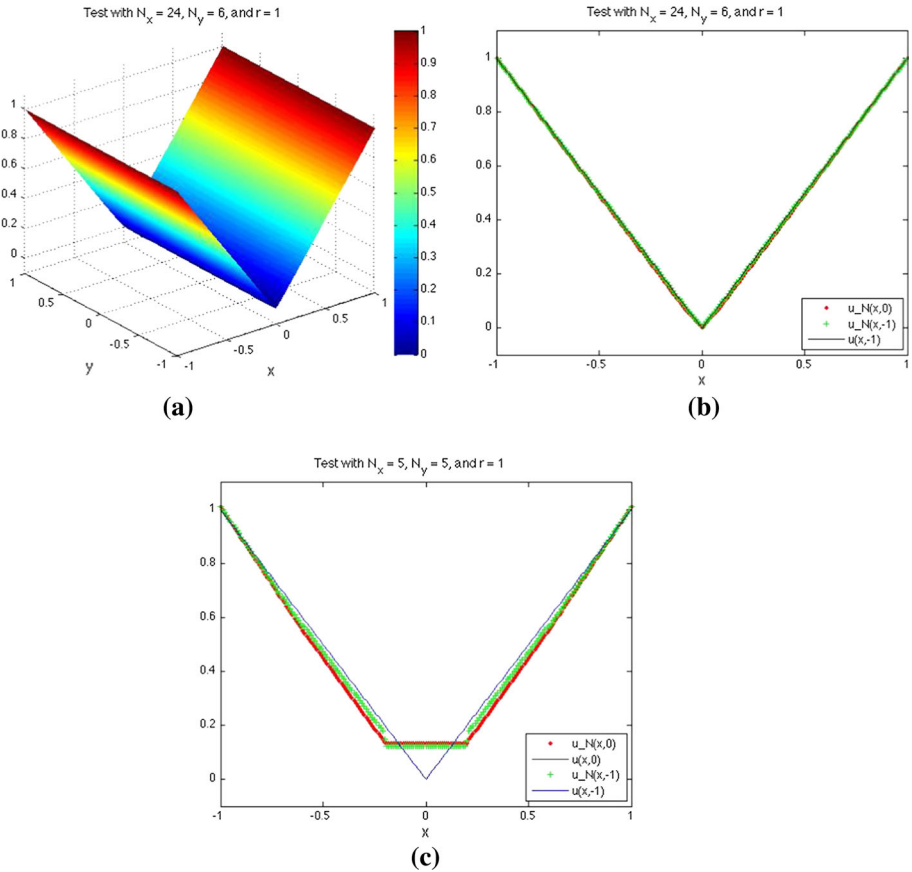


Fig. 5 Computed solution for Example 2 using $r = 1$, $\alpha = I$, and Algorithm 1 with initial guess $u_h^{(0)} = 0$. Note that the top plots correspond to $x = 0$ an edge and the bottom plot does not. **a** $h_x = 4.167e-02$ and $h_y = 1.667e-01$. **b** $h_x = 4.167e-02$ and $h_y = 1.667e-01$. **c** $h_x = 2.000e-01$ and $h_y = 2.000e-01$

Example 4 Consider the infinite-Laplacian problem

$$-\Delta_\infty u := -u_{xx} u_x u_y - u_{xy} u_x u_y - u_{yx} u_y u_y - u_{yy} u_y u_y = 0 \quad \text{in } \Omega, \\ u = g \quad \text{on } \partial\Omega,$$

where $\Omega = (-1, 1) \times (-1, 1)$ and g is chosen such that the viscosity solution is given by $u(x, y) = |x|^{4/3} - |y|^{4/3}$. While this problem is semilinear and not fully nonlinear, the solution has low regularity due to the fact $u \in C^{1, \frac{1}{3}}(\overline{\Omega}) \cap H^1(\Omega)$.

By approximation theory, we expect the error to be bound by $\mathcal{O}(h^1)$ independent of the degree of the polynomial basis. The approximation results for $r = 0, 1, 2$ can be found in Tables 10, 11 and 12, respectively. Plots for $r = 0$ and $r = 2$ can be found in Figs. 8 and 9. Note that while we observe the theoretical first order bound for the approximation error, we also observe that the higher order elements yield more accurate approximations.

