Space-Time Hybridizable Discontinuous Galerkin Method for the Advection–Diffusion Equation on Moving and Deforming Meshes

Sander Rhebergen and Bernardo Cockburn

Abstract We present the first space-time hybridizable discontinuous Galerkin finite element method for the advection–diffusion equation. Space-time discontinuous Galerkin methods have been proven to be very well suited for moving and deforming meshes which automatically satisfy the so-called Geometric Conservation law, for being able to provide higher-order accurate approximations in both time and space by simply increasing the degree of the polynomials used for the space-time finite elements, and for easily handling space-time adaptivity strategies. The hybridizable discontinuous Galerkin methods we introduce here add to these advantages their distinctive feature, namely, that the only globally-coupled degrees of freedom are those of the approximate trace of the scalar unknown. This results in a significant reduction of the size of the matrices to be numerically inverted, a more efficient implementation, and even better accuracy. We introduce the method, discuss its implementation and numerically explore its convergence properties.

Keywords Discontinuous Galerkin methods · Advection–diffusion equations · Space-time methods

1 Introduction

Many applications in fluid dynamics require the solution of a set of partial differential equations in time-dependent flow domains. Examples include fluid-structure interaction, moving spatial configurations (e.g., helicopter rotors) and flows with free-surfaces (e.g., wave impacts on coastal and off-shore structures), see, e.g., [21] and [10]. The accurate solution of partial differential equations by a numerical method on moving and deforming meshes, however, is non-trivial. Many schemes fail to preserve the trivial solution of a uniform flow field on dynamic meshes. This condition, the so-called Geometric Conservation Law (GCL), was proved to be essential for the accuracy of the solution [13].

One class of numerical methods that automatically satisfies the GCL is the space-time Discontinuous Galerkin (DG) method. The main example is nothing but the first DG method [25], originally devised for the numerical simulation of neutron transport. Extensions have been obtained and successfully used in a wide variety of applications, e.g., the compressible Euler and Navier–Stokes equations [11, 24, 33],

C.A. de Moura, C.S. Kubrusly (eds.), *The Courant–Friedrichs–Lewy (CFL) Condition*,
DOI 10.1007/978-0-8176-8394-8_4,

the shallow-water equations [1, 2, 31], two-phase flows [27, 29], hyperbolic nonconservative partial differential equations [26], and advection–diffusion and Oseen flows [30, 32]. In addition to their versatility, these methods can provide higher-order accurate approximations in both time and space and are ideally suited for *hp*-adaptivity. On the other hand, they are computationally expensive and so require the use of sophisticated solvers like Newton-GMRES solvers for the Navier–Stokes equations [22] and like optimized multigrid methods, see [12, 28, 34, 35] for the case of advection-dominated flows.

Recently, a new class of discontinuous Galerkin methods, namely, the hybridizable Discontinuous Galerkin method (HDG) [3, 6, 8], see also [5, 9], was introduced, in the framework of diffusion problems, with the *sole* purpose of reducing the computational complexity of these methods. In the HDG method, the approximate scalar variable and its corresponding flux are expressed in terms of an approximate trace of the scalar variable on the element faces. By enforcing the continuity of the normal component of the flux across the faces, a unique value for the approximate trace can be defined. A global system of equations for the approximate trace only is thus obtained, therefore significantly reducing the globally-coupled degrees of freedom of the discontinuous Galerkin method. The HDG method is computationally more efficient, can be more efficiently implemented, and is more accurate than all previously known discontinuous Galerkin methods. The method has been extended to time-dependent linear and nonlinear convection–diffusion in [4, 14, 15], to incompressible fluid flow [7, 17–19] and to the compressible Euler and Navier–Stokes equations [20]; see the recent review [16]. In all these papers, when dealing with time-dependent problems, implicit finite difference or Runge–Kutta time-marching methods were used. In this article, we extend the HDG method for the first time to a space-time setting. The resulting method thus combines the advantages of a space-time DG method with the efficiency and accuracy of the HDG methods.

The outline of this article is as follows. In Sect. 2, we introduce the advection–diffusion equation to which, in Sect. 3, we apply the space-time HDG method. A thorough numerical study of the convergence properties of the method is presented in Sect. 4. We end this article with some concluding remarks in Sect. 5.

2 The Advection–Diffusion Equation

We consider the following time-dependent advection–diffusion model problem:

$$\begin{aligned} u_{,0} + (a_k u - \kappa_{ks} u_{,s})_{,k} &= f && \text{in } \mathcal{E}, \\ u &= u_0 && \text{on } \Omega(t_0), \\ u &= g_D && \text{on } \mathcal{Q}_D, \end{aligned} \tag{1}$$

where a comma notation denotes differentiation with respect to the Cartesian coordinate x_k and the summation convention is used on repeated indices. Here $\mathcal{E} \in \mathbb{R}^{d+1}$ is the physical space-time domain (with d the spatial dimension), f is a source term,

$a(x) \in \mathbb{R}^d$ a given advective divergence-free velocity field, and $\kappa(x) \in \mathbb{R}^{d\times d}$ a positive definite diffusion tensor. The initial flow field is denoted by u_0 and the Dirichlet boundary data, g_D, is defined on the Dirichlet boundary $\mathcal{Q}_D$.

By introducing an auxiliary variable $\theta_k = -\kappa_{ks} u_{,s}$, we can rewrite (1) as a first-order system of equations:

$$u_{,0} + (a_k u + \theta_k)_{,k} = f \quad \text{in } \mathcal{E}, \tag{2a}$$

$$\theta_k + \kappa_{ks} u_{,s} = 0 \quad \text{in } \mathcal{E}, \tag{2b}$$

with the boundary conditions

$$\begin{aligned} u &= u_0 \quad \text{on } \Omega(t_0), \\ u &= g_D \quad \text{on } \mathcal{Q}_D. \end{aligned} \tag{2c}$$

3 The Space-Time HDG Method

In this section, we will present the space-time hybridizable discontinuous Galerkin method. We closely follow the notation of, e.g., [11, 26, 33] to highlight the similarities and differences between a space-time HDG and a space-time DG method.

3.1 *Space-Time Notation*

In a space-time method, space and time variables are not distinguished. A point at time $t = x_0$ with position vector $\bar{x} = (x_1, x_2, \ldots, x_d)$ has Cartesian coordinates $(x_0, \bar{x})$ in the open domain $\mathcal{E} \subset \mathbb{R}^{d+1}$. At time t the flow domain $\Omega(t)$ is defined as $\Omega(t) := \{\bar{x} \in \mathbb{R}^d : (t, \bar{x}) \in \mathcal{E}\}$. Let the initial and final time of the evolution of the space-time domain be denoted by t_0 and T, then the boundary of the space-time domain, $\partial\mathcal{E}$, consists of the hyper-surfaces

$$\begin{aligned} \Omega(t_0) &:= \{x \in \partial\mathcal{E} : x_0 = t_0\}, \\ \Omega(T) &:= \{x \in \partial\mathcal{E} : x_0 = T\}, \\ \mathcal{Q} &:= \{x \in \partial\mathcal{E} : t_0 < x_0 < T\}. \end{aligned}$$

The time interval $[t_0, T]$ is partitioned using the time levels $t_0 < t_1 < \cdots < T$, where the nth time interval is defined as $I_n = (t_n, t_{n+1})$ with length $\Delta t_n = t_{n+1} - t_n$. The space-time domain $\mathcal{E}$ is then divided into N_t space-time slabs $\mathcal{E}^n = \mathcal{E} \cap I_n$. Each space-time slab $\mathcal{E}^n$ is bounded by $\Omega(t_n)$, $\Omega(t_{n+1})$ and $\mathcal{Q}^n = \partial\mathcal{E}^n / (\Omega(t_n) \cup \Omega(t_{n+1}))$.

The flow domain $\Omega(t_n)$ is approximated by $\Omega_h(t_n)$, where $\Omega_h(t) \to \Omega(t)$ as $h \to 0$, with h the radius of the smallest sphere completely containing the largest space-time element. The domain $\Omega_h(t_n)$ is divided into N_n non-overlapping spatial elements $K_j(t_n)$. Similarly, $\Omega(t_{n+1})$ is approximated by $\Omega_h(t_{n+1})$. The space-time elements $\mathcal{K}_j^n$ are constructed by connecting K_j^n with K_j^{n+1} by using linear interpolation in time. In case of curved boundaries, a higher order accurate interpolation is used for elements connected to the domain boundary. The space-time elements $\mathcal{K}^n$ are connected to a master element $\widehat{\mathcal{K}}$ by an iso-parametric mapping $G_{\mathcal{K}}^n$. The tessellation $\mathcal{T}_h^n$ of the space-time slab $\mathcal{E}_h^n$ consists of all space-time elements $\mathcal{K}_j^n$; thus the tessellation $\mathcal{T}_h$ of the discrete flow domain $\mathcal{E}_h := \bigcup_{n=0}^{N_t-1} \mathcal{E}_h^n$ then is defined as $\mathcal{T}_h := \bigcup_{n=0}^{N_t-1} \mathcal{T}_h^n$.

The element boundary $\partial\mathcal{K}_j^n$, which is the union of open faces of $\mathcal{K}_j^n$, consists of three parts: $K_j(t_n^+) = \lim_{\epsilon\downarrow 0} K_j(t_n + \epsilon)$, $K_j(t_{n+1}^-) = \lim_{\epsilon\downarrow 0} K_j(t_{n+1} - \epsilon)$, and $\mathcal{Q}_j^n = \partial\mathcal{K}_j^n / (K_j(t_n^+) \cup K_j(t_{n+1}^-))$. We define $\mathcal{S}_h^n$ as the set of surfaces $\mathcal{S}$ of the form $\mathcal{Q}_j^n \cap \partial\mathcal{E}$ or of the form $\mathcal{Q}_j^n \cap \mathcal{Q}_{j'}^n$. We set $\mathcal{S}_h := \bigcup_{n=0}^{N_t-1} \mathcal{S}_h^n$.

To obtain the Arbitrary Lagrangian Eulerian (ALE) formulation, we have to introduce the grid velocity $v \in \mathbb{R}^d$. Let $\bar{x}(t^n)$ be a point on $\mathcal{Q}_j^n$ with $x_0 = t^n$. As the mesh moves, the point $\bar{x}(t^n)$ moves along $\mathcal{Q}_j^n$ to $\bar{x}(t^{n+1})$ according to some prescribed movement defined by $\bar{x}(t) = V(t; \bar{x}(t^n))$, $t \in I_n$, with V a given function. The grid velocity on $\mathcal{Q}_j^n$ is then defined by $v = \partial_t V$. The outward space-time normal vector at an element boundary point on $\partial\mathcal{K}_j^n$ can then be shown to be given by [33]:

$$n = \begin{bmatrix} (1, \bar{0}) & \text{at } K_j(t_{n+1}^-), \\ (-1, \bar{0}) & \text{at } K_j(t_n^+), \\ (-v_k \bar{n}_k, \bar{n}) & \text{at } \mathcal{Q}_j^n, \end{bmatrix} \tag{3}$$

where $\bar{0} \in \mathbb{R}^d$ and $\bar{n} \in \mathbb{R}^d$ the space-component of the space-time normal.

3.2 Approximation Spaces

Let $P^p(\mathcal{K})$ denote the space of polynomials of degree at most p on the reference element $\hat{\mathcal{K}}$ and consider $L^2(\Omega)$, that is, the space of square integrable functions on Ω. We introduce the discontinuous finite element spaces

$$W_h^p = \left\{ \omega \in L^2(\mathcal{E}_h) : \omega|_{\mathcal{K}} \circ G_{\mathcal{K}} \in P^p(\hat{\mathcal{K}}), \forall \mathcal{K} \in \mathcal{T}_h \right\},$$

and

$$V_h^p = \left\{ v \in \left(L^2(\mathcal{E}_h)\right)^d : v|_{\mathcal{K}} \circ G_{\mathcal{K}} \in \left(P^p(\hat{\mathcal{K}})\right)^d, \forall \mathcal{K} \in \mathcal{T}_h \right\}.$$

We also introduce a traced finite element space:

$$M_h^p = \left\{ \mu \in L^2(\mathcal{S}_h) : \mu|_{\mathcal{S}} \circ G_{\mathcal{S}} \in P^p(\hat{\mathcal{S}}), \forall \mathcal{S} \in \mathcal{S}_h \right\}.$$

We set $M_h^p(g_D) = \{\mu \in M_h^p : \mu = \mathsf{P} g_D \text{ on } \Gamma_D\}$, where P denotes the L^2-projection into the space $\{\mu|_{\partial\Omega}\ \forall \mu \in M_h^p\}$. Note that M_h^p consists of functions which are continuous inside the faces $\mathcal{S} \in \mathcal{S}_h$ and discontinuous at their borders.

3.3 Weak Formulation on Each Space-Time Element

Here, we are going to find the weak formulation on each of the space-time elements. Our objective is to be able to determine an approximation inside each space-time element only in terms of the data and on the numerical trace

$$\lambda := \hat{u}|_{\mathcal{S}_h} \in M_h^p. \tag{4}$$

We proceed as follows. Multiplying (2a) by a test function $\omega \in W_h^p$ and (2b) by a test function $\nu \in V_h^p$ and integrating by parts in space-time over an element $\mathcal{K} \in \mathcal{T}_h$, we obtain:

$$\begin{aligned} &-\int_{\mathcal{K}} \big(\omega_{,0} u + \omega_{,k}(a_k u + \theta_k)\big)\, dx \\ &\qquad + \int_{\partial\mathcal{K}} \omega^L \big(\hat{u}_{n_0} + (\widehat{a_k u} + \hat{\theta}_k)\bar{n}_k\big)\, ds = \int_{\mathcal{K}} f\omega\, dx, \end{aligned} \tag{5a}$$

$$\int_{\mathcal{K}} \nu_k \theta_k\, dx - \int_{\mathcal{K}} \nu_{k,s}\kappa_{ks} u\, dx + \int_{\mathcal{Q}} \nu_k \kappa_{ks} \hat{u}\bar{n}_s\, ds = 0. \tag{5b}$$

Here, the numerical traces $\widehat{a_k u} + \hat{\theta}_k$ and $\hat{u}$ are approximations to, respectively, $a_k u - \kappa_{ks} u_{,s}$ and u over $\partial\mathcal{K}$, and are introduced to couple local to global information as well as for stability purposes. These numerical traces will be defined later on.

To obtain the ALE-formulation to accommodate moving and deforming meshes, we follow [33] and use the definition of the space-time normal vector (3) to write the boundary integral in (5a) as:

$$\begin{aligned} &\int_{\partial\mathcal{K}} \omega\big(\hat{u}_{n_0} + (\widehat{a_k u} + \hat{\theta}_k)\bar{n}_k\big)\, ds \\ &\quad = \int_{K(t_{n+1}^-)} \omega\hat{u}\, d\bar{x} - \int_{K(t_n^+)} \omega\hat{u}\, d\bar{x} + \int_{\mathcal{Q}} \omega(\widehat{a_k u} - \hat{u}v_k + \hat{\theta}_k)\bar{n}_k\, ds. \end{aligned} \tag{6}$$

The numerical traces $\hat{u}$ on $K(t_{n+1}^-)$ and $K(t_n^+)$ are chosen inspired in a causality-in-time argument and are therefore defined as the upwind flux:

$$\hat{u} = \begin{bmatrix} u_{n+1}^- & \text{at } K(t_{n+1}^-), \\ u_n^- & \text{at } K(t_n^+), \end{bmatrix}$$

where u_n^- and u_{n+1}^- are the traces of u on $K(t_n)$ and $K(t_{n+1})$ from, respectively, the previous and the current space-time slab. The function u_0^- is nothing but the L^2-projection of the initial data u_0 into the space $\{\omega|_{\Omega_h(t_0)} : \omega \in W_h^p\}$.

To be able to solve (5a) and (5b) locally, the numerical trace must depend only on λ and on the traces obtained from the interior of the space-time element $\mathcal{K}$. To achieve this, we take the numerical traces $(\widehat{a_k u} + \hat{\theta}_k - v_k\hat{u})\bar{n}_k$ of the form

$$(\widehat{a_k u} + \hat{\theta}_k - v_k\hat{u})\bar{n}_k = (a_k - v_k)\bar{n}_k\lambda + \theta_k\bar{n}_k + \tau(u - \lambda) \quad \text{on } \mathcal{Q}, \tag{7}$$

for some positive function τ. The selection of τ shall be described later.

Finally, by using (6) in combination with the upwind flux on the time faces and the numerical trace (7), (5a) and (5b) become:

$$-\int_{\mathcal{K}} \big(\omega_{,0} u + \omega_{,k}(a_k u + \theta_k)\big)\,dx + \left(\int_{K(t_{n+1}^-)} \omega u_{n+1}^- \,d\bar{x} - \int_{K(t_n^+)} \omega u_n^- \,d\bar{x}\right) \tag{8a}$$

$$+\int_{\mathcal{Q}} \omega^L\big((a_k - v_k)\bar{n}_k\lambda + \theta_k\bar{n}_k + \tau(u-\lambda)\big)\,ds = \int_{\mathcal{K}} f\omega\,dx, \tag{8b}$$

$$\int_{\mathcal{K}} \nu_k\theta_k\,dx - \int_{\mathcal{K}} \nu_{k,s}\kappa_{ks} u\,dx + \int_{\mathcal{Q}} \nu_k\kappa_{ks}\lambda\bar{n}_s\,ds = 0, \tag{8c}$$

for all $(\omega, \nu) \in W_h^p \times V_h^p$.

3.4 The Global Weak Formulation for the Approximate Trace λ

We still need to determine λ. To do this, we require that the boundary conditions be weakly satisfied and that the normal component of the numerical trace of the flux $\widehat{a_k u} + \hat{\theta}_k - v_k\hat{u}$ given in (7) be single valued. In other words, we require that $\lambda \in M_h^p$ be the solution of

$$\lambda = \mathsf{P}(g_D) \quad \text{on } \mathcal{Q}_D^n, \tag{9a}$$

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{Q}} \mu(\widehat{a_k u} + \hat{\theta}_k - v_k\hat{u})\bar{n}_k\,ds = 0, \tag{9b}$$

for all $\mu \in M_h^p(0)$; recall that this implies that $\mu = 0$ on $\mathcal{Q}_D$.

3.5 The Geometric Conservation Law

We now prove that the Geometric Conservation Law (GCL) is automatically satisfied by the space-time HDG method. The GCL states that uniform flow must be preserved on a moving mesh. Let U denote a uniform flow field. In a uniform flow field, $\lambda = U$ and $\theta_k = 0$. Substituting this into (8a) and considering the element $\mathcal{K}$ on the time interval $(t, t+\varepsilon)$, we obtain:

$$-\int_{\mathcal{K}} (\omega_{,0} U + \omega_{,k} a_k U)\,dx + \int_{K(t+\varepsilon)} \omega U\,d\bar{x} - \int_{K(t)} \omega U\,d\bar{x}$$
$$+\int_{\mathcal{Q}} \omega(a_k - v_k)U\bar{n}_k\,ds = 0. \tag{10}$$

Note that this formulation is exactly the same as a standard space-time DG formulation in uniform flows. Since U is constant and arbitrary, U can be divided out of (10). Furthermore, we can rewrite (10) as:

$$\int_t^{t+\varepsilon}\left(-\int_{K(t)}(\omega_{,0}+\omega_{,k}a_k)\,d\bar{x}+\int_{\partial K(t)}\omega(a_k-v_k)\bar{n}_k\,d\bar{s}\right)dt$$
$$+\int_{K(t+\varepsilon)}\omega\,d\bar{x}-\int_{K(t)}\omega\,d\bar{x}=0. \tag{11}$$

With the following equality

$$\int_{K(t+\varepsilon)}\omega\,d\bar{x}-\int_{K(t)}\omega\,d\bar{x}=\int_t^{t+\varepsilon}\left(\frac{d}{dt}\int_{K(t)}\omega\,d\bar{x}\right)dt,$$

noting that t, $t+\varepsilon$ are arbitrary, and considering a constant polynomial approximation, we obtain the GCL:

$$\frac{d}{dt}\int_{K(t)}d\bar{x}-\int_{\partial K(t)}v_k\bar{n}_k\,d\bar{s}=0, \tag{12}$$

using the fact that integration over a closed surface $\partial K(t)$ of a constant is equal to zero. This law states that to preserve uniform flow on a moving mesh, the change in area/volume of each element must be equal to the area/volume swept by the element boundary [13].

3.6 Existence and Uniqueness of the Approximate Solution

Next, we present a result that shows that when the stabilization function τ is suitably defined, the approximation of our space-time HDG method is well defined.

Theorem 1 *Assume that the matrix-valued function κ is symmetric and positive definite and constant on each space-time element $\mathcal{K}\in\mathcal{T}_h$. Assume that the advective velocity a is divergence-free. Then, if we take the stabilization function τ such that*

$$\tau\geq\frac{1}{2}(a_k-v_k)\bar{n}_k+\tau_0\quad on\ \mathcal{Q}\ \forall\mathcal{K}\in\mathcal{T}_h,$$

where τ_0 is a strictly positive constant, the approximate solution of the HDG method under consideration is well defined.

Proof We only have to show that if the data is equal to zero, the only solution of the weak formulation (8a)–(8c) relating λ to (θ,u) and the equations determining λ (9a), (9b) is the trivial one. It is easy to see that we only need to work on any time slab $\mathcal{E}^n$ assuming that $u_n^-=0$.

Thus, taking $\omega:=u$ in (8a), we get

$$-\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}}\big(u_{,0}u+u_{,k}(a_ku+\theta_k)\big)\,dx+\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{K(t_{n+1}^-)}\big(u_{n+1}^-\big)^2\,d\bar{x}$$
$$+\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}u(\widehat{a_ku}-\hat{u}v_k+\hat{\theta}_k)\bar{n}_k\,ds=0.$$

Integrating by parts and rearranging terms, we obtain

$$
\begin{aligned}
-\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}} u_{,k}\theta_k\,dx &+ \frac{1}{2}\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{K(t_{n+1}^-)}\left(u_{n+1}^-\right)^2 d\bar{x} \\
&+\frac{1}{2}\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{K(t_n^+)}\left(u_n^+\right)^2 d\bar{x} - \frac{1}{2}\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}} u^2(a_k - v_k)\bar{n}_k\,ds \\
&+\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}} u\,(\widehat{a_k u} - \hat{u}v_k + \hat{\theta}_k)\bar{n}_k\,ds = 0.
\end{aligned}
$$

Since the tensor-valued function κ is piecewise constant, we can take $v := \kappa^{-1}\theta$ in (8c) to get

$$
\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}}\left(\kappa^{-1}\right)_{ks}\theta_k\theta_s\,dx - \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}}\theta_{s,s}u\,dx + \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\theta_s\hat{u}\bar{n}_s\,ds = 0.
$$

Adding this equation to the previous one, we obtain

$$
\begin{aligned}
\frac{1}{2}\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{K(t_{n+1}^-)}\left(u_{n+1}^-\right)^2 d\bar{x} &+ \frac{1}{2}\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{K(t_n^+)}\left(u_n^+\right)^2 d\bar{x} \\
&+\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}}\left(\kappa^{-1}\right)_{ks}\theta_k\theta_s\,dx + \Theta_h = 0,
\end{aligned}
$$

where

$$
\Theta_h := \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\left(\theta_k(\hat{u}-u)\bar{n}_k - \frac{1}{2}u^2(a_k - v_k)\bar{n}_k + u(\widehat{a_k u} - \hat{u}v_k + \hat{\theta}_k)\bar{n}_k\right)ds.
$$

We claim that Θ_h is a dissipative term. To see this, note that, by (9b),

$$
\begin{aligned}
\Theta_h := \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\Big(&\theta_k(\hat{u}-u)\bar{n}_k - \frac{1}{2}u^2(a_k - v_k)\bar{n}_k \\
&+ (u-\hat{u})(\widehat{a_k u} - \hat{u}v_k + \hat{\theta}_k)\bar{n}_k\Big)\,ds,
\end{aligned}
$$

and by the definition of the numerical trace (7),

$$
\begin{aligned}
\Theta_h &:= \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\left(-\frac{1}{2}u^2(a_k - v_k)\bar{n}_k + (u-\lambda)\left((a_k - v_k)\bar{n}_k\lambda + \tau(u-\lambda)\right)\right)ds \\
&= \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\left(-\frac{1}{2}\left((u-\lambda)^2 + \lambda^2\right)(a_k - v_k)\bar{n}_k + \tau(u-\lambda)^2\right)ds \\
&= \sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}\left(\tau - \frac{1}{2}(a_k - v_k)\bar{n}_k\right)(u-\lambda)^2\,ds.
\end{aligned}
$$

We can now conclude that $u_{n+1}^{-}=0$ on Ω_{n+1}, that $u_n^{+}=0$ on Ω_n, that $\theta_h=0$ on $\mathcal{T}_h^n$, and that $u=\lambda$ on $\mathcal{S}_h^n$. Equation (8c) now gives that u is constant in space on the time-slab $\mathcal{T}_h^n$ and since $u=\lambda=0$ on the Dirichlet boundary, we obtain that $u=0$ on $\mathcal{T}_h^n$ and that $\lambda=0$ on $\mathcal{S}_h^n$. This completes the proof. □

3.7 *The Local Stabilization Parameter* τ

In the rest of this article, we assume $\kappa_{11}=\kappa_{22}=\kappa$ and $\kappa_{ks}=0$ otherwise. Then the local stabilization parameter τ is chosen similarly as done in [14]. We, however, slightly modify the local stabilization parameter to account for moving grids. Two options are discussed in [14], the centered scheme and the upwinded scheme. To account for the diffusion and advection effects, let $\tau=\tau_a+\tau_d$, where τ_a and τ_d are the local stabilization parameters related to the advection and diffusion, respectively. Consider an interior face $\mathcal{S}=\mathcal{Q}^L\cap\mathcal{Q}^R$ between the space-time elements $\mathcal{K}^L$ and $\mathcal{K}^R$ and denote by $(\cdot)^L$ the trace of $(\cdot)$ on $\mathcal{S}$ from $\mathcal{K}^L$, and similarly for $(\cdot)^R$. Furthermore, let $\bar{n}$ be the outward normal with respect to $\mathcal{K}^L$.

Centered Scheme To obtain a centered scheme, take on each face $\tau_a^L=\tau_a^R=\eta_a$ and $\tau_d^L=\tau_d^R=\eta_d$, where

$$\eta_a=\left|(a_k-v_k)\bar{n}_k\right|, \qquad \eta_d=\frac{\kappa}{\ell}, \tag{13}$$

and ℓ denotes a representative diffusive length scale.

Upwinded Scheme To obtain an upwinded scheme, choose $\tau_a^{L,R}$ and $\tau_d^{L,R}$ according to

$$\left(\tau_a^L,\tau_d^L\right)=(\eta_a,\eta_d)\frac{|(a_k-v_k)\bar{n}_k|+(a_k-v_k)\bar{n}_k}{2|(a_k-v_k)\bar{n}_k|},$$
$$\left(\tau_a^R,\tau_d^R\right)=(\eta_a,\eta_d)\frac{|(a_k-v_k)\bar{n}_k|-(a_k-v_k)\bar{n}_k}{2|(a_k-v_k)\bar{n}_k|},$$

with η_a and η_d given in (13).

4 Numerical Results

In this section, we consider numerical results for the space-time HDG discretization of the advection–diffusion equation. For each test case, we show the convergence history of the flow field u, the auxiliary variables θ_1 and θ_2 and the mean of the flow field $\bar{u}$. Note that

$$\|\bar{u}-\bar{u}_h\|_{L^2(\Omega)}=\sqrt{\sum_K\frac{1}{|K|}\left(\int_K(u-u_h)\,d\bar{x}\right)^2}.$$

4.1 Steady-State Solution of the Advection and the Advection–Diffusion Equation on a Uniform Mesh

In this first test case, we consider both the advection and the advection–diffusion equations on a uniform mesh. For this we consider (2a), (2b) on the space-time domain $\mathcal{E} = (0, T) \times (0, 1)^2$ where the source term $f(x_1, x_2)$ and the Dirichlet boundary condition g are such that the exact solution is given by $u(x_1, x_2) = 4 + \sin(\pi x_1)\sin(\pi x_2) + \sin(2\pi x_1) + \sin(2\pi x_2)$. We take $a_1 = a_2 = 1$ and, in the case of the advection–diffusion equation, $\kappa = 0.01$. In the case of the advection equation, $\kappa = 0$. Therefore, for this test case, we modify the definition of θ such that $\theta_k = u_{,k}$.

We use a space-time HDG discretization using linear-, quadratic-, and cubic-polynomial approximations and obtain convergence orders. The local stabilization parameter τ is chosen such that we obtain a *central scheme*. For this steady-state problem, we take one physical time step of $T = \Delta t = 10^{15}$. In Tables 1 and 2, we show the convergence results obtained when $\kappa = 0.01$ and $\kappa = 0$, respectively.

For the advection–diffusion equation, from Table 1, we see the expected orders of convergence for the scalar variable u and the auxiliary variables θ_1 and θ_2, namely, for a P^p polynomial approximation we obtain the orders of convergence of $p + 1$. For the mean variable $\bar{u}$, for P^1 we obtain superconvergence with order $p + 2$. For P^2 and P^3 we seem to be achieving superconvergence with order $p + 3$!

For the advection equation, from Table 2, we obtain the expected order of convergence for the scalar variable u, namely, for a P^p polynomial approximation we obtain orders of convergence $p + 1$. For the "artificial" auxiliary variables θ_1 and θ_2, we only obtain orders of convergence p. For the mean variable $\bar{u}$, we find the strange behavior that for odd $p = 1, 3$ polynomial approximation we achieve superconvergence of orders $p + 2$ while for even $p = 2$, we only achieve a convergence order of $p + 1$.

4.2 Steady-State Boundary Layer Problem

Next, we consider a boundary layer problem. Consider (2a), (2b) on the space-time domain $\mathcal{E} = (0, T) \times (0, 1)^2$ where $f = 0$ and where $g(x_1, x_2)$ equals at the domain boundary the exact steady-state solution:

$$u(x_1, x_2) = \frac{1}{2}\left(\frac{\exp(a_1/\kappa) - \exp(a_1 x_1/\kappa)}{\exp(a_1/\kappa) - 1} + \frac{\exp(a_2/\kappa) - \exp(a_2 x_2/\kappa)}{\exp(a_2/\kappa) - 1}\right).$$

In the discretization, we use a Shishkin mesh in which the coordinates (x_1^u, x_2^u) of a uniform mesh are mapped onto a mesh suitable for dealing with boundary layers. The mapping is given by:

$$x_i = \left[\begin{array}{ll} 2(1 - \sigma_i)x_i^u, & \text{for } x_i^u < 0.5, \\ 1 + 2\sigma_i(x_i^u - 1), & \text{for } x_i^u \geq 0.5, \end{array}\right. \qquad i = 1, 2,$$

Table 1 History of convergence for the steady-state advection–diffusion equation on a uniform mesh with $\kappa = 0.01$

Degree	N_{cells}	$\|u - u_h\|_{L^2(\Omega)}$		$\|\theta_1 - \theta_1^h\|_{L^2(\Omega)}$		$\|\theta_2 - \theta_2^h\|_{L^2(\Omega)}$		$\|\bar{u} - \bar{u}_h\|_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	3.53e–2	–	6.80e–1	–	6.80e–1	–	1.45e–2	–
	16	7.41e–3	2.3	2.62e–1	1.4	2.62e–1	1.4	1.64e–3	3.1
	32	1.66e–3	2.2	8.84e–2	1.6	8.84e–2	1.6	1.92e–4	3.1
	64	3.93e–4	2.1	2.67e–2	1.7	2.67e–2	1.7	2.38e–5	3.0
	128	9.59e–5	2.0	7.44e–3	1.8	7.44e–3	1.8	3.05e–6	3.0
2	8	2.52e–3	–	5.30e–2	–	5.30e–2	–	1.99e–3	–
	16	2.19e–4	3.5	9.46e–3	2.5	9.46e–3	2.5	1.50e–4	3.7
	32	1.59e–5	3.8	1.54e–3	2.6	1.54e–3	2.6	7.46e–6	4.3
	64	1.28e–6	3.6	2.25e–4	2.8	2.25e–4	2.8	2.90e–7	4.7
	128	1.29e–7	3.3	3.06e–5	2.9	3.06e–5	2.9	1.00e–8	4.9
3	8	9.71e–5	–	1.80e–3	–	1.80e–3	–	2.67e–5	–
	16	6.01e–6	4.0	1.29e–4	3.8	1.29e–4	3.8	3.66e–7	6.2
	32	3.64e–7	4.0	8.26e–6	4.0	8.26e–6	4.0	3.36e–9	6.8
	64	2.13e–8	4.1	5.34e–7	4.0	5.34e–7	4.0	2.74e–11	6.9
	128	1.27e–9	4.1	3.52e–8	3.9	3.52e–8	3.9	5.23e–13	5.7

Table 2 History of convergence for the steady-state advection equation on a uniform mesh in which $\kappa = 0$

Degree	N_{cells}	$\|u - u_h\|_{L^2(\Omega)}$		$\|\theta_1 - \theta_1^h\|_{L^2(\Omega)}$		$\|\theta_2 - \theta_2^h\|_{L^2(\Omega)}$		$\|\bar{u} - \bar{u}_h\|_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	4.43e–2	–	9.71e–1	–	9.71e–1	–	2.22e–2	–
	16	1.01e–2	2.1	5.05e–1	0.9	5.05e–1	0.9	3.06e–3	2.9
	32	2.45e–3	2.0	2.55e–1	1.0	2.55e–1	1.0	3.97e–4	2.9
	64	6.06e–4	2.0	1.28e–1	1.0	1.28e–1	1.0	5.05e–5	3.0
	128	1.51e–4	2.0	6.39e–2	1.0	6.39e–2	1.0	6.35e–6	3.0
2	8	4.31e–3	–	8.57e–2	–	8.57e–2	–	3.89e–3	–
	16	5.76e–4	2.9	2.07e–2	2.0	2.07e–2	2.0	5.26e–4	2.9
	32	7.34e–5	3.0	5.13e–3	2.0	5.13e–3	2.0	6.73e–5	3.0
	64	9.24e–6	3.0	1.28e–3	2.0	1.28e–3	2.0	8.48e–6	3.0
	128	1.16e–6	3.0	3.20e–4	2.0	3.20e–4	2.0	1.06e–6	3.0
3	8	1.20e–4	–	4.77e–3	–	4.77e–3	–	8.77e–5	–
	16	5.82e–6	4.4	5.86e–4	3.0	5.86e–4	3.0	2.85e–6	4.9
	32	3.28e–7	4.1	7.30e–5	3.0	7.30e–5	3.0	9.08e–8	5.0
	64	1.99e–8	4.0	9.12e–6	3.0	9.12e–6	3.0	2.87e–9	5.0
	128	1.23e–9	4.0	1.14e–6	3.0	1.14e–6	3.0	9.16e–11	5.0

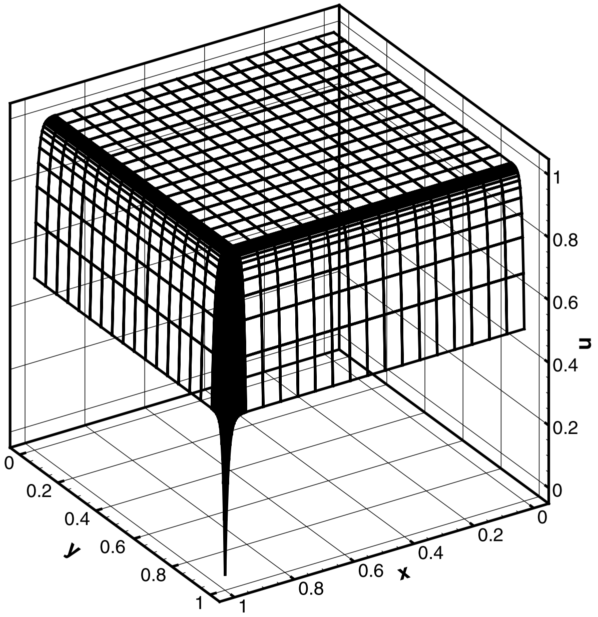

Fig. 1 The steady-state solution of the boundary layer problem using a cubic polynomial approximation on a grid with 32 × 32 elements (using a central flux)

where $\sigma_i = \min(0.5, 2\kappa/a_i \ln(N_i))$, and where N_i is the number of elements in the x_i direction (see, e.g., [28]). For this test case, we take $a_1 = a_2 = 1$ and $\kappa = 0.01$.

We solve the above problem using a space-time HDG discretization using linear-, quadratic-, and cubic- polynomial approximations in space. For steady-state test-cases, it is sufficient to take a constant polynomial approximation. The local stabilization parameter τ is chosen such that we obtain a *central scheme*. We will consider two cases for the diffusive length scale, ℓ, in (13), namely $\ell = 1$ and $\ell = \min(\sqrt{|K_L|}, \sqrt{|K_R|})$, in which $K_{L,R}$ are the areas of the two spatial elements adjacent to the face on which τ is evaluated. Furthermore, we set $\tau = 0$ on all boundaries. We remark that if $\tau \neq 0$ on the boundaries, for this test case we do not achieve expected convergence rates. For this steady-state problem, we take one physical time step of $T = \Delta t = 10^{15}$. The steady-state solution is depicted in Fig. 1. Tables 3 and 4 show the obtained convergence orders when $\ell = 1$ and $\ell = \min(\sqrt{|K_L|}, \sqrt{|K_R|})$, respectively.

For this test case, from Tables 3 and 4, we see that the diffusive length scale ℓ in the stabilization parameter has an effect on the order of convergence for the different variables. For degrees 1 and 2, if $\ell = \min(\sqrt{|K_L|}, \sqrt{|K_R|})$, we only obtain orders of convergence p for the auxiliary variables θ_1 and θ_2 and for degree $p = 1$, $\bar{u}$ converges with order $p + 1$. On the other hand, u converges with orders $p + 1$ for all considered p, the auxiliary variables converge with orders of at least $p + 1$ for $p = 3$ and $\bar{u}$ superconverges with order $p + 2$ for $p = 2, 3$.

Table 3 History of convergence for the steady-state boundary layer problem with $\ell = 1$

Degree	N_{cells}	$\|u - u_h\|_{L^2(\Omega)}$		$\|\theta_1 - \theta_1^h\|_{L^2(\Omega)}$		$\|\theta_2 - \theta_2^h\|_{L^2(\Omega)}$		$\|\bar{u} - \bar{u}_h\|_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	8.46e–3	–	2.54e–3	–	2.54e–3	–	7.59e–4	–
	16	3.40e–3	1.3	1.09e–3	1.2	1.09e–3	1.2	2.26e–4	1.7
	32	1.16e–3	1.6	4.14e–4	1.4	4.14e–4	1.4	5.37e–5	2.1
	64	3.61e–4	1.7	1.47e–4	1.5	1.47e–4	1.5	1.10e–5	2.3
	128	1.06e–4	1.8	5.00e–5	1.6	5.00e–5	1.6	2.04e–6	2.4
2	8	7.08e–4	–	3.15e–4	–	3.15e–4	–	1.29e–4	–
	16	1.71e–4	2.0	5.62e–5	2.5	5.62e–5	2.5	7.96e–6	4.0
	32	3.61e–5	2.2	1.14e–5	2.3	1.14e–5	2.3	8.70e–7	3.2
	64	6.71e–6	2.4	2.38e–6	2.3	2.38e–6	2.3	5.26e–8	4.0
	128	1.13e–6	2.6	4.69e–7	2.3	4.69e–7	2.3	1.11e–9	5.6
3	8	4.14e–4	–	2.89e–4	–	2.89e–4	–	1.48e–4	–
	16	5.44e–5	2.9	3.84e–5	2.9	3.84e–5	2.9	1.21e–5	3.6
	32	3.72e–6	3.9	2.63e–6	3.9	2.63e–6	3.9	3.60e–7	5.1
	64	1.29e–7	4.9	9.17e–8	4.8	9.17e–8	4.8	4.06e–9	6.5
	128	5.80e–9	4.5	4.10e–9	4.5	4.10e–9	4.5	1.97e–11	7.7

Table 4 History of convergence for the steady-state boundary layer problem with $\ell = \min(\sqrt{|K_L|}, \sqrt{|K_R|})$

Degree	N_{cells}	$\|u - u_h\|_{L^2(\Omega)}$		$\|\theta_1 - \theta_1^h\|_{L^2(\Omega)}$		$\|\theta_2 - \theta_2^h\|_{L^2(\Omega)}$		$\|\bar{u} - \bar{u}_h\|_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	7.57e–3	–	2.54e–3	–	2.54e–3	–	7.85e–4	–
	16	2.83e–3	1.4	1.13e–3	1.2	1.13e–3	1.2	2.42e–4	1.7
	32	8.78e–4	1.7	4.82e–4	1.2	4.82e–4	1.2	6.53e–5	1.9
	64	2.42e–4	1.9	2.27e–4	1.0	2.27e–4	1.0	1.77e–5	1.9
	128	6.42e–5	1.9	1.20e–4	0.9	1.20e–4	0.9	5.29e–6	1.7
2	8	7.97e–4	–	3.15e–4	–	3.15e–4	–	1.27e–4	–
	16	1.59e–4	2.3	5.95e–5	2.4	5.95e–5	2.4	9.00e–6	3.8
	32	2.69e–5	2.6	1.47e–5	2.0	1.47e–5	2.0	1.44e–6	2.6
	64	4.09e–6	2.7	4.39e–6	1.7	4.39e–6	1.7	9.69e–8	3.9
	128	5.55e–7	2.9	1.42e–6	1.6	1.42e–6	1.6	2.72e–9	5.2
3	8	4.29e–4	–	2.96e–4	–	2.96e–4	–	1.54e–4	–
	16	5.76e–5	2.9	4.03e–5	2.9	4.03e–5	2.9	1.31e–5	3.6
	32	4.06e–6	3.8	2.86e–6	3.8	2.86e–6	3.8	4.40e–7	4.9
	64	1.44e–7	4.8	1.02e–7	4.8	1.02e–7	4.8	5.96e–9	6.2
	128	6.06e–9	4.6	4.29e–9	4.6	4.29e–9	4.6	3.55e–11	7.4

Taking $\ell = 1$, for degree 1 and 2, it seems that u has difficulty converging with the expected orders of convergence $p+1$, while the auxiliary variables and the mean show better orders of convergence than for the case $\ell = \min(\sqrt{|K_L|}, \sqrt{|K_R|})$. For degree $p = 3$, u, θ_1, θ_2 converge with orders of at least $p+1$ and $\bar{u}$ converges with order of at least $p+2$!

4.3 A Rotating Gaussian Pulse on a Moving/Deforming Mesh

Finally, we consider the transport of a two-dimensional rotating Gaussian pulse, a test case that was presented in [14]. We, however, consider a moving and deforming space-time domain $\mathcal{E}$. Let the rotating velocity field be prescribed as $a = (-4x_2, 4x_1)$. We consider the solution at final time $T = \pi/4$, which is the time period for one-half rotation of the Gaussian pulse. The initial condition is given by

$$u_0(x_1, x_2) = \exp\left(-\frac{(x_1 - x_{1c})^2 + (x_2 - x_{2c})^2}{2\sigma^2}\right),$$

where (x_{1c}, x_{2c}) is the center and σ is the standard deviation. The exact solution with constant diffusivity constant κ is given by

$$u(x_1, x_2) = \frac{2\sigma^2}{2\sigma^2 + 4\kappa t} \exp\left(-\frac{(\tilde{x}_1 - x_{1c})^2 + (\tilde{x}_2 - x_{2c})^2}{2\sigma^2 + 4\kappa t}\right),$$

where $\tilde{x}_1 = x_1 \cos(4t) + x_2 \sin(4t)$ and $\tilde{x}_2 = -x_1 \sin(4t) + x_2 \cos(4t)$. The Dirichlet boundary condition g is deduced from the exact solution. As in [14], we choose $(x_{1c}, x_{2c}) = (-0.2, 0)$ and take $\sigma = 0.1$. As diffusivity constant, we take $\kappa = 0.01$.

The deformation of the space-time domain $\mathcal{E}$ is based on the following transformation of a uniform mesh of the space-time domain $[t, t+\Delta t] \times [-0.5, 0.5]^2$. Let (x_0^u, x_1^u, x_2^u) be the coordinates on the uniform mesh. Then we consider the following mapping:

$$x_i = x_i^u + A\left(\frac{1}{2} - x_i^u\right) \sin\left(2\pi\left(\frac{1}{2} - x_*^u + t^*\right)\right),$$

$$t^* = \begin{bmatrix} t & \text{if } x_0^u = t, \\ t + \Delta t & \text{if } x_0^u = t + \Delta t, \end{bmatrix}$$

$$x_*^u = \begin{bmatrix} x_2 & \text{if } i = 1, \\ x_1 & \text{if } i = 2, \end{bmatrix}$$

where A is the amplitude. We set $A = 0.1$. Furthermore, for the diffusivity constant we take $\kappa = 0.01$. We consider the convergence properties of the space-time HDG method for two given CFL numbers, namely $CFL = 1$ and $CFL = 10$. The history of convergence for the given CFL numbers is given in, respectively, Tables 5 and 6. In Fig. 2, we show some snapshots of the solution and mesh at different time levels.

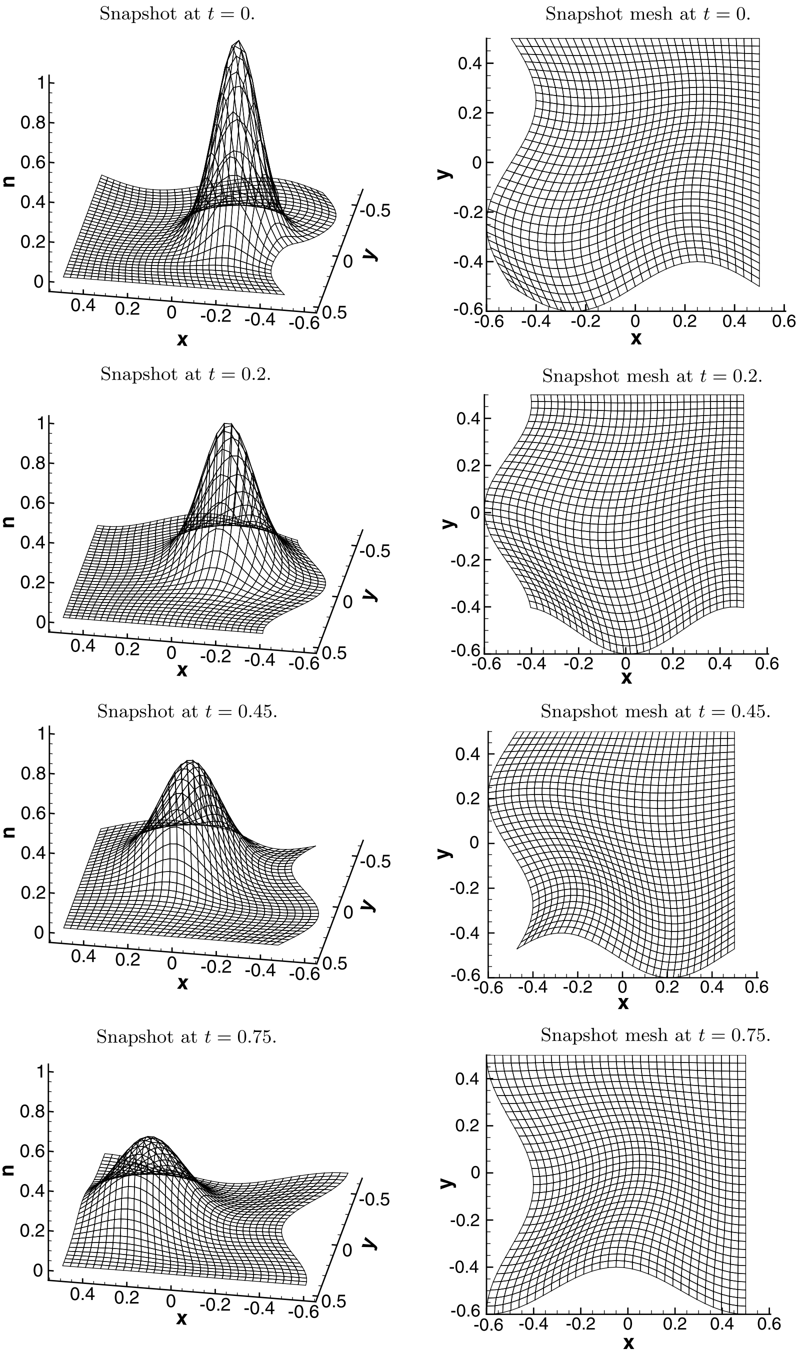

Fig. 2 Snapshots of the rotating Gaussian pulse on a moving/deforming mesh with $CFL = 1$ on a grid with 32×32 elements using a cubic polynomial approximation

Table 5 History of convergence for the rotating Gaussian pulse on a moving/deforming mesh with *CFL* = 1

Degree	N_{cells}	$\lVert u - u_h \rVert_{L^2(\Omega)}$		$\lVert \theta_1 - \theta_1^h \rVert_{L^2(\Omega)}$		$\lVert \theta_2 - \theta_2^h \rVert_{L^2(\Omega)}$		$\lVert \bar{u} - \bar{u}_h \rVert_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	1.50e–2	–	1.26e–3	–	1.23e–3	–	1.31e–2	–
	16	3.26e–3	2.2	3.48e–4	1.9	3.57e–4	1.8	2.93e–3	2.2
	32	5.46e–4	2.6	8.30e–5	2.1	9.31e–5	1.9	4.47e–4	2.7
	64	9.78e–5	2.5	2.19e–5	1.9	2.60e–5	1.8	5.91e–5	2.9
2	8	1.61e–3	–	2.09e–4	–	2.22e–4	–	1.22e–3	–
	16	1.19e–4	3.8	2.93e–5	2.8	3.46e–5	2.7	6.19e–5	4.3
	32	1.22e–5	3.3	4.78e–6	2.6	5.71e–6	2.6	2.25e–6	4.8
	64	1.48e–6	3.0	7.30e–7	2.7	8.81e–7	2.7	7.61e–8	4.9
3	8	1.49e–4	–	2.91e–5	–	3.34e–5	–	8.67e–5	–
	16	6.72e–6	4.5	2.46e–6	3.6	2.82e–6	3.6	1.21e–6	6.2
	32	3.94e–7	4.1	2.06e–7	3.6	2.35e–7	3.6	1.41e–8	6.4
	64	2.38e–8	4.0	1.60e–8	3.7	1.81e–8	3.7	2.07e–10	6.0

Table 6 History of convergence for the rotating Gaussian pulse on a moving/deforming mesh with *CFL* = 10

Degree	N_{cells}	$\lVert u - u_h \rVert_{L^2(\Omega)}$		$\lVert \theta_1 - \theta_1^h \rVert_{L^2(\Omega)}$		$\lVert \theta_2 - \theta_2^h \rVert_{L^2(\Omega)}$		$\lVert \bar{u} - \bar{u}_h \rVert_{L^2(\Omega)}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	8	4.44e–2	–	2.76e–3	–	3.51e–3	–	4.04e–2	–
	16	1.79e–2	1.3	1.25e–3	1.1	1.56e–3	1.2	1.69e–2	1.3
	32	4.81e–3	1.9	3.44e–4	1.9	4.84e–4	1.7	4.57e–3	1.9
	64	9.43e–4	2.4	7.67e–5	2.2	1.05e–4	2.2	8.64e–4	2.4
2	8	2.01e–2	–	1.70e–3	–	1.90e–3	–	1.69e–2	–
	16	3.44e–3	2.5	3.57e–4	2.3	3.85e–4	2.3	3.09e–3	2.5
	32	2.83e–4	3.6	3.42e–5	3.4	4.46e–5	3.1	2.25e–4	3.8
	64	2.37e–5	3.6	4.69e–6	2.9	6.03e–6	2.9	9.71e–6	4.5
3	8	7.64e–3	–	9.59e–4	–	8.82e–4	–	6.03e–3	–
	16	5.08e–4	3.9	7.67e–5	3.6	8.38e–5	3.4	4.13e–4	3.9
	32	2.47e–5	4.4	5.49e–6	3.8	6.87e–6	3.6	7.97e–6	5.7
	64	1.64e–6	3.9	3.99e–7	3.8	5.40e–7	3.7	1.36e–7	5.9

For this test case, from Tables 5 and 6, we consider the effect of the *CFL* number on the convergence orders. Even though the mesh is moving/deforming, the results are very good. Indeed, for $CFL = 1$ for degree p, we achieve for u, θ_1, θ_2 the order of convergence $p + 1$ while $\bar{u}$ seems to superconverge with order at least $p + 2$! Moreover, for $CFL = 10$, we also find orders of convergence $p + 1$ for u, θ_1, θ_2. For $p = 1$, it seems that $\bar{u}$ only converges with order $p + 1$, but for $p = 2, 3$, we find again that $\bar{u}$ superconvergence with an order of at least $(p + 2)$!

5 Conclusions

We have introduced and numerically tested the first space-time HDG method for time-dependent advection–diffusion problems. We have showed that, when the stabilization function is suitably defined, the method provides optimally convergent approximations, even in the advection-dominated regime and with highly-deformed and moving meshes. Moreover, the superconvergence of the local averages seems to be, to the knowledge of the authors, a new phenomenon whose theoretical study constitutes the subject of ongoing research.

Acknowledgements All test cases were implemented using hpGEM [23] for which we thank V.R. Ambati for technical support. Sander Rhebergen gratefully acknowledges funding by a Rubicon Fellowship from the Netherlands Organisation for Scientific Research (NWO) and the Marie Curie Cofund Action. Bernardo Cockburn was supported in part by the National Science Foundation (Grant DMS-0712955) and by the University of Minnesota Supercomputing Institute.

References

1. Ambati, V.R., Bokhove, O.: Space-time discontinuous Galerkin finite element method for shallow water flows. J. Comput. Appl. Math. **204**, 452–462 (2007)
2. Ambati, V.R., Bokhove, O.: Space-time discontinuous Galerkin discretization of rotating shallow water equations. J. Comput. Phys. **225**, 1233–1261 (2007)
3. Cockburn, B., Dong, B., Guzmán, J.: A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems. Math. Comput. **77**, 1887–1916 (2008)
4. Cockburn, B., Dong, B., Guzmán, J., Restelli, M., Sacco, R.: Superconvergent and optimally convergent LDG-hybridizable discontinuous Galerkin methods for convection-diffusion-reaction problems. SIAM J. Sci. Comput. **31**, 3827–3846 (2009)
5. Cockburn, B., Gopalakrishnan, J.: A characterization of hybridized mixed methods for the Dirichlet problem. SIAM J. Numer. Anal. **42**, 283–301 (2004)
6. Cockburn, B., Gopalakrishnan, J., Lazarov, R.: Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. SIAM J. Numer. Anal. **47**, 1319–1365 (2009)
7. Cockburn, B., Gopalakrishnan, J., Nguyen, N.C., Peraire, J., Sayas, F.J.: Analysis of an HDG method for Stokes flow. Math. Comput. **80**, 723–760 (2011)
8. Cockburn, B., Gopalakrishnan, J., Sayas, F.-J.: A projection-based error analysis of HDG methods. Math. Comput. **79**, 1351–1367 (2010)
9. Gopalakrishnan, J., Tan, S.: A convergent multigrid cycle for the hybridized mixed method. Numer. Linear Algebra Appl. **16**, 689–714 (2009)
10. Hueber, B., Walhorn, E., Dinkler, D.: A monolithic approach to fluid-structure interaction using space-time finite elements. Comput. Methods Appl. Mech. Eng. **193**, 2087–2104 (2004)

11. Klaij, C.M., van der Vegt, J.J.W., van der Ven, H.: Space-time discontinuous Galerkin method for the compressible Navier–Stokes equations. J. Comput. Phys. **217**, 589 (2006)
12. Klaij, C.M., van Raalte, M.H., van der Ven, H., van der Vegt, J.J.W.: Vegt, h-multigrid for space-time discontinuous Galerkin discretizations of the compressible Navier–Stokes equations. J. Comput. Phys. **227**, 1024–1045 (2007)
13. Lesoinne, M., Farhat, C.: Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. Comput. Methods Appl. Mech. Eng. **134**, 71–90 (1996)
14. Nguyen, N.C., Peraire, J., Cockburn, B.: An implicit high-order hybridizable discontinuous Galerkin method for linear convection–diffusion equations. J. Comput. Phys. **228**, 3232–3254 (2009)
15. Nguyen, N.C., Peraire, J., Cockburn, B.: An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection–diffusion equations. J. Comput. Phys. **228**, 8841–8855 (2009)
16. Nguyen, N.C., Peraire, J., Cockburn, B.: Hybridizable discontinuous Galerkin methods. In: Proceedings of the International Conference on Spectral and High Order Methods, June 2009, Trondheim, Norway. LNCSE. Springer, Berlin (2009)
17. Nguyen, N.C., Peraire, J., Cockburn, B.: A hybridizable discontinuous Galerkin method for Stokes flow. Comput. Methods Appl. Mech. Eng. **193**, 2087–2104 (2010). I have: **199**, 582–597 (2010)
18. Nguyen, N.C., Peraire, J., Cockburn, B.: An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. J. Comput. Phys. **230**, 1147–1170 (2011)
19. Peraire, J., Nguyen, N.C., Cockburn, B.: A hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. In: AIAA, Orlando, Florida, p. 362 (2010)
20. Peraire, J., Nguyen, N.C., Cockburn, B.: A hybridizable discontinuous Galerkin finite method for the compressible Euler and Navier–Stokes equations. In: AIAA, Orlando, Florida, p. 363 (2010)
21. Persson, P.-O., Bonet, J., Peraire, J.: Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains. Comput. Methods Appl. Mech. Eng. **198**, 1585–1595 (2009)
22. Persson, P.-O., Peraire, J.: Newton-GMRES preconditionning for discontinuous Galerkin discretizations of the Navier–Stokes equations. SIAM J. Sci. Comput. **30**, 2709–2733 (2008)
23. Pesch, L., Bell, A., Sollie, W.E.H., Ambati, V.R., Bokhove, O., van der Vegt, J.J.W.: hpGEM—a software framework for discontinuous Galerkin finite element methods. ACM Trans. Math. Softw. **33** (2007)
24. Pesch, L., van der Vegt, J.J.W.: A discontinuous Galerkin finite element discretization of the Euler equations for compressible and incompressible fluids. J. Comput. Phys. **227**, 5426–5446 (2008)
25. Reed, W.H., Hill, T.R.: Triangular mesh methods for the neutron transport equation. Los Alamos Scientific Laboratory, Report LA-UR-73-479 (1973)
26. Rhebergen, S., Bokhove, O., van der Vegt, J.J.W.: Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. J. Comput. Phys. **227**, 1887 (2008)
27. Rhebergen, S., Bokhove, O., van der Vegt, J.J.W.: Discontinuous Galerkin finite element method for shallow two-phase flows. Comput. Methods Appl. Mech. Eng. **198**, 819–830 (2009)
28. Rhebergen, S., van der Vegt, J.J.W., van der Ven, H.: Multigrid optimization for space-time discontinuous Galerkin discretizations of advection dominated flows. In: Kroll, N., