



An explicit hybridizable discontinuous Galerkin method for the acoustic wave equation

M. Stanglmeier^a, N.C. Nguyen^{a,*}, J. Peraire^a, B. Cockburn^b

^a Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

^b School of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA

Received 9 April 2015; received in revised form 26 October 2015; accepted 4 December 2015

Available online 17 December 2015

Highlights

- An explicit hybridizable discontinuous Galerkin method is developed for the acoustic wave equation.
- It can yield optimal convergence rates for all the approximate variables.
- It has some superconvergence properties that can be exploited to improve the convergence rate of the solution.
- The method is extended to treat the wave equation with perfectly matched layers.

Abstract

We present an explicit hybridizable discontinuous Galerkin (HDG) method for numerically solving the acoustic wave equation. The method is fully explicit, high-order accurate in both space and time, and coincides with the classic discontinuous Galerkin (DG) method with upwinding fluxes for a particular choice of its stabilization function. This means that it has the same computational complexity as other explicit DG methods. However, just as its implicit version, it provides optimal convergence of order $k + 1$ for all the approximate variables including the gradient of the solution, and, when the time-stepping method is of order $k + 2$, it displays a superconvergence property which allow us, by means of local postprocessing, to obtain new improved approximations of the scalar field variables at any time levels for which an enhanced accuracy is required. In particular, the new approximations converge with order $k + 2$ in the L^2 -norm for $k \geq 1$. These properties do not hold for all numerical fluxes. Indeed, our results show that, when the HDG numerical flux is replaced by the Lax–Friedrichs flux, the above-mentioned superconvergence properties are lost, although some are recovered when the Lax–Friedrichs flux is used only in the interior of the domain. Finally, we extend the explicit HDG method to treat the wave equation with perfectly matched layers. We provide numerical examples to demonstrate the performance of the proposed method.

© 2015 Elsevier B.V. All rights reserved.

Keywords: Finite element method; Discontinuous Galerkin methods; Runge–Kutta methods; Superconvergence; Wave equation

* Corresponding author.

E-mail addresses: max.stanglmeier@gmail.com (M. Stanglmeier), cuongng@mit.edu (N.C. Nguyen), peraire@mit.edu (J. Peraire), cockburn@math.umn.edu (B. Cockburn).

1. Introduction

Discontinuous Galerkin (DG) methods have been used for the spatial discretization of wave propagation problems [1–19] due to their ability to handle complex geometries and inhomogeneous materials, provide high-order accurate solutions with very low dispersion errors, as well as perform h/p mesh adaptivity. DG methods are traditionally combined with explicit time integration schemes such as Runge–Kutta methods to numerically solve wave propagation problems [20,15,18]. Explicit DG methods are attractive because they only require the inverse of a block-diagonal mass matrix and the computation of a residual vector. However, it is well known that the time step of explicit DG methods is limited by the Courant–Friedrichs–Lewy (CFL) condition [21]. In order to avoid the CFL condition, DG methods can be combined with implicit time integration schemes, resulting in implicit DG methods. However, one of the main criticism of implicit DG methods is that they have too many globally coupled degrees of freedom [22,23]. The same criticism has been raised against DG methods for steady-state problems like, for example, all the DG methods used to solve diffusion problems considered in [24], which, moreover, provide sub-optimally convergent approximations for the gradient of the field variable.

The hybridizable DG (HDG) methods first introduced in [25] for elliptic problems seek to address some of the above criticism. Indeed, the HDG methods guarantee that only the degrees of freedom of the approximation of the scalar variable on the interelement boundaries are globally coupled, and that the approximate gradient attains optimal order of convergence for elliptic problems [26–28]. This motivated the further development of the HDG methods for a variety of partial differential equations: diffusion problems [29,30], convection–diffusion problems [31–34], incompressible flow [35–39], compressible flows [40–43], continuum mechanics [44,41,42,45], and, more recently, to time-dependent acoustic and elastic wave propagation in [46,47], the Helmholtz equation in [9,10,48], and the time-harmonic Maxwell’s equations [49,50]. In the setting of wave propagation problems, the HDG methods compare with other finite element methods favorably because they achieve optimal orders of convergence for both the scalar and gradient unknowns and display superconvergence properties [46,9,48,47].

Thus far, only implicit HDG methods [47,39,41,32–34] have been developed to solve time-dependent problems. Our goal in this paper is to introduce an explicit hybridizable discontinuous Galerkin (EHDG) method for numerically solving the acoustic wave equation and explore its convergence properties. It turns out that the EHDG method is an explicit DG method whose numerical fluxes coincide with the upwinding fluxes for a particular choice of the stabilization function. Hence, the method has the same computational cost as other explicit DG methods and provides optimal convergence of order $k + 1$ for all the approximate variables. However, when the time-stepping method is of order $k + 2$, it displays a superconvergence property that allows us, by means of local postprocessing, to obtain new improved approximation of the field variable at any time levels for which an enhanced accuracy might be required. Indeed, the new approximation converges with order $k + 2$ in the L^2 -norm for $k \geq 1$ in full agreement with the theoretical results of the semidiscrete version of the method obtained in [46]. These superconvergence properties, which are new even for the well-known DG method using upwinding fluxes, seem to strongly depend on the choice of the numerical flux. Indeed, we present numerical examples to demonstrate these features and show that, when we replace the HDG numerical fluxes by those of the Lax–Friedrichs method, the superconvergence property is lost, and that some of them are recovered when the HDG flux is used at the boundary. We also extend the method to treat the wave equation with perfectly matched layers. Let us emphasize that, although we solely focus on the acoustic wave equation, the extension of the proposed method to the elastic wave equations and the time-dependent Maxwell’s equations is straightforward.

The article is organized as follows. In Section 2, we describe the EHDG method and in Section 3, we present numerical results to assess the convergence and accuracy of the method. In Section 4, we extend the method to treat perfectly matched layers and present numerical results to demonstrate the effectiveness of the method. Finally, in Section 5, we provide some concluding remarks on future work.

2. Explicit HDG method

2.1. The acoustic wave equation

Let $\Omega \in \mathbb{R}^d$ be a bounded domain with Lipschitz continuous boundary $\partial\Omega$ and let $T > 0$ be the final time. We consider the following acoustic wave equation

$$\rho(\mathbf{x}) \frac{\partial^2 u}{\partial t^2} - \nabla \cdot (\kappa(\mathbf{x}) \nabla u) = f, \quad \text{in } \Omega \times (0, T]. \quad (1)$$

Here u is the scalar field variable, κ is the bulk modulus of the medium, ρ is the density of the medium, and f is a given source term.

We next introduce a new field variable $v = \frac{\partial u}{\partial t}$ and the gradient $\mathbf{q} = \kappa \nabla u$. We can write (1) into a system of first-order equations as

$$\begin{aligned} \frac{\partial \mathbf{q}}{\partial t} - \kappa \nabla v &= 0, & \text{in } \Omega \times (0, T], \\ \rho \frac{\partial v}{\partial t} - \nabla \cdot \mathbf{q} &= f, & \text{in } \Omega \times (0, T]. \end{aligned} \quad (2)$$

The exact solution (v, \mathbf{q}) satisfies the following initial conditions

$$\begin{aligned} v(\mathbf{x}, t = 0) &= v_0(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ \mathbf{q}(\mathbf{x}, t = 0) &= \mathbf{q}_0(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \end{aligned} \quad (3)$$

and a general Robin boundary condition

$$\mathbf{q} \cdot \mathbf{n} + \beta v = g, \quad \text{on } \partial \Omega \times (0, T]. \quad (4)$$

The coefficient β varies on the boundary $\partial \Omega$ and represents different types of boundary conditions. Specifically, the Neumann boundary condition corresponds to $\beta = 0$, the Dirichlet boundary condition to $\beta = \infty$, and the first-order absorbing boundary condition to $\beta = 1$.

2.2. Approximation spaces

Let \mathcal{T}_h be a collection of disjoint elements that partition Ω . We denote by $\partial \mathcal{T}_h$ the set $\{\partial K : K \in \mathcal{T}_h\}$. For an element K of the collection \mathcal{T}_h , $F = \partial K \cap \partial \Omega$ is the boundary face if the $d - 1$ Lebesgue measure of F is nonzero. For two elements K^+ and K^- of the collection \mathcal{T}_h , $F = \partial K^+ \cap \partial K^-$ is the interior face between K^+ and K^- if the $d - 1$ Lebesgue measure of F is nonzero. Let \mathcal{E}_h^o and \mathcal{E}_h^∂ denote the set of interior and boundary faces, respectively. We denote by \mathcal{E}_h the union of \mathcal{E}_h^o and \mathcal{E}_h^∂ .

Let $\mathcal{P}_k(D)$ denote the set of polynomials of degree at most k on a domain D . We are going to use the following discontinuous finite element spaces:

$$\begin{aligned} W_h &= \{w \in L^2(\Omega) : w|_K \in W(K), \forall K \in \mathcal{T}_h\}, \\ \mathbf{V}_h &= \{\mathbf{p} \in (L^2(\Omega))^d : \mathbf{p}|_K \in \mathbf{V}(K), \forall K \in \mathcal{T}_h\}, \end{aligned}$$

for $W(K) \equiv \mathcal{P}_k(K)$ and $\mathbf{V}(K) \equiv (\mathcal{P}_k(K))^d$. In addition, we introduce a traced finite element space

$$M_h = \{\mu \in L^2(\mathcal{E}_h) : \mu|_F \in \mathcal{P}_k(F), \forall F \in \mathcal{E}_h\}.$$

We also set $M_h(g_D) = \{\mu \in M_h : \mu = \mathbf{P}g_D \text{ on } \partial \Omega_D\}$, where \mathbf{P} denotes the L^2 -projection into the space $\{\mu|_{\partial \Omega} \forall \mu \in M_h\}$.

For functions \mathbf{w} and \mathbf{v} in $(L^2(D))^d$, we denote $(\mathbf{w}, \mathbf{v})_D = \int_D \mathbf{w} \cdot \mathbf{v}$. For functions w and v in $L^2(D)$, we denote $(w, v)_D = \int_D wv$ if D is a domain in \mathbb{R}^d and $\langle w, v \rangle_D = \int_D wv$ if D is a domain in \mathbb{R}^{d-1} . We finally introduce

$$(w, v)_{\mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} (w, v)_K, \quad \langle \mu, \eta \rangle_{\partial \mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} \langle \mu, \eta \rangle_{\partial K},$$

for w, v defined on \mathcal{T}_h and μ, η defined on $\partial \mathcal{T}_h$.

2.3. Semi-discrete formulation

The HDG method for the wave equation (2)–(4) seeks to define $(\mathbf{q}_h, v_h, \widehat{v}_h) \in \mathbf{V}_h \times W_h \times M_h$ as a solution of

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\partial \mathbf{q}_h}{\partial t}, \mathbf{r} \right)_{\mathcal{T}_h} + (v_h, \nabla \cdot \mathbf{r})_{\mathcal{T}_h} - \langle \widehat{v}_h, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial \mathcal{T}_h} &= 0, \\ \left(\rho \frac{\partial v_h}{\partial t}, w \right)_{\mathcal{T}_h} + (\mathbf{q}_h, \nabla w)_{\mathcal{T}_h} - \langle \widehat{\mathbf{q}}_h \cdot \mathbf{n}, w \rangle_{\partial \mathcal{T}_h} &= (f, w)_{\mathcal{T}_h}, \\ \langle \widehat{\mathbf{q}}_h \cdot \mathbf{n}, \mu \rangle_{\partial \mathcal{T}_h \setminus \partial \Omega} + \langle \widehat{\mathbf{q}}_h \cdot \mathbf{n} + \beta \widehat{v}_h - g, \mu \rangle_{\partial \Omega} &= 0, \end{aligned} \quad (5)$$

for all $(\mathbf{r}, w, \mu) \in \mathbf{V}_h \times W_h \times M_h$ and all $t \in (0, T]$, where the numerical flux is defined as

$$\widehat{\mathbf{q}}_h \cdot \mathbf{n} = \mathbf{q}_h \cdot \mathbf{n} - \tau(v_h - \widehat{v}_h), \quad \text{on } \partial\mathcal{T}_h. \tag{6}$$

In a recent paper [47], we use implicit time-stepping methods such as the diagonally implicit Runge–Kutta schemes to discretize the HDG method (5)–(6) in time. This results in a class of implicit HDG methods.

In this paper, we aim to introduce a new class of explicit HDG methods. To this end, we consider the following equivalent (when we take τ to be constant on each face) reformulation of the HDG method (5)–(6): find $(\mathbf{q}_h, v_h) \in \mathbf{V}_h \times W_h$ such that for all $K \in \mathcal{T}_h$,

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\partial \mathbf{q}_h}{\partial t}, \mathbf{r} \right)_K + (v_h, \nabla \cdot \mathbf{r})_K - \langle \widehat{v}_h, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial K} &= 0, \quad \forall \mathbf{r} \in \mathbf{V}(K), \\ \left(\rho \frac{\partial v_h}{\partial t}, w \right)_K + (\mathbf{q}_h, \nabla w)_K - \langle \widehat{\mathbf{q}}_h \cdot \mathbf{n}, w \rangle_{\partial K} &= (f, w)_K, \quad \forall w \in W(K), \end{aligned} \tag{7}$$

where, for any given face $F \in \partial K$,

$$\widehat{v}_h = \begin{cases} \frac{\tau^+ v_h^+ + \tau^- v_h^-}{\tau^+ + \tau^-} - \frac{1}{\tau^+ + \tau^-} (\mathbf{q}_h^+ \cdot \mathbf{n}^+ + \mathbf{q}_h^- \cdot \mathbf{n}^-), & \text{if } F \in \mathcal{E}_h^o, \\ \frac{\tau}{\tau + \beta} v_h + \frac{1}{\tau + \beta} (\mathbf{P}g - \mathbf{q}_h \cdot \mathbf{n}), & \text{if } F \in \partial\Omega, \end{cases} \tag{8}$$

and

$$\widehat{\mathbf{q}}_h \cdot \mathbf{n} = \mathbf{q}_h \cdot \mathbf{n} - \tau(v_h - \widehat{v}_h) \quad \text{on } \partial K. \tag{9}$$

Here $\mathbf{P}g$ denotes the L^2 projection of g onto the space M_h , and

$$v_h^\pm|_F = v_h|_{F \in \partial K^\pm}, \quad \text{and} \quad \mathbf{q}_h^\pm|_F = \mathbf{q}_h|_{F \in \partial K^\pm},$$

where K^+ and K^- are two elements sharing the face F . Hence, v_h^- and \mathbf{q}_h^- (respectively, v_h^+ and \mathbf{q}_h^+) are nothing but the value of v_h and \mathbf{q}_h on the face F from the element K^- (respectively, K^+). We can show that the system (7)–(9) is equivalent to the original formulation (5)–(6) by explicitly deriving the expression (8) from the last equation of (5); see [32,33] for the detailed derivation. While the original formulation (5)–(6) is useful for implicit time integration, the reformulation (7)–(9) is better suited to explicit time integration.

Let us now establish a relation between these numerical traces and the solution of a Riemann problem for the equation under consideration. Such a solution is the weak solution of the wave equation (2) on $\mathbb{R}^d \times (0, T)$ with initial condition

$$v(\mathbf{x}, t = 0) = v^\pm, \quad \mathbf{q}(\mathbf{x}, t = 0) = \mathbf{q}^\pm \quad \forall \mathbf{x} \cdot \mathbf{n}^\pm < 0.$$

Assuming that $\kappa(\mathbf{x}) = \kappa^\pm$ and $\rho(\mathbf{x}) = \rho^\pm$ for all $\mathbf{x} \cdot \mathbf{n}^\pm > 0$, the solution (\mathbf{q}, v) of the above Riemann problem is such that

$$\mathbf{q}(\mathbf{0}, t) \cdot \mathbf{n}^\pm = \mathbf{q}^\pm \cdot \mathbf{n}^\pm - \sqrt{\kappa^\pm \rho^\pm} (v^\pm - v(\mathbf{0}, t)), \quad \forall t > 0,$$

where the normal component of $\mathbf{q}(\mathbf{0}, t)$ and $v(\mathbf{0}, t)$ are well-defined, constant quantities for $t > 0$. We can now see that when we take the stabilization function as $\tau^\pm := \sqrt{\kappa^\pm \rho^\pm}$, the numerical traces of the HDG method are nothing but the well-known *upwinding* fluxes. In this case, we see that the stabilization function τ is nothing but the ratio of the bulk modulus κ to the sound speed $\sqrt{\kappa/\rho}$ of the medium. It is thus reasonable to take τ in this way although this is not the only choice that provides good convergence results.

2.4. L^2 stability and energy conservation

We can easily show that the semi-discrete formulation (5) satisfies the following energy identity

$$e_h(T) = \frac{1}{2}(\rho v_0, v_0)_{\mathcal{T}_h} + \frac{1}{2}(\kappa^{-1} \mathbf{q}_0, \mathbf{q}_0)_{\mathcal{T}_h} + \int_0^T ((f, v_h)_{\mathcal{T}_h} + \langle g, \widehat{v}_h \rangle_{\partial\Omega}) dt, \tag{10}$$

where

$$e_h(T) = \frac{1}{2}(\rho v_h(T), v_h(T))_{\mathcal{T}_h} + \frac{1}{2}(\kappa^{-1} \mathbf{q}_h(T), \mathbf{q}_h(T))_{\mathcal{T}_h} + \int_0^T (\langle \tau(u_h - \widehat{u}_h), (u_h - \widehat{u}_h) \rangle_{\partial \mathcal{T}_h} + \langle \beta \widehat{v}_h, \widehat{v}_h \rangle_{\partial \Omega}) dt. \quad (11)$$

To prove the identity (10), we choose $w = v_h$, $\mathbf{r} = \mathbf{q}_h$, $\mu = \widehat{v}_h$ and add the resulting equations in (5) up to obtain

$$\left(\rho \frac{\partial v_h}{\partial t}, v_h \right)_{\mathcal{T}_h} + \left(\frac{1}{\kappa} \frac{\partial \mathbf{q}_h}{\partial t}, \mathbf{q}_h \right)_{\mathcal{T}_h} + \langle \tau(u_h - \widehat{u}_h), (u_h - \widehat{u}_h) \rangle_{\partial \mathcal{T}_h} + \langle \beta \widehat{v}_h, \widehat{v}_h \rangle_{\partial \Omega} = (f, v_h)_{\mathcal{T}_h} + \langle g, \widehat{v}_h \rangle_{\partial \Omega}. \quad (12)$$

The desired result then follows from integrating this equation from 0 to T . The energy identity (10) implies that $\mathbf{q}_h = u_h = \widehat{v}_h = 0$ when the right-hand side is set to zero and the parameters ρ , κ , β and τ are positive. As a result, the HDG method (5) is stable and yield unique solutions at all time.

Furthermore, when $f = 0$ and $g = 0$, it follows from the above result that

$$\frac{\partial E_h(T)}{\partial t} = - \langle \tau(u_h(T) - \widehat{u}_h(T)), (u_h(T) - \widehat{u}_h(T)) \rangle_{\partial \mathcal{T}_h} < 0, \quad (13)$$

where

$$E_h(T) = \frac{1}{2}(\rho v_h(T), v_h(T))_{\mathcal{T}_h} + \frac{1}{2}(\kappa^{-1} \mathbf{q}_h(T), \mathbf{q}_h(T))_{\mathcal{T}_h} + \int_0^T \langle \beta \widehat{v}_h, \widehat{v}_h \rangle_{\partial \Omega} dt \quad (14)$$

is the L^2 energy of the PDE system (2) at time T . Therefore, the HDG method is dissipative since the L^2 energy decays with time. The rate of the energy decay is exactly equal to the jump term $\sum_{K \in \mathcal{T}_h} \int_{\partial K} \tau |u_h - \widehat{u}_h|^2 dx$.

2.5. The HDG method as a particular case of a DG method

Let us place the HDG method in the general framework of DG methods. This will show the strong link with those methods and will allow us to make comparisons with some of them. Note that the (two-dimensional) wave equation under consideration can be written as

$$\mathbf{B} \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{u}) = \mathbf{0}, \quad (15)$$

where

$$\mathbf{B} = \begin{pmatrix} \rho & 0 & 0 \\ 0 & 1/\kappa & 0 \\ 0 & 0 & 1/\kappa \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} v \\ q_x \\ q_y \end{pmatrix}, \quad \mathbf{F}(\mathbf{u}) = - \begin{pmatrix} q_x & q_y \\ v & 0 \\ 0 & v \end{pmatrix}. \quad (16)$$

We apply the standard DG space discretization [5] to this linear hyperbolic first-order system to arrive at the following semidiscrete scheme: Find $\mathbf{u}_h \in [\mathcal{P}_k(K)]^3$ such that

$$\left(\mathbf{B} \frac{\partial \mathbf{u}_h}{\partial t}, \mathbf{V} \right)_K - (\mathbf{F}(\mathbf{u}_h), \nabla \mathbf{V})_K + \langle \widehat{\mathbf{f}}, \mathbf{V} \rangle_{\partial K} = \mathbf{0}, \quad (17)$$

for all $\mathbf{V} \in [\mathcal{P}_k(K)]^3$ and all $K \in \mathcal{T}_h$. Here $\widehat{\mathbf{f}}$ is the numerical flux vector depending on \mathbf{u}_h , the normal vector \mathbf{n} , and the boundary conditions. Note that the weak formulation of our HDG method (7) has this very form.

For any interior face $F \in \mathcal{E}^o$, a general formula for the numerical flux is given by

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^-, \mathbf{n}) = \frac{1}{2} (\mathbf{F}(\mathbf{u}_h) + \mathbf{F}(\mathbf{u}_h^-)) \mathbf{n} + \frac{1}{2} \mathbf{S}(\mathbf{u}_h - \mathbf{u}_h^-), \quad (18)$$

where \mathbf{u}_h^- is the value of the DG solution \mathbf{u}_h from the neighboring element that shares the interior face and \mathbf{S} is the so-called stabilization matrix. Different choices of the stabilization matrix lead to different DG methods. For instance, a popular choice is the Lax–Friedrichs flux which corresponds to setting \mathbf{S} as:

$$\mathbf{S} := \mathbf{S}_{LF} \equiv |\lambda_{\max}| \mathbf{I}, \quad (19)$$

where \mathbf{I} is the identity matrix, and $|\lambda_{\max}| = \max\{1/\sqrt{\kappa\rho}, \sqrt{\kappa\rho}\}$. For the HDG method, we have the following choice of the stabilization matrix when $\tau = \tau^\pm$:

$$\mathbf{S} := \mathbf{S}_{HDG} \equiv \begin{pmatrix} \tau & 0 & 0 \\ 0 & \frac{n_x^2}{\tau} & \frac{n_x n_y}{\tau} \\ 0 & \frac{n_x n_y}{\tau} & \frac{n_y^2}{\tau} \end{pmatrix}. \tag{20}$$

We can now clearly see what we have emphasized already, namely, that our HDG method is a particular case of the DG method (17)–(18). It coincides with the DG method with upwinding fluxes for $\tau = \sqrt{\kappa\rho}$ [3].

The numerical flux $\widehat{\mathbf{f}}$ must also be defined on the boundary faces to account for the boundary conditions. There, it is usually defined by

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^b, \mathbf{n}) = \frac{1}{2} (\mathbf{F}(\mathbf{u}_h) + \mathbf{F}(\mathbf{u}_h^b)) \cdot \mathbf{n} + \frac{1}{2} \mathbf{S} (\mathbf{u}_h - \mathbf{u}_h^b), \tag{21}$$

where the vector \mathbf{u}_h^b depends on \mathbf{u}_h and the boundary condition. To impose the boundary condition $v = 0$ on $\partial\Omega$, we can take $\mathbf{u}_h^b = (0, \mathbf{q}_h)^T$. This results in the following form of the numerical flux:

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^b, \mathbf{n}) = \begin{pmatrix} \mathbf{q}_h \cdot \mathbf{n} + \frac{|\lambda_{\max}|}{2} v_h \\ \frac{1}{2} v_h \\ \frac{1}{2} v_h \end{pmatrix}, \tag{22}$$

for the Lax–Friedrichs scheme. For the HDG scheme, however, we do not follow the formula (21). Instead, the HDG method takes a choice consistent with the solution of the Riemann problem, namely,

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^b, \mathbf{n}) = \begin{pmatrix} -\widehat{\mathbf{q}}_h \cdot \mathbf{n} \\ -\widehat{v}_h n_x \\ -\widehat{v}_h n_y \end{pmatrix}. \tag{23}$$

So, to impose the boundary condition $v = 0$ on $\partial\Omega$, we take

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^b, \mathbf{n}) = \begin{pmatrix} -\mathbf{q}_h \cdot \mathbf{n} + \tau v_h \\ 0 \\ 0 \end{pmatrix}$$

and to impose the boundary condition $\mathbf{q} \cdot \mathbf{n} + \beta v = g$, we take

$$\widehat{\mathbf{f}}(\mathbf{u}_h, \mathbf{u}_h^b, \mathbf{n}) = \begin{pmatrix} -(\mathbf{P}g - \beta \widehat{v}_h) \\ -\widehat{v}_h n_x \\ -\widehat{v}_h n_y \end{pmatrix}, \quad \widehat{v}_h := \frac{\tau}{\tau + \beta} v_h + \frac{1}{\tau + \beta} (\mathbf{P}g - \mathbf{q}_h \cdot \mathbf{n}).$$

Of course, we can also use the HDG boundary flux (23) for the Lax–Friedrichs scheme instead of (22).

2.6. Temporal discretization by the SSPRK methods

We use the SSPRK(s, s) scheme [3,51] to integrate the semi-discrete HDG system in time. We first set $(\mathbf{q}_h^{n,0}, v_h^{n,0}, u_h^{n,0}) = (\mathbf{q}_h^n, v_h^n, u_h^n)$. For $i = 0, \dots, s - 1$, we compute

$$\widehat{v}_h^{n,i} = \begin{cases} \frac{\tau^+ v_h^{+,n,i} + \tau^- v_h^{-,n,i}}{\tau^+ + \tau^-} - \frac{1}{\tau^+ + \tau^-} (\mathbf{q}_h^{+,n,i} \cdot \mathbf{n}^+ + \mathbf{q}_h^{-,n,i} \cdot \mathbf{n}^-), & \text{if } F \in \mathcal{E}_h \setminus \partial\Omega, \\ \frac{\tau}{\tau + \beta} v_h^{n,i} + \frac{1}{\tau + \beta} (\mathbf{P}g^{n,i} - \mathbf{q}_h^{n,i} \cdot \mathbf{n}), & \text{if } F \in \partial\Omega, \end{cases} \tag{24}$$

and $\widehat{\mathbf{q}}_h^{n,i} \cdot \mathbf{n} = \mathbf{q}_h^{n,i} \cdot \mathbf{n} - \tau(v_h^{n,i} - \widehat{v}_h^{n,i})$ for all faces F of \mathcal{E}_h ; we then determine $(\mathbf{q}_h^{n,i+1}, v_h^{n,i+1}, u_h^{n,i+1}) \in V(K) \times W(K) \times W(K)$ as the solution of

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\mathbf{q}_h^{n,i+1} - \mathbf{q}_h^{n,i}}{\Delta t^n}, \mathbf{r}\right)_K + (v_h^{n,i}, \nabla \cdot \mathbf{r})_K - (\widehat{v}_h^{n,i}, \mathbf{r} \cdot \mathbf{n})_{\partial K} &= 0, \\ \left(\rho \frac{v_h^{n,i+1} - v_h^{n,i}}{\Delta t^n}, w\right)_K + (\mathbf{q}_h^{n,i}, \nabla w)_K - (\widehat{\mathbf{q}}_h^{n,i} \cdot \mathbf{n}, w)_{\partial K} &= (f^{n,i}, w)_K, \\ \left(\frac{u_h^{n,i+1} - u_h^{n,i}}{\Delta t^n}, z\right)_K - (v_h^{n,i}, z)_K &= 0, \end{aligned} \tag{25}$$

for all $(\mathbf{r}, w, z) \in V(K) \times W(K) \times W(K)$ and for all elements $K \in \mathcal{T}_h$. We finally set

$$(\mathbf{q}_h^{n+1}, v_h^{n+1}, u_h^{n+1}) = \sum_{i=0}^s \alpha_{s,i} (\mathbf{q}_h^{n,i}, v_h^{n,i}, u_h^{n,i}), \tag{26}$$

where the coefficients $\alpha_{s,i}$ are precisely those corresponding to the SSPRK scheme (s, s) [3,51], namely

$$\begin{aligned} \alpha_{1,0} &= 1, & \alpha_{s,i} &= \frac{1}{i} \alpha_{s-1,i-1}, & i &= 1, \dots, s-2, \\ \alpha_{s,s} &= \frac{1}{s!}, & \alpha_{s,s-1} &= 0, & \alpha_{s,0} &= 1 - \sum_{i=1}^{s-1} \alpha_{s,i}. \end{aligned} \tag{27}$$

The SSPRK (s, s) scheme has s stages and s orders of accuracy.

2.7. Temporal discretization by the ERK methods

We begin by reviewing the class of explicit Runge–Kutta (ERK) methods for computing the solution of the ODE

$$\frac{dy}{dt} = r(y). \tag{28}$$

The general s -stage ERK method for the above ODE is written in the form

$$\begin{bmatrix} y^{n,1} \\ y^{n,2} \\ \dots \\ y^{n,s-1} \\ y^{n,s} \end{bmatrix} = \begin{bmatrix} y^{n,0} \\ y^{n,0} \\ \dots \\ y^{n,0} \\ y^{n,0} \end{bmatrix} + \Delta t^n \begin{bmatrix} a_{21} & 0 & \dots & 0 & 0 \\ a_{31} & a_{32} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} & 0 \\ b_1 & b_2 & \dots & b_{s-1} & b_s \end{bmatrix} \begin{bmatrix} r(y^{n,0}) \\ r(y^{n,1}) \\ \dots \\ r(y^{n,s-2}) \\ r(y^{n,s-1}) \end{bmatrix}, \tag{29}$$

where $y^{n,0} \equiv y^n$, $y^{n,i} \equiv y(t^n + c_i \Delta t^n)$, $i = 1, \dots, s$, and $y^{n,s} \equiv y^{n+1}$. To specify a particular ERK method, one needs to provide the coefficients a_{ij} , b_i and c_i . These RK coefficients are usually arranged in the form

0	0	0	0	...	0	0
c_2	a_{21}	0	0	...	0	0
c_3	a_{31}	a_{32}	0	...	0	0
...
c_s	a_{s1}	a_{s2}	a_{s3}	...	$a_{s,s-1}$	0
	b_1	b_2	b_3	...	b_{s-1}	b_s

(30)

which is known as Butcher tableau. Let us introduce the following matrix

$$\begin{bmatrix} d_{11} & 0 & \dots & 0 & 0 \\ d_{21} & d_{22} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ d_{s-1,1} & d_{s-1,2} & \dots & d_{s-1,s-1} & 0 \\ d_{s1} & d_{s2} & \dots & d_{s,s-1} & d_{ss} \end{bmatrix} = \begin{bmatrix} a_{21} & 0 & \dots & 0 & 0 \\ a_{31} & a_{32} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} & 0 \\ b_1 & b_2 & \dots & b_{s-1} & b_s \end{bmatrix}^{-1}. \tag{31}$$

Then we can rewrite the ERK method (29) as

$$y^{n,i+1} = y^{n,0} - \sum_{j=0}^{i-1} \frac{d_{ij}}{d_{ii}} (y^{n,j} - y^{n,0}) + \frac{\Delta t^n}{d_{ii}} r(y^{n,i}), \quad i = 0, \dots, s - 1. \tag{32}$$

In this form, we see that the i th stage of the ERK method corresponds to a forward Euler step with a timestep size $\Delta t^{n,i} = \Delta t^n/d_{ii}$ and a starting point $\bar{y}^{n,i} = y^{n,0} - \sum_{j=0}^{i-1} (d_{ij}/d_{ii})(y^{n,j} - y^{n,0})$.

We are now ready to describe the ERK method for solving the HDG system (7)–(9). For the i th stage of the ERK method, we first compute

$$\begin{aligned} \bar{\mathbf{q}}_h^{n,i} &= \mathbf{q}_h^{n,0} - \sum_{j=1}^{i-1} \frac{d_{ij}}{d_{ii}} (\mathbf{q}_h^{n,j} - \mathbf{q}_h^{n,0}), \\ \bar{v}_h^{n,i} &= v_h^{n,0} - \sum_{j=1}^{i-1} \frac{d_{ij}}{d_{ii}} (v_h^{n,j} - v_h^{n,0}), \\ \bar{u}_h^{n,i} &= u_h^{n,0} - \sum_{j=1}^{i-1} \frac{d_{ij}}{d_{ii}} (u_h^{n,j} - u_h^{n,0}); \end{aligned} \tag{33}$$

we next evaluate

$$\widehat{v}_h^{n,i} = \begin{cases} \frac{\tau^+ v_h^{+,n,i} + \tau^- v_h^{-,n,i}}{\tau^+ + \tau^-} - \frac{1}{\tau^+ + \tau^-} (\mathbf{q}_h^{+,n,i} \cdot \mathbf{n}^+ + \mathbf{q}_h^{-,n,i} \cdot \mathbf{n}^-), & \text{if } F \in \mathcal{E}_h \setminus \partial\Omega, \\ \frac{\alpha\tau}{\alpha\tau + \beta} v_h^{n,i} + \frac{1}{\alpha\tau + \beta} (\mathbf{P}g^{n,i} - \alpha\mathbf{q}_h^{n,i} \cdot \mathbf{n}), & \text{if } F \in \partial\Omega, \end{cases} \tag{34}$$

and $\widehat{\mathbf{q}}_h^{n,i} \cdot \mathbf{n} = \mathbf{q}_h^{n,i} \cdot \mathbf{n} - \tau(v_h^{n,i} - \widehat{v}_h^{n,i})$ for all faces F of \mathcal{E}_h ; we finally determine $(\mathbf{q}_h^{n,i+1}, v_h^{n,i+1}, u_h^{n,i+1}) \in \mathbf{V}(K) \times W(K) \times W(K)$ as the solution of

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\mathbf{q}_h^{n,i+1} - \bar{\mathbf{q}}_h^{n,i}}{\Delta t^n/d_{ii}}, \mathbf{r} \right)_K + (v_h^{n,i}, \nabla \cdot \mathbf{r})_K - \langle \widehat{v}_h^{n,i}, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial K} &= 0, \\ \left(\rho \frac{v_h^{n,i+1} - \bar{v}_h^{n,i}}{(\Delta t^n/d_{ii})}, w \right)_K + (\mathbf{q}_h^{n,i}, \nabla w)_K - \langle \widehat{\mathbf{q}}_h^{n,i} \cdot \mathbf{n}, w \rangle_{\partial K} &= (f^{n,i}, w)_K, \\ \left(\frac{u_h^{n,i+1} - \bar{u}_h^{n,i}}{\Delta t^n/d_{ii}}, z \right)_K - (v_h^{n,i}, z)_K &= 0, \end{aligned} \tag{35}$$

for all $(\mathbf{r}, w, z) \in \mathbf{V}(K) \times W(K) \times W(K)$ and for all elements $K \in \mathcal{T}_h$. This completes the description of the ERK method for temporal discretization.

2.8. Local postprocessing

To postprocess the numerical solution for the new approximation $u_h^{n,*}$, we define $u_h^{n,*} \in \mathcal{P}_{k+1}(K)$ on every simplex $K \in \mathcal{T}_h$ to satisfy

$$\begin{aligned} (\nabla u_h^{n,*}, \nabla w)_K &= (\mathbf{q}_h^n, \nabla w)_K, \quad \forall w \in \mathcal{P}_{k+1}(K), \\ (u_h^{n,*}, 1)_K &= (u_h^n, 1)_K. \end{aligned} \tag{36}$$

To postprocess the numerical solution for the new velocity $v_h^{n,*}$, we first compute an approximation $\mathbf{p}_h^n \in \mathbf{V}(K)$ to the velocity gradient $\mathbf{p}(t^n) = \nabla v(t^n)$ by locally solving the below system

$$(\mathbf{p}_h^n, \mathbf{v})_K = - (v_h^n, \nabla \cdot \mathbf{v})_K + \langle \widehat{v}_h^n, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial K}, \quad \forall \mathbf{v} \in \mathbf{V}(K). \tag{37}$$

We then find $v_h^{n*} \in \mathcal{P}_{k+1}(K)$ such that

$$\begin{aligned} (\nabla v_h^{n*}, \nabla w)_K &= (\mathbf{p}_h^n, \nabla w)_K, \quad \forall w \in \mathcal{P}_{k+1}(K), \\ (v_h^{n*}, 1)_K &= (v_h^n, 1)_K. \end{aligned} \quad (38)$$

The local postprocessing requires us to compute the inverse of a local mass matrix whose size is the dimension of $\mathcal{P}_{k+1}(K)$.

It is important to point out that we can compute u_h^{n*} and v_h^{n*} at any time step without advancing in time. Hence, the local postprocessing can be performed whenever we need higher accuracy at particular time steps.

3. The effect of the choice of the numerical fluxes

In this section, we study the effect of the choice of the numerical fluxes in the convergence properties of explicit DG methods. We consider the wave equation (1) on a unit square $\Omega = (0, 1) \times (0, 1)$ with boundary condition $v = 0$ on $\partial\Omega$ and initial condition $u(x, y, t = 0) = 0$ and $v(x, y, t = 0) = \sin(\pi x) \sin(\pi y)$. For $\kappa = \rho = 1$ and $f = 0$ the problem has the following exact solution

$$u = \frac{1}{\sqrt{2\pi}} \sin(\pi x) \sin(\pi y) \sin(\sqrt{2\pi}t), \quad v = \sin(\pi x) \sin(\pi y) \cos(\sqrt{2\pi}t).$$

This solution represents the vibration of the square membrane under an initial velocity.

In the first set of numerical experiments, we test the convergence properties of the explicit HDG method with $\tau^\pm = \tau_{upw}^\pm := \sqrt{\rho^\pm \kappa^\pm}$, that is, of the well-known upwinding DG method. We then take the stabilization function as $\tau = 10 \tau_{upw}$ and as $\tau = 0.1 \tau_{upw}$ to see if this affects the performance of the method. Our experiments show optimal convergence for the approximations \mathbf{q}_h and v_h as well as superconvergence of the local averages u_h (which is reflected in the optimal convergence of the postprocessing u_h^*), and that the orders of convergence are not affected when we change the stabilization function, in full agreement with the theoretical results proved for the semidiscrete version of the HDG method in [46]. (The optimal convergence of the postprocessing v_h^* was not proved in [46] but has been verified once again here.) Note that although the explicit upwinding method has been known for a few decades, this is the first time we uncover the above-mentioned superconvergence.

In the second set of numerical experiments, we replace the HDG fluxes by the popular Lax–Friedrichs flux to see if the superconvergence properties still hold. Our experiments show that, although we still achieve optimal orders of convergence for the approximations \mathbf{q}_h and v_h , the superconvergence of the local average of u_h is actually lost. Our experiments show that using the HDG boundary flux (23) produces smaller errors and better convergence rates than using the flux (22).

3.1. The HDG flux

We start by setting $\tau = \tau_{upw} := \sqrt{\kappa \rho}$ so that the explicit HDG method under consideration becomes the classic DG method with upwinding fluxes; see, for example, [3].

We consider triangular meshes that are obtained by splitting a regular $n \times n$ Cartesian grid into a total of $2n^2$ triangles, giving uniform element sizes of $h = 1/n$. On these meshes, we consider polynomials of degree k to represent all the approximate variables using a high-order nodal basis [15] within each element. We use SSPRK($k + 1, k + 1$) and SSPRK($k + 2, k + 2$) for temporal discretization. The timestep size is set to $\Delta t = \frac{h}{2(2k+1)}$. Unless otherwise indicated, the final time is $T = 1$.

We present the L^2 -errors and associated orders of convergence for the numerical approximations obtained using the HDG-SSPRK($k + 1, k + 1$) scheme in Table 1 and the HDG-SSPRK($k + 2, k + 2$) scheme in Table 2. We observe in both cases that all of the numerical approximations converge with the optimal order $k + 1$ and that, for the HDG-SSPRK($k + 2, k + 2$) scheme, the postprocessed field variable, u_h^* , converges with order $k + 2$, in full agreement with the theoretical results in [46]; the theoretical justification of the observed order of convergence for v_h^* , $k + 2$, remains an open problem. We also see that the HDG-SSPRK($k + 1, k + 1$) scheme performs just as well as the HDG-SSPRK($k + 2, k + 2$) scheme (the errors in time seem to be much smaller than those in space for $T = 1$), except for u_h^* for $k = 2$. In such a case, it only converges with order $k + 1$ with the HDG-SSPRK($k + 1, k + 1$) scheme whereas

Table 1

Errors and orders of convergence for the HDG-SSPRK($k + 1, k + 1$) scheme with $\tau = \tau_{upw}$ on structured meshes.

Degree	Mesh	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
k	$1/h$	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	4.24e-3	–	1.01e-2	–	1.69e-2	–	2.30e-3	–	8.98e-3	–
	4	4.07e-4	3.38	1.07e-3	3.24	1.68e-3	3.33	1.23e-4	4.23	5.41e-4	4.05
	8	4.48e-5	3.19	1.27e-4	3.07	1.84e-4	3.19	7.23e-6	4.09	2.96e-5	4.19
	16	5.27e-6	3.09	1.60e-5	2.99	2.16e-5	3.09	5.67e-7	3.67	1.78e-6	4.06
	32	6.39e-7	3.04	2.02e-6	2.99	2.62e-6	3.04	5.82e-8	3.28	1.23e-7	3.85
3	2	5.75e-4	–	1.62e-3	–	2.66e-3	–	1.82e-4	–	1.33e-3	–
	4	3.12e-5	4.21	8.22e-5	4.30	1.38e-4	4.27	4.64e-6	5.30	3.58e-5	5.21
	8	1.78e-6	4.13	5.21e-6	3.98	7.74e-6	4.15	1.31e-7	5.15	1.03e-6	5.13
	16	1.06e-7	4.07	3.32e-7	3.97	4.56e-7	4.08	3.89e-9	5.07	3.06e-8	5.07
	32	6.46e-9	4.04	2.09e-8	3.99	2.77e-8	4.04	1.19e-10	5.03	9.12e-10	5.07
4	2	8.63e-5	–	2.29e-4	–	4.00e-4	–	2.23e-5	–	1.65e-4	–
	4	2.56e-6	5.07	6.46e-6	5.15	1.16e-5	5.11	3.42e-7	6.03	2.67e-6	5.95
	8	7.71e-8	5.06	1.98e-7	5.03	3.33e-7	5.12	5.19e-9	6.04	3.73e-8	6.16
	16	2.35e-9	5.03	6.13e-9	5.01	1.00e-8	5.05	7.95e-11	6.03	6.09e-10	5.94
	32	7.25e-11	5.02	1.92e-10	5.00	3.07e-10	5.03	1.23e-12	6.02	1.06e-11	5.84

Table 2

Errors and orders of convergence for the HDG-SSPRK($k + 2, k + 2$) scheme with $\tau = \tau_{upw}$ on structured meshes.

Degree	Mesh	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
k	$1/h$	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	4.13e-3	–	9.84e-3	–	1.65e-2	–	2.13e-3	–	8.64e-3	–
	4	4.01e-4	3.37	1.06e-3	3.22	1.65e-3	3.32	1.02e-4	4.38	5.19e-4	4.06
	8	4.44e-5	3.17	1.27e-4	3.06	1.83e-4	3.18	4.82e-6	4.40	2.80e-5	4.21
	16	5.24e-6	3.08	1.60e-5	2.99	2.15e-5	3.09	2.59e-7	4.22	1.61e-6	4.12
	32	6.36e-7	3.04	2.02e-6	2.99	2.61e-6	3.04	1.53e-8	4.08	9.81e-8	4.04
3	2	5.75e-4	–	1.62e-3	–	2.66e-3	–	1.82e-4	–	1.33e-3	–
	4	3.12e-5	4.21	8.22e-5	4.30	1.38e-4	4.27	4.63e-6	5.29	3.59e-5	5.21
	8	1.78e-6	4.13	5.21e-6	3.98	7.74e-6	4.15	1.31e-7	5.15	1.03e-6	5.13
	16	1.06e-7	4.07	3.32e-7	3.97	4.56e-7	4.08	3.88e-9	5.07	3.05e-8	5.07
	32	6.46e-9	4.04	2.09e-8	3.99	2.77e-8	4.04	1.19e-10	5.03	8.97e-10	5.09
4	2	8.63e-5	–	2.29e-4	–	4.00e-4	–	2.23e-5	–	1.65e-4	–
	4	2.56e-6	5.07	6.46e-6	5.15	1.16e-5	5.11	3.42e-7	6.03	2.67e-6	5.95
	8	7.71e-8	5.06	1.98e-7	5.03	3.33e-7	5.12	5.19e-9	6.04	3.73e-8	6.16
	16	2.35e-9	5.03	6.13e-9	5.01	1.00e-8	5.05	7.95e-11	6.03	6.09e-10	5.94
	32	7.25e-11	5.02	1.92e-10	5.00	3.07e-10	5.03	1.23e-12	6.02	1.06e-11	5.84

it converges with order $k + 2$ with the HDG-SSPRK($k + 2, k + 2$) scheme, as expected. In conclusion, by using the local postprocessing, we can increase the convergence rate by one order from $k + 1$ to $k + 2$ with little additional cost.

Next, we show the convergence properties of the explicit HDG method for $\tau = 10\tau_{upw}$ in Table 3 and $\tau = 0.1\tau_{upw}$ in Table 4, where $\tau_{unw} := \sqrt{\kappa} \rho$. Our numerical results show that the convergence rates are similar to those shown in Table 1 for $\tau = \tau_{upw}$. However, in both cases, we must use a smaller timestep size in order to ensure the stability of the explicit HDG method. In particular, the timestep size is now set to $\Delta t = \frac{0.1h}{2(2k+1)}$ for both $\tau = 10\tau_{upw}$ and $\tau = 0.1\tau_{upw}$. (Note that the explicit HDG method is unstable when we consider a larger timestep size than $\Delta t = \frac{0.1h}{2(2k+1)}$.) More generally, we observe from our numerical experiments that for $\tau = c\tau_{upw}$ the timestep size required for the explicit HDG method is $\Delta t = \min\{\frac{ch}{2(2k+1)}, \frac{h}{2c(2k+1)}\}$. This implies that the choice of $\tau = \tau_{upw}$ is optimal. Hence, the value of τ does not affect the convergence rate of the explicit HDG method, but it does have a significant impact on the required timestep size.

Table 3

Errors and orders of convergence for the HDG-SSPRK($k + 1, k + 1$) scheme with $\tau = 10\tau_{upw}$ on structured meshes.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	5.11e-3	–	2.99e-2	–	7.88e-2	–	5.94e-3	–	3.01e-2	–
	4	4.88e-4	3.39	2.11e-3	3.83	1.55e-2	2.34	5.73e-4	3.37	2.19e-3	3.78
	8	3.32e-5	3.88	1.45e-4	3.86	2.18e-3	2.83	3.79e-5	3.92	1.70e-4	3.69
	16	2.13e-6	3.96	1.04e-5	3.80	2.32e-4	3.23	1.95e-6	4.28	1.31e-5	3.69
	32	1.83e-7	3.54	7.21e-7	3.85	2.29e-5	3.34	9.51e-8	4.36	9.01e-7	3.86
3	2	8.73e-4	–	3.18e-3	–	1.71e-2	–	1.01e-3	–	3.22e-3	–
	4	3.43e-5	4.67	1.19e-4	4.73	1.52e-3	3.49	4.12e-5	4.62	1.32e-4	4.61
	8	1.04e-6	5.04	5.19e-6	4.52	9.59e-5	3.99	1.24e-6	5.05	6.22e-6	4.41
	16	3.01e-8	5.12	1.96e-7	4.73	4.87e-6	4.30	3.11e-8	5.32	2.38e-7	4.71
	32	1.16e-9	4.69	7.00e-9	4.81	2.43e-7	4.33	7.73e-10	5.33	8.19e-9	4.86
4	2	1.04e-4	–	2.96e-4	–	2.76e-3	–	1.24e-4	–	3.05e-4	–
	4	2.20e-6	5.56	9.20e-6	5.01	1.12e-4	4.62	2.46e-6	5.66	9.71e-6	4.97
	8	4.90e-8	5.49	2.11e-7	5.45	3.24e-6	5.12	3.56e-8	6.11	2.20e-7	5.46
	16	1.44e-9	5.09	4.40e-9	5.59	8.03e-8	5.33	4.43e-10	6.33	4.16e-9	5.73
	32	4.56e-11	4.98	9.05e-11	5.60	2.04e-9	5.30	5.67e-12	6.29	6.49e-11	6.00

Table 4

Errors and orders of convergence for the HDG-SSPRK($k + 1, k + 1$) scheme with $\tau = 0.1\tau_{upw}$ on structured meshes.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	2.96e-2	–	2.51e-2	–	1.39e-2	–	2.31e-3	–	1.53e-2	–
	4	3.92e-3	2.92	6.59e-3	1.93	1.23e-3	3.49	1.44e-4	4.00	7.55e-4	4.34
	8	4.56e-4	3.10	1.11e-3	2.57	1.26e-4	3.29	6.56e-6	4.45	2.74e-5	4.79
	16	5.30e-5	3.10	1.53e-4	2.86	1.59e-5	2.98	2.93e-7	4.48	9.06e-7	4.92
	32	6.33e-6	3.07	1.98e-5	2.95	2.04e-6	2.97	1.51e-8	4.28	3.30e-8	4.78
3	2	5.04e-3	–	5.54e-3	–	1.98e-3	–	2.01e-4	–	7.78e-4	–
	4	3.16e-4	4.00	6.38e-4	3.12	8.48e-5	4.54	4.47e-6	5.49	1.62e-5	5.58
	8	1.83e-5	4.11	4.86e-5	3.71	5.11e-6	4.05	1.16e-7	5.27	3.28e-7	5.63
	16	1.08e-6	4.09	3.24e-6	3.91	3.29e-7	3.96	3.43e-9	5.08	9.16e-9	5.16
	32	6.50e-8	4.05	2.08e-7	3.97	2.09e-8	3.98	1.06e-10	5.02	2.73e-10	5.07
4	2	7.05e-4	–	9.98e-4	–	2.77e-4	–	2.40e-5	–	7.95e-5	–
	4	2.15e-5	5.03	4.85e-5	4.36	8.17e-6	5.08	3.26e-7	6.20	9.50e-7	6.39
	8	6.27e-7	5.10	1.75e-6	4.80	2.62e-7	4.96	4.83e-9	6.08	1.33e-8	6.16
	16	1.86e-8	5.07	5.72e-8	4.93	8.35e-9	4.97	7.43e-11	6.02	1.96e-10	6.08
	32	5.65e-10	5.04	1.82e-9	4.97	2.63e-10	4.99	1.15e-12	6.01	3.43e-12	5.84

We now study the performance of the EHDG method on unstructured meshes. To this end, we consider an initial mesh shown in Fig. 1 and refine it by splitting each triangle into 4 smaller triangles to obtain its first refinement. The initial mesh has $h_{\min} = 0.2357$ and the l th refinement has $h_{\min} = 0.2357/2^\ell$. We show in Table 5 the results for the SSPRK($k + 1, k + 1$) and in Table 6 the results for the SSPRK($k + 2, k + 2$) scheme. We observe that the results are quite similar to those obtained on the structured meshes.

Furthermore, we consider a much longer final time $T = 10$ to study the performance of the HDG method for long time integration. To this end, we choose to keep the ratio h/k fixed to $1/32$. Fig. 2 depicts the error in the processed quantities as a function of time. For the case $(k, h) = (1, 1/32)$, the errors increase rapidly at the early time and grow quite slowly at the later time. In contrast, for the case $(k, h) = (4, 1/8)$, the errors decrease at the early time and maintain the same level of accuracy at the later time. The results clearly indicate that high-order approximations are significantly more effective than low-order approximations for high accuracy in long time. Therefore, both the

Table 5
Errors and orders of convergence for the HDG-SSPRK($k + 1, k + 1$) scheme $\tau = \tau_{upw}$ on unstructured meshes.

Degree	Mesh	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
k	h_{\min}	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
1	0.2357	1.41e-2	–	3.11e-2	–	3.45e-2	–	8.70e-3	–	3.33e-2	–
	0.1179	2.5e-3	2.50	7.00e-3	2.15	7.10e-3	2.28	1.30e-3	2.74	5.60e-3	2.57
	0.0589	5.06e-4	2.30	1.60e-3	2.13	1.60e-3	2.15	1.59e-4	3.03	9.17e-4	2.61
	0.0295	1.15e-4	2.14	3.91e-4	2.03	3.84e-4	2.06	1.71e-5	3.22	1.70e-4	2.44
2	0.2357	1.00e-3	–	2.90e-3	–	3.30e-3	–	4.36e-4	–	1.90e-3	–
	0.1179	1.17e-4	3.10	3.49e-4	3.05	4.12e-4	3.00	2.23e-5	4.29	1.19e-4	4.00
	0.0589	1.40e-5	3.06	4.38e-5	3.00	4.92e-5	3.07	1.19e-6	4.23	7.92e-6	3.90
	0.0295	1.72e-6	3.03	5.50e-6	2.99	6.01e-6	3.03	7.91e-8	3.91	5.23e-7	3.92
3	0.2357	1.63e-4	–	3.55e-4	–	4.85e-4	–	2.77e-5	–	1.01e-4	–
	0.1179	9.03e-6	4.17	2.44e-5	3.86	2.67e-5	4.18	7.30e-7	5.25	4.56e-6	4.48
	0.0589	5.35e-7	4.08	1.58e-6	3.95	1.61e-6	4.05	2.13e-8	5.10	1.70e-7	4.74
	0.0295	3.26e-8	4.04	9.92e-8	3.99	9.96e-8	4.02	6.52e-10	5.03	4.81e-9	5.14
4	0.2357	1.28e-5	–	3.64e-5	–	3.51e-5	–	1.65e-6	–	1.06e-5	–
	0.1179	4.69e-7	4.77	1.03e-6	5.15	9.87e-7	5.15	2.26e-8	6.19	3.50e-7	4.91
	0.0589	1.49e-8	4.98	3.13e-8	5.04	2.98e-8	5.05	3.55e-10	5.99	7.95e-9	5.46
	0.0295	4.66e-10	5.00	9.65e-10	5.02	9.05e-10	5.04	5.36e-12	6.05	2.09e-10	5.25

Table 6
Errors and orders of convergence for the HDG-SSPRK($k + 2, k + 2$) scheme with $\tau = \tau_{upw}$ on unstructured meshes.

Degree	Mesh	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
k	h_{\min}	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
1	0.2357	1.45e-2	–	2.60e-2	–	3.57e-2	–	9.20e-3	–	2.71e-2	–
	0.1179	2.60e-3	2.48	6.00e-3	2.12	7.30e-3	2.29	1.4e-3	2.72	3.90e-3	2.80
	0.0589	5.19e-4	2.32	1.50e-3	2.00	1.60e-3	2.19	1.89e-4	2.89	4.88e-4	3.00
	0.0295	1.16e-4	2.16	3.63e-4	2.05	3.88e-4	2.04	2.42e-5	2.97	6.15e-5	2.99
2	0.2357	1.00e-3	–	2.90e-3	–	3.30e-3	–	4.17e-4	–	1.90e-3	–
	0.1179	1.16e-4	3.10	3.49e-4	3.06	4.11e-4	3.01	2.04e-5	4.35	1.17e-4	4.02
	0.0589	1.40e-5	3.05	4.38e-5	2.99	4.91e-5	3.06	9.93e-7	4.36	7.85e-6	3.90
	0.0295	1.71e-6	3.03	5.50e-6	2.99	6.00e-6	3.03	5.53e-8	4.16	5.18e-7	3.92
3	0.2357	1.63e-4	–	3.55e-4	–	4.85e-4	–	2.77e-5	–	1.02e-4	–
	0.1179	9.03e-6	4.17	2.44e-5	3.86	2.67e-5	4.18	7.30e-7	5.25	4.56e-6	4.48
	0.0589	5.35e-7	4.08	1.58e-6	3.95	1.61e-6	4.05	2.13e-8	5.10	1.70e-7	4.74
	0.0295	3.26e-8	4.04	9.92e-8	3.99	9.96e-8	4.02	6.52e-10	5.03	4.81e-9	5.14
4	0.2357	1.28e-5	–	3.64e-5	–	3.51e-5	–	1.65e-6	–	1.06e-5	–
	0.1179	4.69e-7	4.77	1.03e-6	5.15	9.87e-7	5.15	2.26e-8	6.19	3.50e-7	4.91
	0.0589	1.49e-8	4.98	3.13e-8	5.04	2.98e-8	5.05	3.55e-10	5.99	7.95e-9	5.46
	0.0295	4.66e-10	5.00	9.65e-10	5.02	9.05e-10	5.04	5.36e-12	6.05	2.09e-10	5.25

dissipation and dispersion characteristics of the HDG method depend crucially on the polynomial degree used to represent the approximate solution.

3.2. The Lax–Friedrichs flux

We now compare the performance of our explicit HDG method to that of the Lax–Friedrichs DG method on this particular example. Here we consider structured meshes and a much longer final time $T = 10$ instead of $T = 1$. We present the errors and convergence rates for the HDG-SSPRK($k + 1, k + 1$) scheme in Table 7, the HDG-SSPRK($k + 2, k + 2$) scheme in Table 8, and for the Lax–Friedrichs DG-SSPRK($k + 2, k + 2$) scheme using the boundary flux (22) in Table 9 and the HDG boundary flux (23) in Table 10. We see that, as expected,

Table 7

Errors and orders of convergence for the HDG-SSPRK($k + 1, k + 1$) scheme with $\tau = \tau_{upw}$ on structured meshes. Here the final time is $T = 10$.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	3.69e-3	–	9.70e-2	–	2.34e-2	–	3.84e-3	–	9.73e-2	–
	4	3.40e-4	3.44	4.80e-3	4.34	3.05e-3	2.94	3.44e-4	3.48	4.81e-3	4.34
	8	2.45e-5	3.79	2.67e-4	4.17	3.49e-4	3.13	2.45e-5	3.81	2.63e-4	4.19
	16	2.27e-6	3.44	2.30e-5	3.54	4.22e-5	3.05	2.21e-6	3.47	2.17e-5	3.60
	32	2.58e-7	3.14	2.57e-6	3.16	5.20e-6	3.02	2.47e-7	3.16	2.36e-6	3.20
3	2	2.45e-4	–	3.85e-3	–	3.82e-3	–	2.76e-4	–	3.87e-3	–
	4	6.62e-6	5.21	3.88e-5	6.63	2.43e-4	3.98	7.71e-6	5.16	3.64e-5	6.73
	8	2.10e-7	4.98	1.85e-6	4.39	1.50e-5	4.02	2.30e-7	5.07	5.74e-7	5.99
	16	1.03e-8	4.35	1.36e-7	3.77	9.27e-7	4.01	8.54e-9	4.75	2.08e-8	4.79
	32	6.33e-10	4.02	9.20e-9	3.89	5.76e-8	4.01	4.11e-10	4.38	9.43e-10	4.46
4	2	2.41e-5	–	2.22e-4	–	5.47e-4	–	2.59e-5	–	1.24e-4	–
	4	6.18e-7	5.28	6.53e-6	5.08	1.73e-5	4.98	4.00e-7	6.01	1.37e-6	6.50
	8	1.97e-8	4.97	2.09e-7	4.97	5.36e-7	5.02	6.17e-9	6.02	2.21e-8	5.95
	16	6.41e-10	4.94	6.61e-9	4.98	1.66e-8	5.01	9.57e-11	6.01	3.5e-10	5.98
	32	2.06e-11	4.96	2.08e-10	4.99	5.16e-10	5.01	1.49e-12	6.01	5.56e-12	5.98

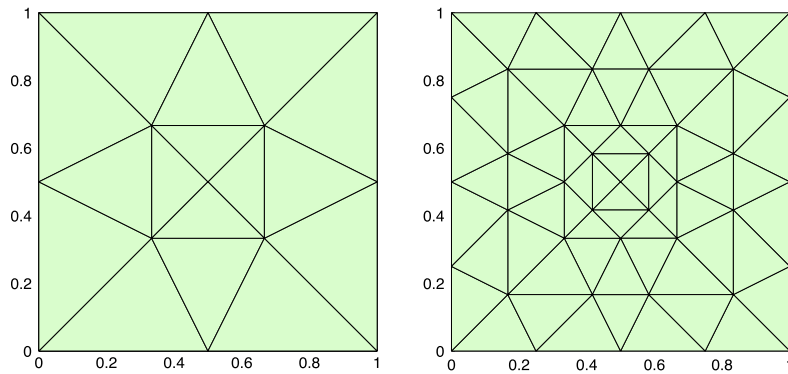


Fig. 1. The initial unstructured mesh (left) and its first refinement (right).

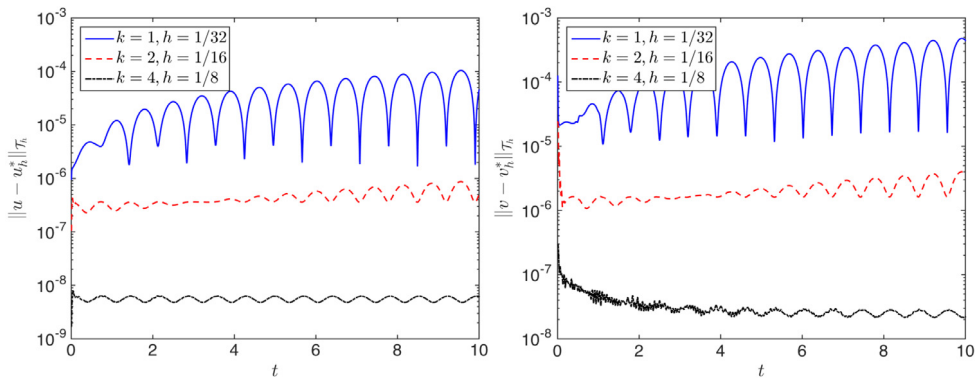


Fig. 2. The L^2 error in the postprocessed solution $\|u - u_h^*\|_{\mathcal{T}_h}$ (left) and the postprocessed velocity $\|v - v_h^*\|_{\mathcal{T}_h}$ (right) as a function of time for three choices of (k, h) such that $h/k = 1/32$.

the HDG-SSPRK($k + 1, k + 1$) does not yield $k + 2$ convergence rate for the postprocessed solutions, while the HDG-SSPRK($k + 2, k + 2$) does. Furthermore, it turns out that the definition of the numerical flux on the domain boundary has a significant impact on the accuracy of the numerical approximations and the postprocessed solutions. In particular,

Table 8

Errors and orders of convergence for the HDG-SSPRK($k + 2, k + 2$) scheme with $\tau = \tau_{upw}$ on structured meshes. Here the final time is $T = 10$.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	3.14e-3	–	8.90e-2	–	2.20e-2	–	3.31e-3	–	8.93e-2	–
	4	2.49e-4	3.66	3.61e-3	4.62	2.88e-3	2.93	2.55e-4	3.70	3.62e-3	4.62
	8	1.18e-5	4.40	1.27e-4	4.83	3.36e-4	3.10	1.17e-5	4.45	1.17e-4	4.95
	16	7.14e-7	4.04	8.39e-6	3.92	4.11e-5	3.03	5.06e-7	4.53	3.76e-6	4.96
	32	7.63e-8	3.23	1.02e-6	3.04	5.09e-6	3.01	2.49e-8	4.34	1.31e-7	4.85
3	2	2.28e-4	–	3.88e-3	–	3.80e-3	–	2.62e-4	–	3.91e-3	–
	4	5.83e-6	5.29	4.10e-5	6.57	2.42e-4	3.97	7.04e-6	5.22	3.88e-5	6.65
	8	1.75e-7	5.06	1.87e-6	4.45	1.50e-5	4.01	1.99e-7	5.15	6.29e-7	5.95
	16	8.28e-9	4.40	1.36e-7	3.78	9.27e-7	4.01	6.01e-9	5.05	1.84e-8	5.09
	32	5.16e-10	4.00	9.17e-9	3.89	5.75e-8	4.01	1.85e-10	5.02	5.8e-10	4.99
4	2	2.41e-5	–	2.22e-4	–	5.47e-4	–	2.59e-5	–	1.25e-4	–
	4	6.18e-7	5.28	6.53e-6	5.09	1.73e-5	4.98	4.00e-7	6.01	1.37e-6	6.51
	8	1.97e-8	4.97	2.09e-7	4.97	5.36e-7	5.02	6.17e-9	6.02	2.21e-8	5.96
	16	6.41e-10	4.94	6.61e-9	4.98	1.66e-8	5.01	9.56e-11	6.01	3.50e-10	5.98
	32	2.06e-11	4.96	2.08e-10	4.99	5.16e-10	5.01	1.49e-12	6.01	5.53e-12	5.98

the numerical results presented show that using the HDG boundary flux (23) produces smaller errors and better convergence rates than using the boundary flux (22). Note further that the Lax–Friedrichs DG-SSPRK($k + 2, k + 2$) scheme using the HDG boundary flux (23) provides similar results as the HDG-SSPRK($k + 2, k + 2$) scheme except that it does not yield the convergence rate of order $k + 2$ for the postprocessed velocity v_h^* , which appears to converge with order $k + 1$ only.

4. Extension to the wave equation with perfectly matched layers

4.1. Perfectly matched layers

We consider the PML formulation that was first introduced in [52] for the wave equation in its standard second-order form and subsequently extended in [53]. This PML formulation applied to the wave equation (1) yields the following system of equations:

$$\begin{aligned}
 \rho \frac{\partial^2 u}{\partial t^2} + \gamma \frac{\partial u}{\partial t} + \zeta u + \xi w - \nabla \cdot (\kappa \nabla u) - \nabla \cdot \mathbf{p} &= 0, \\
 \frac{\partial \mathbf{p}}{\partial t} + \mathbf{A} \mathbf{p} + \mathbf{B} (\kappa \nabla u) - \mathbf{C} \nabla w &= 0, \\
 \frac{\partial w}{\partial t} - u &= 0,
 \end{aligned}
 \tag{39}$$

where

$$\begin{aligned}
 \gamma &= \rho (\sigma_x + \sigma_y + \sigma_z), & \zeta &= \rho (\sigma_x \sigma_y + \sigma_x \sigma_z + \sigma_y \sigma_z), & \xi &= \rho (\sigma_x \sigma_y \sigma_z), \\
 \mathbf{A} &= \begin{bmatrix} \sigma_x & 0 & 0 \\ 0 & \sigma_y & 0 \\ 0 & 0 & \sigma_z \end{bmatrix}, & \mathbf{C} &= \begin{bmatrix} \sigma_y \sigma_z & 0 & 0 \\ 0 & \sigma_x \sigma_z & 0 \\ 0 & 0 & \sigma_x \sigma_y \end{bmatrix}, \\
 \mathbf{B} &= \begin{bmatrix} \sigma_x - \sigma_y - \sigma_z & 0 & 0 \\ 0 & \sigma_y - \sigma_x - \sigma_z & 0 \\ 0 & 0 & \sigma_z - \sigma_x - \sigma_y \end{bmatrix}.
 \end{aligned}
 \tag{40}$$

Here the damping functions $\sigma_x, \sigma_y, \sigma_z$ are positive inside the PML domain $\Omega_{\text{PML}} \equiv \Omega \setminus \Omega_{\text{FTD}}$ and vanish inside the finite truncated domain $\Omega_{\text{FTD}} \equiv (x_1, x_2) \times (y_1, y_2) \times (z_1, z_2)$, where $\Omega \equiv (x_1 - a, x_2 + a) \times (y_1 - b, y_2 + b) \times (z_1 -$

Table 9

Errors and orders of convergence for the Lax–Friedrichs DG–SSPRK($k + 2, k + 2$) scheme using the boundary flux (22) on structured meshes for $T = 10$.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	6.83e-2	–	7.29e-2	–	3.09e-1	–	6.83e-2	–	7.28e-2	–
	4	1.21e-2	2.50	1.03e-2	2.82	5.47e-2	2.50	1.21e-2	2.50	1.03e-2	2.82
	8	1.50e-3	3.01	2.68e-3	1.94	6.81e-3	3.01	1.50e-3	3.01	2.68e-3	1.95
	16	1.71e-4	3.14	4.50e-4	2.57	7.80e-4	3.12	1.71e-4	3.14	4.50e-4	2.57
	32	1.77e-5	3.27	6.57e-5	2.78	8.18e-5	3.25	1.77e-5	3.27	6.57e-5	2.78
3	2	1.37e-2	–	4.60e-2	–	6.31e-2	–	1.37e-2	–	4.61e-2	–
	4	4.54e-4	4.92	3.46e-3	3.73	2.12e-3	4.89	4.54e-4	4.92	3.46e-3	3.74
	8	4.12e-6	6.78	2.25e-4	3.94	3.27e-5	6.02	4.09e-6	6.79	2.25e-4	3.94
	16	7.20e-7	2.52	1.28e-5	4.14	3.49e-6	3.23	7.20e-7	2.51	1.28e-5	4.14
	32	7.14e-8	3.33	6.98e-7	4.20	3.39e-7	3.37	7.14e-8	3.33	6.98e-7	4.20
4	2	1.01e-3	–	4.77e-3	–	4.58e-3	–	1.01e-3	–	4.78e-3	–
	4	7.01e-5	3.85	3.15e-5	7.24	3.17e-4	3.85	7.01e-5	3.85	3.14e-5	7.25
	8	2.17e-6	5.01	3.74e-6	3.07	9.9e-6	5.00	2.17e-6	5.01	3.74e-6	3.07
	16	6.17e-8	5.14	1.42e-7	4.72	2.84e-7	5.12	6.17e-8	5.14	1.42e-7	4.72
	32	1.80e-9	5.10	4.37e-9	5.03	8.37e-9	5.08	1.80e-9	5.10	4.37e-9	5.03

Table 10

Errors and orders of convergence for the Lax–Friedrichs DG–SSPRK($k + 2, k + 2$) scheme using the HDG boundary flux (23) on structured meshes for $T = 10$.

Degree k	Mesh $1/h$	$\ u - u_h\ _{\mathcal{T}_h}$		$\ v - v_h\ _{\mathcal{T}_h}$		$\ q - q_h\ _{\mathcal{T}_h}$		$\ u - u_h^*\ _{\mathcal{T}_h}$		$\ v - v_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
2	2	3.88e-3	–	8.75e-2	–	2.39e-2	–	4.00e-3	–	8.78e-2	–
	4	3.25e-4	3.58	3.56e-3	4.62	2.89e-3	3.05	3.20e-4	3.65	3.55e-3	4.63
	8	1.74e-5	4.22	1.55e-4	4.53	3.12e-4	3.21	1.31e-5	4.60	1.38e-4	4.69
	16	1.62e-6	3.42	1.50e-5	3.37	3.88e-5	3.01	5.35e-7	4.62	1.19e-5	3.53
	32	1.90e-7	3.09	1.84e-6	3.03	4.98e-6	2.96	2.80e-8	4.25	1.44e-6	3.05
3	2	2.49e-4	–	3.86e-3	–	3.84e-3	–	2.72e-4	–	3.90e-3	–
	4	7.33e-6	5.09	4.13e-5	6.55	2.59e-4	3.89	6.41e-6	5.41	4.10e-5	6.57
	8	2.46e-7	4.89	2.11e-6	4.29	1.76e-5	3.89	1.79e-7	5.16	5.91e-7	6.12
	16	1.00e-8	4.62	1.69e-7	3.64	1.08e-6	4.02	5.51e-9	5.02	3.77e-8	3.97
	32	6.24e-10	4.01	1.19e-8	3.83	6.70e-8	4.01	1.72e-10	5.00	2.99e-9	3.66
4	2	2.45e-5	–	2.17e-4	–	5.52e-4	–	2.55e-5	–	1.25e-4	–
	4	5.83e-7	5.39	6.18e-6	5.13	1.73e-5	5.00	3.90e-7	6.03	2.60e-6	5.59
	8	1.67e-8	5.12	1.83e-7	5.08	5.41e-7	5.00	5.94e-9	6.04	9.53e-8	4.77
	16	5.02e-10	5.06	5.47e-9	5.06	1.75e-8	4.95	9.37e-11	5.99	3.14e-9	4.92
	32	1.54e-11	5.03	1.67e-10	5.04	5.30e-10	5.04	1.46e-12	6.00	1.00e-10	4.97

$c, z_2 + c$) is the computational domain, and a, b, c denote the thickness of the PML layer in the x, y, z directions, respectively. In particular, the damping function in the x -axis is defined as

$$\sigma_x(x) = \begin{cases} 0, & x_1 \leq x \leq x_2, \\ \frac{|x - x_1|^m}{a^m} \sigma_x^{\max}, & x_1 - a \leq x < x_1, \\ \frac{|x - x_2|^m}{a^m} \sigma_x^{\max}, & x_2 < x \leq x_2 + a. \end{cases} \tag{42}$$

Similar scaling functions are used for σ_y and σ_z . The constant σ_x^{\max} depends on the discretization and the thickness of the PML layer a . It has also been found that the optimal order for the polynomial scaling m is typically in the

range of $3 \leq m \leq 4$. On the exterior boundary of the PML domain Ω_{PML} we apply a first-order absorbing boundary condition.

In two space dimensions, σ_z and w vanish so that the above PML formulation reduces to

$$\begin{aligned} \rho \frac{\partial^2 u}{\partial t^2} + \rho (\sigma_x + \sigma_y) \frac{\partial u}{\partial t} + \rho \sigma_x \sigma_y u - \nabla \cdot (\kappa \nabla u) - \nabla \cdot \mathbf{p} &= 0 \\ \frac{\partial \mathbf{p}}{\partial t} + \mathbf{A} \mathbf{p} + \mathbf{B} \kappa \nabla u &= 0, \end{aligned} \tag{43}$$

with

$$\mathbf{A} = \begin{bmatrix} \sigma_x & 0 \\ 0 & \sigma_y \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \sigma_x - \sigma_y & 0 \\ 0 & \sigma_y - \sigma_x \end{bmatrix}. \tag{44}$$

By setting $v = \frac{\partial u}{\partial t}$ and $\mathbf{q} = \kappa \nabla u$, the second-order system can be reduced to a first-order system to be later solved with the Hybridizable Discontinuous Galerkin method. This results in the following PML formulation for the wave equation (1) in two space dimensions:

$$\begin{aligned} \rho \frac{\partial v}{\partial t} + \rho (\sigma_x + \sigma_y) v + \rho \sigma_x \sigma_y u - \nabla \cdot \mathbf{q} - \nabla \cdot \mathbf{p} &= 0, \\ \frac{1}{\kappa} \frac{\partial \mathbf{q}}{\partial t} - \nabla v &= 0, \\ \frac{\partial \mathbf{p}}{\partial t} + \mathbf{A} \mathbf{p} + \mathbf{B} \mathbf{q} &= 0, \\ \frac{\partial u}{\partial t} - v &= 0. \end{aligned} \tag{45}$$

It is important to point out that the last two equations are ordinary differential equations (ODEs). Hence, they can be solved locally in the HDG framework.

4.2. Explicit HDG method

For simplicity of exposition we consider the two-dimensional case. The explicit HDG method for the wave equation with PML (45) can be derived as shown in Section 2 for the wave equation without PML. In particular, the HDG method seeks to find $(\mathbf{q}_h, v_h, \hat{v}_h, \mathbf{p}_h, u_h) \in \mathbf{V}_h \times W_h \times M_h \times \mathbf{V}_h \times W_h$ such that

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\partial \mathbf{q}_h}{\partial t}, \mathbf{r} \right)_{\mathcal{T}_h} + (v_h, \nabla \cdot \mathbf{r})_{\mathcal{T}_h} - \langle \hat{v}_h, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial \mathcal{T}_h} &= 0, \\ \left(\rho \frac{\partial v_h}{\partial t}, w \right)_{\mathcal{T}_h} + (\rho ((\sigma_x + \sigma_y) v_h + \sigma_x \sigma_y u_h), w)_{\mathcal{T}_h} \\ + (\mathbf{q}_h, \nabla w)_{\mathcal{T}_h} - \langle \hat{\mathbf{q}}_h \cdot \mathbf{n}, w \rangle_{\partial \mathcal{T}_h} - (\nabla \mathbf{p}_h, w)_{\mathcal{T}_h} &= 0, \\ \langle \hat{\mathbf{q}}_h \cdot \mathbf{n}, \mu \rangle_{\partial \mathcal{T}_h \setminus \partial \Omega} + \langle \alpha \hat{\mathbf{q}}_h \cdot \mathbf{n} + \beta \hat{v}_h - g, \mu \rangle_{\partial \Omega} &= 0, \\ \left(\frac{\mathbf{p}_h}{\partial t}, \mathbf{s} \right)_{\mathcal{T}_h} + (\mathbf{A} \mathbf{p}_h, \mathbf{s})_{\mathcal{T}_h} + (\mathbf{B} \mathbf{q}_h, \mathbf{s})_{\mathcal{T}_h} &= 0, \\ \left(\frac{u_h}{\partial t}, z \right)_{\mathcal{T}_h} - (v_h, z)_{\mathcal{T}_h} &= 0, \end{aligned} \tag{46}$$

for all $(\mathbf{r}, w, \mu, \mathbf{s}, z) \in \mathbf{V}_h \times W_h \times M_h \times \mathbf{V}_h \times W_h$ and all $t \in (0, T]$, where the numerical flux is defined as

$$\hat{\mathbf{q}}_h \cdot \mathbf{n} = \mathbf{q}_h \cdot \mathbf{n} - \tau(v_h - \hat{v}_h), \quad \text{on } \partial \mathcal{T}_h. \tag{47}$$

We now need to discretize this system in time. Because each stage of the general ERK scheme and the SSPRK scheme is similar to the forward Euler method, we will describe the forward Euler method for temporal discretization.

Starting with $(\mathbf{q}_h^n, \mathbf{p}_h^n, v_h^n, u_h^n)$ as the numerical approximation to the solution $(\mathbf{q}(t^n), \mathbf{p}(t^n), v(t^n), u(t^n))$ at $t^n = n\Delta t^n$, we compute the numerical trace as

$$\hat{v}_h^n = \begin{cases} \frac{\tau^+ v_h^{+n} + \tau^- v_h^{-n}}{\tau^+ + \tau^-} - \frac{1}{\tau^+ + \tau^-} (\mathbf{q}_h^{+n} \cdot \mathbf{n}^+ + \mathbf{q}_h^{-n} \cdot \mathbf{n}^-), & \text{if } F \in \mathcal{E}_h \setminus \partial\Omega, \\ \frac{\alpha\tau}{\alpha\tau + \beta} v_h^n + \frac{1}{\alpha\tau + \beta} (\mathbf{P}g^n - \alpha\mathbf{q}_h^n \cdot \mathbf{n}), & \text{if } F \in \partial\Omega, \end{cases} \tag{48}$$

and the numerical flux as $\hat{\mathbf{q}}_h^n \cdot \mathbf{n} = \mathbf{q}_h^n \cdot \mathbf{n} - \tau(v_h^n - \hat{v}_h^n)$ for all faces F of \mathcal{E}_h . We then determine $(\mathbf{q}_h^{n+1}, \mathbf{p}_h^{n+1}, v_h^{n+1}, u_h^{n+1}) \in \mathbf{V}(K) \times \mathbf{V}(K) \times W(K) \times W(K)$ as the solution of

$$\begin{aligned} \left(\frac{1}{\kappa} \frac{\mathbf{q}_h^{n+1}}{\Delta t^n}, \mathbf{r} \right)_K &= \left(\rho \frac{\mathbf{q}_h^n}{\Delta t^n}, \mathbf{r} \right)_K + \langle \hat{v}_h^n, \mathbf{r} \cdot \mathbf{n} \rangle_{\partial K} - (v_h^n, \nabla \cdot \mathbf{r})_K, \\ \left(\rho \frac{v_h^{n+1}}{\Delta t^n}, w \right)_K &= \left(\frac{v_h^n}{\kappa \Delta t^n}, w \right)_K + \langle \hat{\mathbf{q}}_h^n \cdot \mathbf{n}, w \rangle_{\partial K} - (\mathbf{q}_h^n, \nabla w)_K \\ &\quad + (\nabla \cdot \mathbf{p}_h^n, w)_K - (\rho((\sigma_x + \sigma_y)v_h^n + \sigma_x \sigma_y u_h^n), w)_K, \\ \left(\frac{\mathbf{p}_h^{n+1}}{\Delta t^n}, \mathbf{s} \right)_K &= \left(\frac{\mathbf{p}_h^n}{\Delta t^n}, \mathbf{s} \right)_K - (\mathbf{A}\mathbf{p}_h^n, \mathbf{s})_K - (\mathbf{B}\mathbf{q}_h^n, \mathbf{s})_K, \\ \left(\frac{u_h^{n+1}}{\Delta t^n}, z \right)_K &= \left(\frac{u_h^n}{\Delta t^n}, z \right)_K + (v_h^n, z)_K, \end{aligned} \tag{49}$$

for all $(\mathbf{r}, w, \mathbf{s}, z) \in \mathbf{V}(K) \times W(K) \times \mathbf{V}(K) \times W(K)$ and for all elements $K \in \mathcal{T}_h$. Here the initial approximations $(\mathbf{q}_h^0, \mathbf{p}_h^0, v_h^0, u_h^0) \in \mathbf{V}(K) \times \mathbf{V}(K) \times W(K) \times W(K)$ are given by

$$\begin{aligned} (\mathbf{q}_h^0, \mathbf{r})_K &= (\mathbf{q}_0, \mathbf{r})_K, & (\mathbf{p}_h^0, \mathbf{s})_K &= (\mathbf{p}_0, \mathbf{s})_K, \\ (v_h^0, w)_{\mathcal{T}_h} &= (v_0, w)_K, & (u_h^0, z)_K &= (u_0, z)_K, \end{aligned} \tag{50}$$

for all $(\mathbf{r}, w, \mathbf{s}, z) \in \mathbf{V}(K) \times W(K) \times \mathbf{V}(K) \times W(K)$, where $(\mathbf{q}_0, v_0, \mathbf{p}_0, u_0)$ are the given initial data.

4.3. Scattering of plane wave from a circular cylinder

This example illustrates the performance of the explicit HDG method for curved geometry in an unbounded domain. We consider the acoustic scattering from the unit circular cylinder of an incident planar wave of the form

$$u^i(x, y, t) = \sin(\mathbf{k} \cdot \mathbf{x} - c|\mathbf{k}|t),$$

where c is the wave speed and $\mathbf{k} = (k_x, k_y)$ is the wave vector with $|\mathbf{k}| = \sqrt{k_x^2 + k_y^2}$. For our particular problem we consider $c = 1$ and $\mathbf{k} = (20, 0)$, which represents the plane wave along the x direction. The scattered wave u satisfies the homogeneous wave equation (1) ($f = 0, \rho = 1$, and $\kappa = 1$) and the following boundary condition on the cylinder surface $\partial\Omega_{\text{Cir}}$:

$$\nabla u \cdot \mathbf{n} = -\nabla u^i \cdot \mathbf{n}, \quad \text{on } \partial\Omega_{\text{Cir}}.$$

We study the scattered field in a finite truncated domain $\Omega_{\text{FTD}} \equiv (-5, 5) \times (-5, 5)$ and consider $a = b = c = 2$ for the PML layer thicknesses. The full computational domain is thus $\Omega \equiv \Omega_{\text{Square}} \setminus \Omega_{\text{Cir}}$, where $\Omega_{\text{Square}} \equiv (-7, 7) \times (-7, 7)$ and Ω_{Cir} is the unit circle. We use a first-order absorbing boundary condition on the outer boundary:

$$\frac{\partial u}{\partial t} + \nabla u \cdot \mathbf{n} = 0, \quad \text{on } \partial\Omega_{\text{Out}}.$$

The initial condition is $u(\mathbf{x}, t = 0) = 0$ and $\partial u(\mathbf{x}, t = 0)/\partial t = 0$. The computational domain and associated grid are shown in Fig. 3.

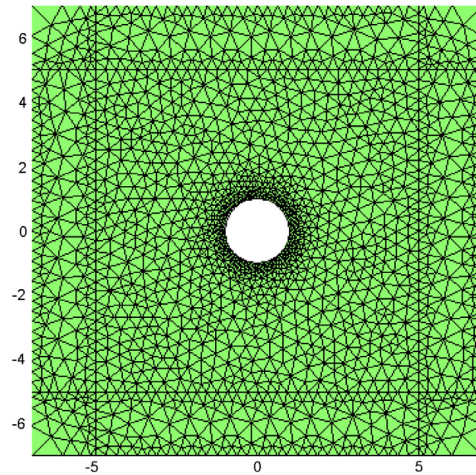


Fig. 3. The unstructured mesh for the problem of the scattering of plane waves.

The normalized analytical solution to this problem is given by

$$u(x, y, t) = \Re[e^{ic|k|t} U(x, y)], \quad (51)$$

where $U(x, y)$ is the solution in the frequency domain

$$U(x, y) = -\frac{J_0(k)}{H_0^{(1)}(k)} H_0^{(1)}(kr) - 2 \sum_{n=1}^{\infty} i^n \frac{J_n(k)}{H_n^{(1)}(k)} H_n^{(1)}(kr) \cos(n\theta). \quad (52)$$

Here J_n and $H_n^{(1)}$ denote the Bessel functions and the Hankel functions of the first kind, respectively; and (r, θ) represents the polar coordinates.

We consider solving this problem with the first-order absorbing boundary condition only and with the PML formulation. The former corresponds to setting $\sigma_x^{\max} = \sigma_y^{\max} = 0$ in the PML domain, while the latter corresponds to setting σ_x and σ_y according to (42) with $m = 4$ and $\sigma_x^{\max} = \sigma_y^{\max} = 15$. We display in Fig. 4 the approximate solution u_h at times $t = 5, 10$, and 20 for the cases with and without PML layers. Fig. 5 shows the difference between the solution plotted in Fig. 4 and the analytical solution (51). These results are obtained using polynomial degree $k = 3$ and the classical fourth-order Runge–Kutta scheme with $\Delta t = 0.00125$. We observe that the PML formulation is clearly superior to the first-order absorbing boundary condition and is barely distinguishable from the analytical solution in most of the truncated domain.

5. Conclusions

We have presented the first explicit HDG method for numerically solving the acoustics wave equation. The method yields the optimal convergence of order $k + 1$ for all the approximate variables including the gradient of the solution, while having the same computational complexity as other explicit DG methods. Moreover, it has some superconvergence properties that allow us to improve the convergence rate of the numerical solution by one order by means of a local postprocessing. Therefore, this explicit HDG method can provide more accurate solutions than existing DG methods for the same computational cost. Furthermore, we extended the method to treat the wave equation with perfectly matched layers. Numerical experiments were presented to demonstrate the optimal and super-convergence properties of the present method.

Let us emphasize that, for a particular choice of the stabilization function, the explicit HDG method is nothing but the well known DG method with upwinding fluxes. Although this method has been known for a few decades, the above-mentioned superconvergence property we uncover here was not known. There seems to be something special about the HDG numerical fluxes since, when they are replaced by the Lax–Friedrichs fluxes, these superconvergence properties are lost and, when the HDG flux is used on the boundary, some of these superconvergence properties are recovered. The theoretical understanding of this new phenomenon constitutes the subject of ongoing work.

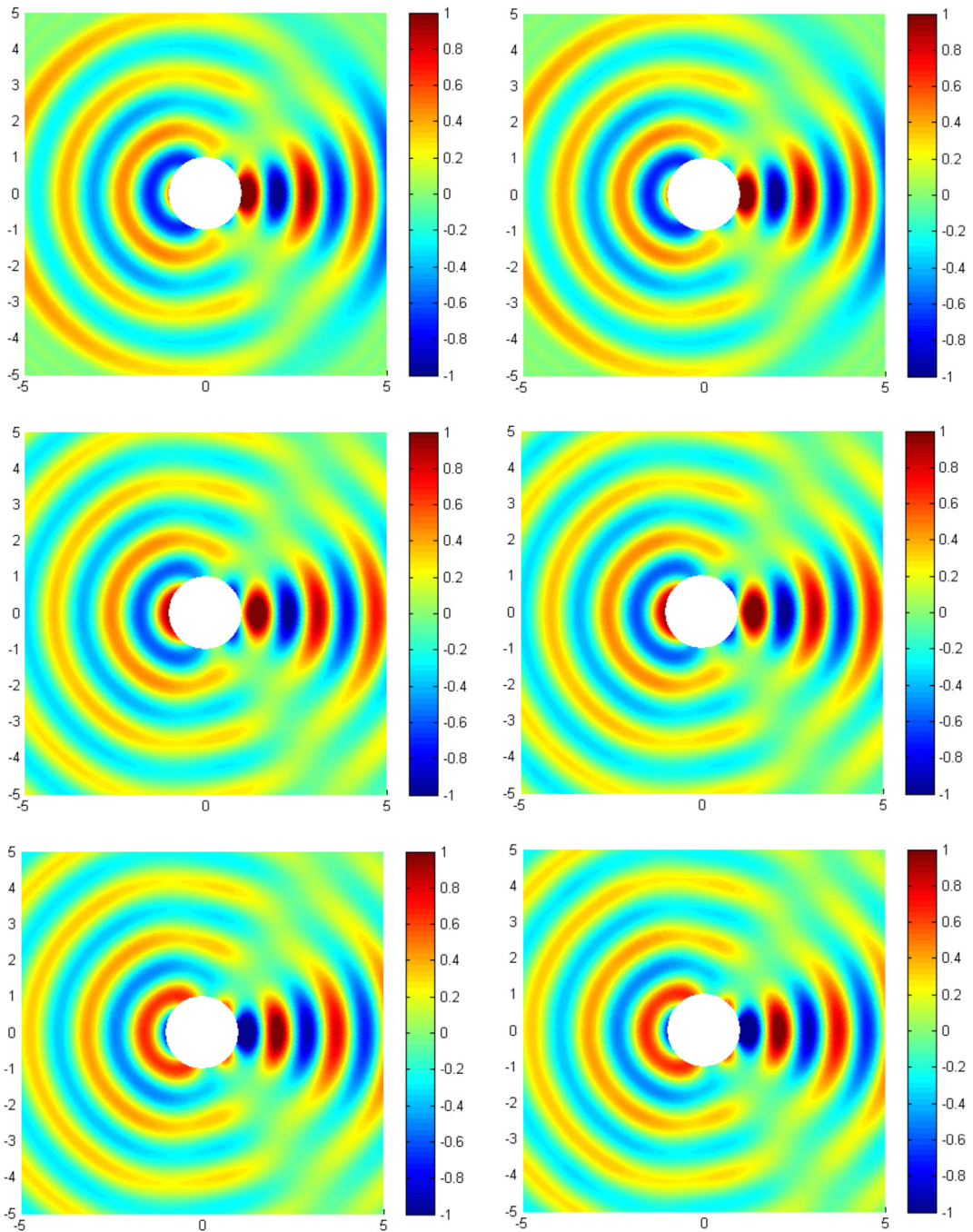


Fig. 4. Simulation of the wave scattering problem with a surrounding PML (left) and a first order ABC (right) using the explicit HDG method with a classical fourth order Runge–Kutta scheme ($\delta t = 0.00125$). Both solutions (PML and first order ABC) were computed on a (7×7) mesh for a fair comparison, but only the domain of interest (5×5) is shown in these plots as the extra layer in all four directions works as boundary condition in the PML case. Plots show results for u at times $t = 5$ s (top), $t = 10$ s (middle) and $t = 20$ s (bottom).

There are a number of other research directions that we would like to pursue. Extension of the method to electrodynamics and elastodynamics is relatively straightforward. Of particular interest is the possibility of developing a hybrid scheme that combines the explicit HDG method with the implicit one [47] to have even better performance than both the explicit scheme and the implicit scheme.

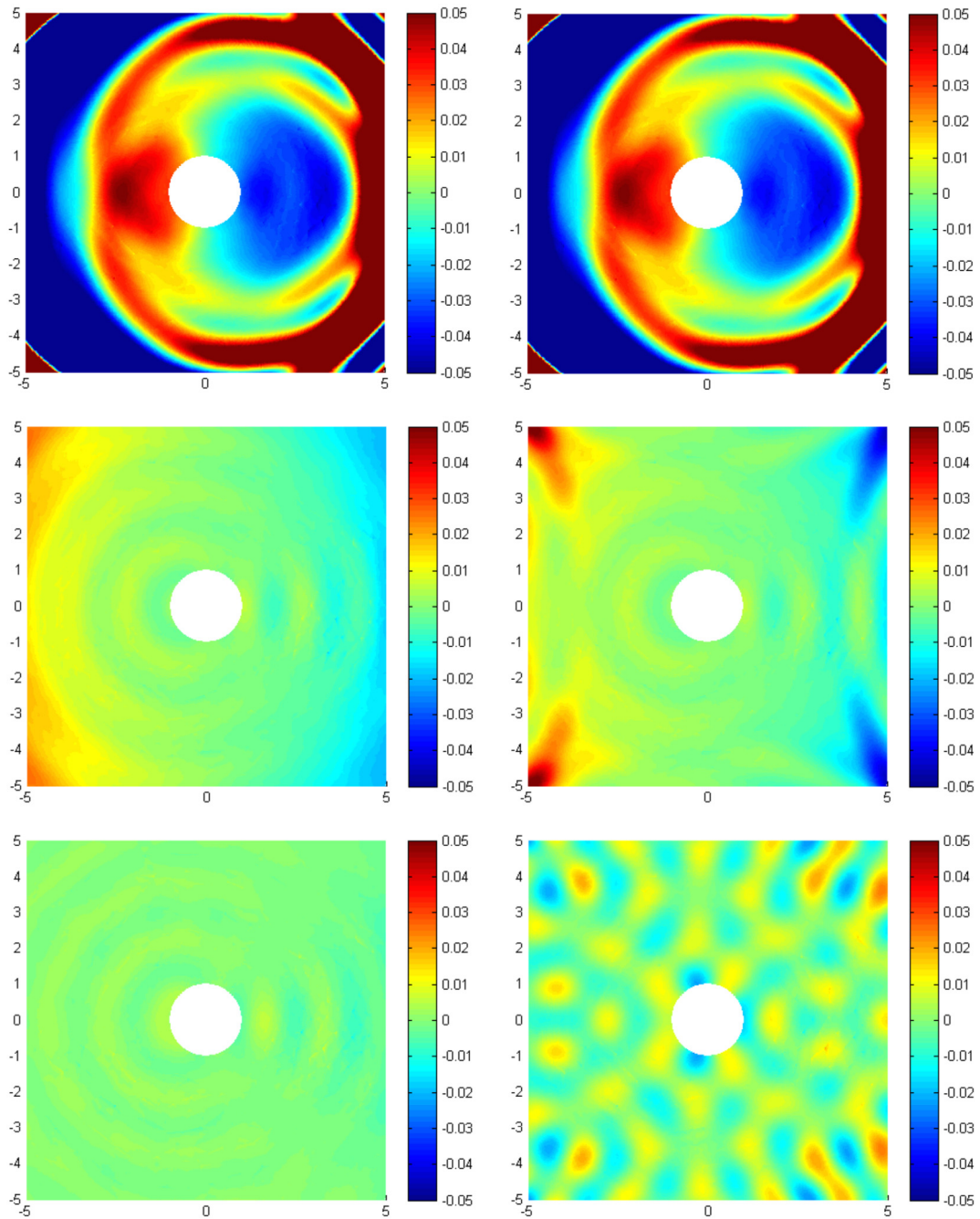


Fig. 5. Simulation of the wave scattering problem with a surrounding PML (left) and a first order ABC (right) using the explicit HDG method with a classical fourth order Runge–Kutta scheme ($\delta t = 0.00125$). Both solutions (PML and first order ABC) were computed on a (7×7) mesh for a fair comparison, but only the domain of interest (5×5) is shown in these plots as the extra layer in all four directions works as boundary condition in the PML case. Plots show results for the error $u - u_{exact}$ at times $t = 5$ s (top), $t = 10$ s (middle) and $t = 20$ s (bottom).

Acknowledgments

N.C. Nguyen and J. Peraire would like to acknowledge the partial support by the Air Force Office of Scientific Research under the AFOSR Grant FA9550-12-1-0357. B. Cockburn was partially supported by the National Science Foundation (Grant DMS-1115331) and by the Minnesota Supercomputing Institute.

References

- [1] M. Ainsworth, P. Monk, W. Muniz, Dispersive and dissipative properties of discontinuous Galerkin finite element methods for the second-order wave equation, *J. Sci. Comput.* 27 (1–3) (2006) 5–40.
- [2] T. Bui-Thanh, O. Ghattas, Analysis of an *hp*-nonconforming discontinuous Galerkin spectral element method for wave propagation, *SIAM J. Numer. Anal.* 50 (3) (2012) 1801–1826.
- [3] M.-H. Chen, B. Cockburn, F. Reitich, High-order RKDG methods for computational electromagnetics, *J. Sci. Comput.* 22/23 (2005) 205–226.
- [4] E.T. Chung, B. Engquist, Optimal discontinuous Galerkin methods for the acoustic wave equation in higher dimensions, *SIAM J. Numer. Anal.* 47 (5) (2009) 3820–3848.
- [5] B. Cockburn, C.-W. Shu, Runge-Kutta discontinuous Galerkin methods for convection-dominated problems, *J. Sci. Comput.* 16 (2001) 173–261.
- [6] B. Cockburn, F. Li, C.-W. Shu, Locally divergence-free discontinuous Galerkin methods for the Maxwell equations, *J. Comput. Phys.* 194 (2) (2004) 588–610.
- [7] G. Cohen, X. Ferrieres, S. Pernet, A spatial high-order hexahedral discontinuous Galerkin method to solve Maxwell's equations in time domain, *J. Comput. Phys.* 217 (2) (2006) 340–363.
- [8] M. Dumbser, M. Käser, E.F. Toro, An arbitrary high-order Discontinuous Galerkin method for elastic waves on unstructured meshes—V. Local time stepping and *p*-adaptivity, *Geophys. J. Int.* 171 (2) (2007) 695–717.
- [9] X. Feng, Y. Xing, Absolutely stable local discontinuous Galerkin methods for the Helmholtz equation with large wave number, *Math. Comp.* 82 (283) (2012) 1269–1296.
- [10] G. Giorgiani, S. Fernández-Méndez, A. Huerta, Hybridizable discontinuous Galerkin *p*-adaptivity for wave propagation problems, *Internat. J. Numer. Methods Fluids* 72 (12) (2013) 1244–1262.
- [11] G. Giorgiani, D. Modesto, S. Fernández-Méndez, A. Huerta, High-order continuous and discontinuous Galerkin methods for wave problems, *Internat. J. Numer. Methods Fluids* 73 (10) (2013) 883–903.
- [12] F.X. Giraldo, J.S. Hesthaven, T. Warburton, Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations, *J. Comput. Phys.* 181 (2) (2002) 499–525.
- [13] S. Govindjee, P.-O. Persson, A time-domain discontinuous Galerkin method for mechanical resonator quality factor computations, *J. Comput. Phys.* 231 (19) (2012) 6380–6392.
- [14] M.J. Grote, A. Schneebeli, D. Schötzau, Discontinuous Galerkin finite element method for the wave equation, *SIAM J. Numer. Anal.* 44 (6) (2006) 2408–2431.
- [15] J.S. Hesthaven, T. Warburton, Nodal high-order methods on unstructured grids I. Time-domain solution of Maxwell's equations, *J. Comput. Phys.* 181 (1) (2002) 186–221.
- [16] L. Noels, R. Radovitzky, An explicit discontinuous Galerkin method for non-linear solid dynamics: Formulation, parallel implementation and scalability properties, *Internat. J. Numer. Methods Engrg.* 74 (2008) 1393–1420.
- [17] P.-O. Persson, J. Bonet, J. Peraire, Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains, *Comput. Methods Appl. Mech. Engrg.* 198 (17–20) (2009) 1585–1595.
- [18] R. Sevilla, S. Fernández-Méndez, A. Huerta, NURBS-enhanced finite element method (NEFEM), *Internat. J. Numer. Methods Engrg.* 76 (1) (2008) 56–83.
- [19] L.C. Wilcox, G. Stadler, C. Burstedde, O. Ghattas, A high-order discontinuous Galerkin method for wave propagation through coupled elastic–acoustic media, *J. Comput. Phys.* 229 (24) (2010) 9373–9396.
- [20] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, *SIAM J. Numer. Anal.* 35 (6) (1998) 2440–2463.
- [21] L. Krivodonova, R. Qin, An analysis of the spectrum of the discontinuous Galerkin method, *Appl. Numer. Math.* 64 (February) (2013) 1–18.
- [22] J. Peraire, P.-O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, *SIAM J. Sci. Comput.* 30 (4) (2008) 1806–1824.
- [23] A. Huerta, A. Angeloski, X. Roca, J. Peraire, Efficiency of high-order elements for continuous and discontinuous Galerkin methods, *Internat. J. Numer. Methods Engrg.* 96 (9) (2013) 529–560.
- [24] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, *SIAM J. Numer. Anal.* 39 (5) (2001) 1749–1779.
- [25] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems, *SIAM J. Numer. Anal.* 47 (2) (2009) 1319–1365.
- [26] B. Cockburn, B. Dong, J. Guzmán, A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems, *Math. Comp.* 77 (2008) 1887–1916.
- [27] B. Cockburn, J. Gopalakrishnan, F.-J. Sayas, A projection-based error analysis of HDG methods, *Math. Comp.* 79 (2010) 1351–1367.
- [28] B. Cockburn, J. Guzmán, H. Wang, Superconvergent discontinuous Galerkin methods for second-order elliptic problems, *Math. Comp.* 78 (2009) 1–24.
- [29] B. Chabaud, B. Cockburn, Uniform-in-time superconvergence of HDG methods for the heat equation, *Math. Comp.* 81 (2012) 107–129.
- [30] L.N.T. Huynh, N.C. Nguyen, J. Peraire, B.C. Khoo, A high-order hybridizable discontinuous Galerkin method for elliptic interface problems, *Internat. J. Numer. Methods Engrg.* 93 (2) (2013) 183–200.
- [31] B. Cockburn, B. Dong, J. Guzmán, M. Restelli, R. Sacco, A hybridizable discontinuous Galerkin method for steady-state convection–diffusion–reaction problems, *SIAM J. Sci. Comput.* 31 (5) (2009) 3827–3846.
- [32] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for linear convection diffusion equations, *J. Comput. Phys.* 228 (9) (2009) 3232–3254.
- [33] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection diffusion equations, *J. Comput. Phys.* 228 (23) (2009) 8841–8855.

- [34] M.P. Ueckermann, P.F.J. Lermusiaux, High-order schemes for 2D unsteady biogeochemical ocean models, *Ocean Dyn.* 60 (6) (2010) 1415–1445.
- [35] B. Cockburn, J. Gopalakrishnan, N.C. Nguyen, J. Peraire, F.-J. Sayas, Analysis of HDG methods for Stokes flow, *Math. Comp.* 80 (2011) 723–760.
- [36] B. Cockburn, N.C. Nguyen, J. Peraire, A comparison of HDG methods for Stokes flow, *J. Sci. Comput.* 45 (1–3) (2010) 215–237.
- [37] B. Cockburn, F.J. Sayas, Divergence-conforming HDG methods for Stokes flow, *Math. Comp.* 83 (2014) 1571–1598.
- [38] N.C. Nguyen, J. Peraire, B. Cockburn, A hybridizable discontinuous Galerkin method for Stokes flow, *Comput. Methods Appl. Mech. Engrg.* 199 (9–12) (2010) 582–597.
- [39] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations, *J. Comput. Phys.* 230 (4) (2011) 1147–1170.
- [40] D. Moro, N.C. Nguyen, J. Peraire, Navier–Stokes solution using hybridizable discontinuous Galerkin methods, in: 20th AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, Honolulu, Hawaii, June, 2011, p. AIAA–2011–3407.
- [41] N.C. Nguyen, J. Peraire, Hybridizable discontinuous Galerkin methods for partial differential equations in continuum mechanics, *J. Comput. Phys.* 231 (18) (2012) 5955–5988.
- [42] J. Peraire, N.C. Nguyen, B. Cockburn, A hybridizable discontinuous Galerkin method for the compressible Euler and Navier–Stokes equations, in: 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, number June, 2010, p. AIAA–2010–363.
- [43] J. Schütz, G. May, A hybrid mixed method for the compressible Navier–Stokes equations, *J. Comput. Phys.* 240 (2013) 58–75.
- [44] F. Celiker, B. Cockburn, K. Shi, Hybridizable discontinuous Galerkin methods for timoshenko beams, *J. Sci. Comput.* 44 (1) (2010) 1–37.
- [45] S.-C. Soon, B. Cockburn, H.K. Stolarski, A hybridizable discontinuous Galerkin method for linear elasticity, *Internat. J. Numer. Methods Engrg.* 80 (8) (2009) 1058–1092.
- [46] B. Cockburn, V. Quenneville-Bélaïr, Uniform-in-time superconvergence of HDG methods for the acoustic wave equation, *Math. Comp.* 83 (2014) 65–85.
- [47] N.C. Nguyen, J. Peraire, B. Cockburn, High-order implicit hybridizable discontinuous Galerkin methods for acoustics and elastodynamics, *J. Comput. Phys.* 230 (10) (2011) 3695–3718.
- [48] R. Griesmaier, P. Monk, Error analysis for a hybridizable discontinuous Galerkin method for the Helmholtz equation, *J. Sci. Comput.* 49 (3) (2011) 291–310.
- [49] N.C. Nguyen, J. Peraire, B. Cockburn, Hybridizable discontinuous Galerkin methods for the time-harmonic Maxwell’s equations, *J. Comput. Phys.* 230 (19) (2011) 7151–7175.
- [50] L. Li, S. Lanteri, R. Perrussel, A hybridizable discontinuous Galerkin method combined to a Schwarz algorithm for the solution of 3D time-harmonic Maxwell’s equation, *J. Comput. Phys.* 256 (January) (2014) 563–581.
- [51] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability preserving high order time discretization methods, *SIAM Rev.* 43 (2000) 89–112.
- [52] I. Sim, Nonreflecting boundary conditions for time-dependent wave propagation (Ph.D. thesis), University of Basel, 2010.
- [53] B. Kaltenbacher, M. Kaltenbacher, I. Sim, A modified and stable version of a perfectly matched layer technique for the 3-d second order wave equation in time domain with an application to aeroacoustics, *J. Comput. Phys.* 235 (100) (2013) 407–422.