Table 6 Approximation errors when varying $\alpha = \gamma \mathbf{1}$ for Example 2 using $r = 0$, $h = 7.071\text{e-}02$, and *fsolve* with initial guess $u_h^{(0)} = 0$

γ	L^∞ norm	L^2 norm	<i>fsolve</i> iterations
600	2.43e-01	2.43e-01	9
60	2.29e-01	2.27e-01	9
12	2.02e-01	1.98e-01	10
4	1.81e-01	1.74e-01	10
1	3.40e-01	2.08e-01	100
0*	2.84e-01	1.96e-01	100

The entry 0* corresponds to an initial guess given by the L^2 -projection of $u(x, y) = |x|$, $q_h^\pm(x, y) = \text{sgn } x$, and $P_h^{\mu\nu}(x, y) = 0$ for $\mu, \nu \in \{+, -\}$. The nonlinear solver *fsolve* is set to perform a maximum of 100 iterations

Table 7 Rates of convergence for Example 3 using $r = 0$, $\alpha = 2I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
5.55e-01	2.59e-01		2.73e-01	
3.70e-01	1.63e-01	1.14	1.75e-01	1.10
2.78e-01	1.17e-01	1.17	1.29e-01	1.06
1.85e-01	7.29e-02	1.16	8.48e-02	1.03
1.39e-01	5.27e-02	1.13	6.33e-02	1.02

Table 8 Rates of convergence for Example 3 using $r = 1$, $\alpha = 2I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
5.55e-01	4.89e-02		2.84e-02	
3.70e-01	2.23e-02	1.93	1.29e-02	1.94
2.78e-01	1.27e-02	1.97	7.38e-03	1.95

Table 9 Rates of convergence for Example 3 using $r = 2$, $\alpha = 2I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
2.22e+00	2.82e-01		1.25e-01	
7.40e-01	9.04e-03	3.13	9.52e-03	2.35
4.44e-01	2.39e-03	2.60	2.88e-03	2.34

Example 5 Consider the dynamic Hamilton–Jacobi–Bellman problem

$$\begin{aligned}
 u_t + \min \{-\Delta u, -\Delta u/2\} &= f && \text{in } \Omega \times (0, 1], \\
 u &= g && \text{on } \partial\Omega \times (0, 1], \\
 u &= u_0 && \text{in } \Omega \times \{0\},
 \end{aligned}$$

where $\Omega = (-1, 1) \times (-1, 1)$, $f(x, y, t) = s(x, y, t) + 2t(x|x| + y|y|)$,

$$s(x, y, t) = \begin{cases} 2t^2, & \text{if } x < 0 \text{ and } y < 0, \\ -4t^2, & \text{if } x > 0 \text{ and } y > 0, \\ 0, & \text{otherwise,} \end{cases}$$

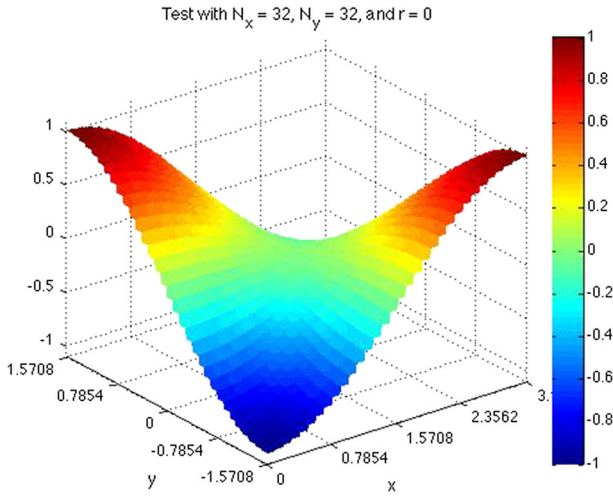


Fig. 6 Computed solution for Example 3 using $r = 0, \alpha = 2I, h = 1.388e-01,$ and *fsolve* with initial guess $u_h^{(0)} = 0$

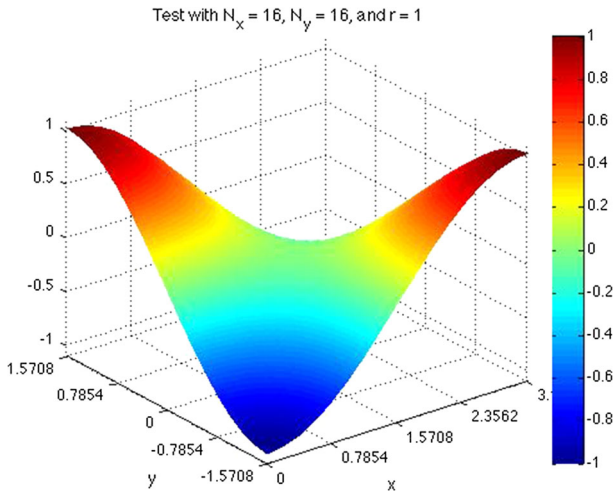


Fig. 7 Computed solution for Example 3 using $r = 1, \alpha = 2I, h = 2.777e-01,$ and *fsolve* with initial guess $u_h^{(0)} = 0$

and g and u_0 are chosen such that the viscosity solution is given by $u(x, y, t) = t^2 x |x| + t y |y|$. Then, for all t , we have $u(\cdot, \cdot, t) \in H^2(\Omega)$.

We expect the spatial rate of convergence to be bound by 2. However, due to the low order time discretization scheme, we can see that our error is dominated by the time discretization for $r \geq 1$. The spatial orders of convergence for $r = 0$ and $r = 1$ are recorded in Tables 13 and 14, respectively. For $r = 0$, the spatial discretization order matches the time discretization order, and we do observe an optimal rate of convergence. Using $r = 2$, we have the solution $u \in V^h$. Due to the high level of accuracy when using $r = 2$, we observe that the time

Table 10 Rates of convergence for Example 4 using $r = 0$, $\alpha = 60I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
2.83e-01	4.50e-01		3.37e-01	
1.41e-01	2.83e-01	0.67	2.02e-01	0.74
1.18e-01	2.46e-01	0.78	1.72e-01	0.88
9.43e-02	2.05e-01	0.82	1.40e-01	0.93

Table 11 Rates of convergence for Example 4 using $r = 1$, $\alpha = 60I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
4.71e-01	4.36e-02		3.17e-02	
2.83e-01	2.79e-02	0.88	1.81e-02	1.09
2.02e-01	2.20e-02	0.71	1.29e-02	1.02

Table 12 Rates of convergence for Example 4 using $r = 2$, $\alpha = 60I$, and *fsolve* with initial guess $u_h^{(0)} = 0$

h	L^∞ norm	Order	L^2 norm	Order
5.66e-01	2.41e-02		8.71e-03	
4.71e-01	1.48e-02	2.66	7.58e-03	0.76
3.54e-01	1.06e-02	1.16	4.64e-03	1.71

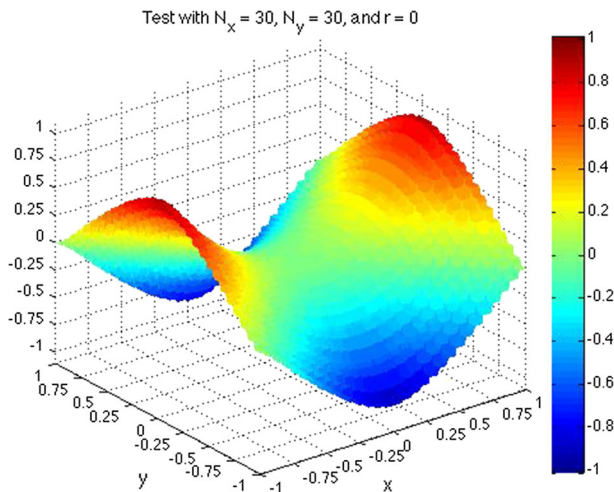


Fig. 8 Computed solution for Example 4 using $r = 0$, $\alpha = 60I$, $h = 9.428e-02$, and *fsolve* with initial guess $u_h^{(0)} = 0$

discretization order is in fact 1 as shown in Table 15. Plots for some of the approximations can be found in Figs. 10, 11 and 12.

7 Conclusion

In this paper, we have formulated a framework for designing LDG methods that approximate the viscosity solution of fully nonlinear second order elliptic and parabolic PDEs in

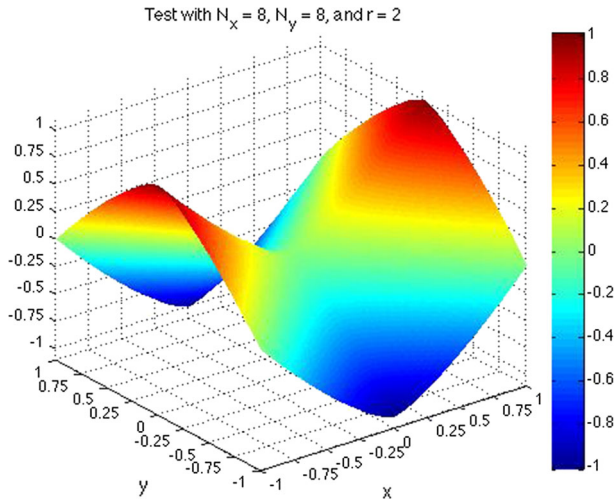


Fig. 9 Computed solution for Example 4 using $r = 2$, $\alpha = 60I$, $h = 3.536e-01$, and *fsolve* with initial guess $u_h^{(0)} = 0$

Table 13 Rates of convergence in space for Example 5 at time $t = 1$ using backward Euler time-stepping with $r = 0$, $\alpha = 2I$, $\Delta t = 0.1$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

h	L^∞ norm	Order	L^2 norm	Order
2.83e-01	5.62e-01		2.63e-01	
1.77e-01	3.62e-01	0.93	1.71e-01	0.92
1.41e-01	2.92e-01	0.96	1.38e-01	0.96

Table 14 Rates of convergence in space for Example 5 at time $t = 1$ using backward Euler time-stepping with $r = 1$, $\alpha = 2I$, $\Delta t = 0.1$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

h	L^∞ norm	order	L^2 norm	order
4.71e-01	7.41e-02		5.00e-02	
3.54e-01	4.21e-02	1.96	3.56e-02	1.18
2.83e-01	3.10e-02	1.38	2.76e-02	1.14

Table 15 Rates of convergence in time for Example 5 at time $t = 1$ using backward Euler time-stepping with $r = 2$, $\alpha = 2I$, $h = 1.414$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

Δt	L^∞ norm	Order	L^2 norm	Order
5.00e-01	4.12e-02		4.12e-02	
2.50e-01	2.11e-02	0.96	2.11e-02	0.97
1.00e-01	8.55e-03	0.99	8.49e-03	0.99
5.00e-02	4.29e-03	1.00	4.25e-03	1.00

high dimensions. We then focused on a particular LDG method within the framework that corresponded to the Lax–Friedrichs-like numerical operator. The key tools in designing the numerical operator are the introduction of a numerical viscosity and numerical moment. Through numerical tests, we observed the potential for the given framework that was originally motivated by successful numerical techniques for Hamilton–Jacobi equations as well a finite difference framework that abstracts the indirect techniques of the vanishing moment

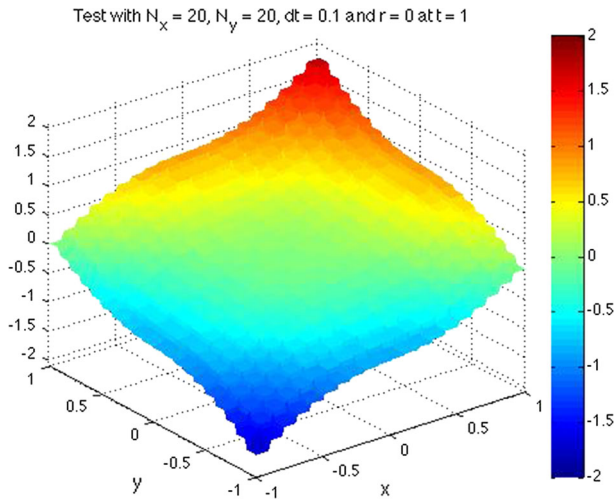


Fig. 10 Computed solution at time $t = 1$ for Example 5 using backward Euler time-stepping with $r = 0$, $\alpha = 2I$, $h = 1.414e-01$, $\Delta t = 0.1$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

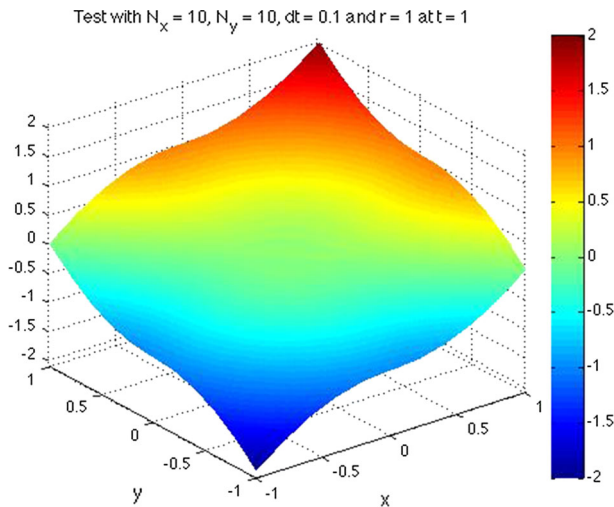


Fig. 11 Computed solution at time $t = 1$ for Example 5 using backward Euler time-stepping with $r = 1$, $\alpha = 2I$, $h = 2.828e-01$, $\Delta t = 0.1$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

method. We also proposed a nonlinear solver that took advantage of the special structure of the proposed numerical operator.

One interesting direction of research is to explore the possibility for using adaptive (or locally defined) coefficients in numerical moment and numerical viscosity terms to improve the convergence rate when the PDE solution is not smooth. Another promising direction of research is using the numerical moment as a low-regularity indicator to design and implement adaptive methods.

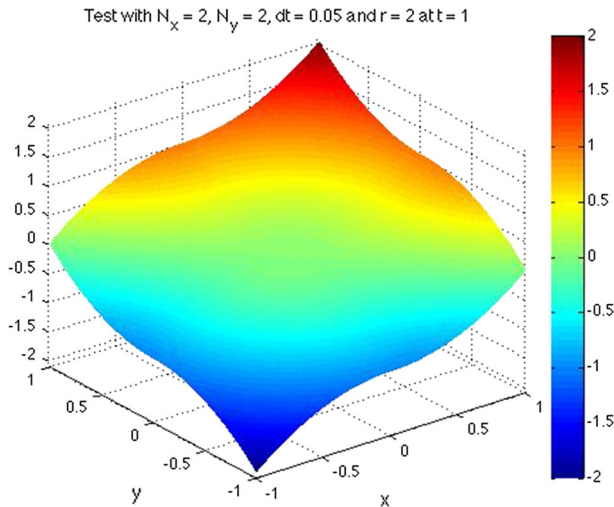


Fig. 12 Computed solution at time $t = 1$ for Example 5 using backward Euler time-stepping with $r = 2$, $\alpha = 2I$, $h = 1.414$, $\Delta t = 0.05$, and *fsolve* with initial guess $u_h^0 = \mathcal{P}_h u_0$

References

1. Barles, G., Souganidis, P.E.: Convergence of approximation schemes for fully nonlinear second order equations. *Asymptot. Anal.* **4**(3), 271–283 (1991)
2. Caffarelli, L.A., Cabré, X.: *Fully Nonlinear Elliptic Equations*, Vol. 43 of American Mathematical Society Colloquium Publications. American Mathematical Society, Providence (1995)
3. Crandall, M.G., Lions, P.-L.: Viscosity solutions of Hamilton–Jacobi equations. *Trans. Am. Math. Soc.* **277**(1), 1–42 (1983)
4. Debrabant, K., Jakobsen, E.: Semi-Lagrangian schemes for linear and fully non-linear diffusion equations. *Math. Comput.* **82**, 1433–1462 (2013)
5. Feng, X., Glowinski, R., Neilan, M.: Recent developments in numerical methods for second order fully nonlinear partial differential equations. *SIAM Rev.* **55**(2), 205–267 (2013)
6. Feng, X., Jensen, M.: Convergent semi-Lagrangian methods for the Monge–Ampère equation on unstructured grids. *SIAM J. Numer. Anal.* **55**, 691–712 (2017)
7. Feng, X., Kao, C., Lewis, T.: Convergent finite difference methods for one-dimensional fully nonlinear second order partial differential equations. *J. Comput. Appl. Math.* **254**, 81–98 (2013)
8. Feng, X., Lewis, T.: A narrow-stencil finite difference method for approximating viscosity solutions of fully nonlinear elliptic partial differential equations with applications to Hamilton–Jacobi–Bellman equations (2018)
9. Feng, X., Lewis, T.: Mixed interior penalty discontinuous Galerkin methods for one-dimensional fully nonlinear second order elliptic and parabolic equations. *J. Comput. Math.* **32**(2), 107–135 (2014)
10. Feng, X., Lewis, T.: Local discontinuous Galerkin methods for one-dimensional second order fully nonlinear elliptic and parabolic equations. *J. Sci. Comput.* **59**, 129–157 (2014)
11. Feng, X., Lewis, T.: Mixed interior penalty discontinuous Galerkin methods for fully nonlinear second order elliptic and parabolic equations in high dimensions. *Numer. Methods Partial Differ. Equ.* **30**(5), 1538–1557 (2014)
12. Feng, X., Lewis, T., Neilan, M.: Discontinuous Galerkin finite element differential calculus and applications to numerical solutions of linear and nonlinear partial differential equations. *J. Comput. Appl. Math.* **299**, 68–91 (2016)
13. Feng, X., Neilan, M.: Vanishing moment method and moment solutions for fully nonlinear second order partial differential equations. *J. Sci. Comput.* **38**(1), 74–98 (2009)
14. Feng, X., Neilan, M.: *The vanishing moment method for fully nonlinear second order partial differential equations: formulation, theory, and numerical analysis* (2011). [arXiv:1109.1183v2](https://arxiv.org/abs/1109.1183v2)
15. Fleming, W.H., Rishel, R.W.: *Deterministic and Stochastic Optimal Control. Applications of Mathematics*, No. 1. Springer, Berlin (1975)

16. Fleming, W.H., Soner, H.M.: *Controlled Markov Processes and Viscosity Solutions*, Volume 25 of *Applications of Mathematics*. Springer, New York (1993)
17. Gilbarg, D., Trudinger, N.S.: *Elliptic Partial Differential Equations of Second Order*, *Classics in Mathematics*. Springer, Berlin (2001) (**reprint of the 1998 edition**)
18. Jensen, M., Smears, I.: On the convergence of finite element methods for Hamilton–Jacobi–Bellman equations. *SIAM J. Numer. Anal.* **51**, 137–162 (2013)
19. Lewis, T.L.: Finite difference and discontinuous Galerkin finite element methods for fully nonlinear second order partial differential equations. Ph.D. thesis, University of Tennessee (2013). http://trace.tennessee.edu/utk_graddiss/2446
20. Lewis, T., Neilan, M.: Convergence analysis of a symmetric dual-wind discontinuous Galerkin method. *J. Sci. Comput.* **59**(3), 602–625 (2014)
21. Lieberman, G.M.: *Second Order Parabolic Differential Equations*. World Scientific Publishing Co., Inc., River Edge (1996)
22. Neilan, M., Salgado, A.J., Zhang, W.: Numerical analysis of strongly nonlinear PDEs. *Acta Numer.* [arXiv:1610.07992](https://arxiv.org/abs/1610.07992) [math.NA] (2017) (**to appear**)
23. Nitsche, J.A.: Über ein Variationsprinzip zur Lösung von Dirichlet Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* **36**, 9–15 (1970/71)
24. Nochetto, R.H., Ntougakas, D., Zhang, W.: Two-scale method for the Monge–Ampère equation: convergence rates (2017). [arXiv:1706.06193](https://arxiv.org/abs/1706.06193) [math.NA]
25. Pogorelov, A.V.: *Monge–Ampère Equations of Elliptic Type*. P. Noordhoff Ltd., Groningen (1964)
26. Salgado, A.J., Zhang, W.: Finite element approximation of the Isaacs equation (2016). [arXiv:1512.09091v1](https://arxiv.org/abs/1512.09091v1) [math.NA]
27. Shu, C.-W.: High order numerical methods for time dependent Hamilton–Jacobi equations. In: *Mathematics and computation in imaging science and information processing*, Vol. 11 of *Lecture Notes Series Institute Mathematics Science Natural University Singapore*, pp. 47–91. World Sci. Publ., Hackensack (2007)
28. Smears, I., Süli, E.: Discontinuous Galerkin finite element approximation of Hamilton–Jacobi–Bellman equations with Cordes coefficients. *SIAM J. Numer. Anal.* **52**, 993–1016 (2014)
29. Yan, J., Osher, S.: A local discontinuous Galerkin method for directly solving Hamilton–Jacobi equations. *J. Comput. Phys.* **230**, 232–244 (2011)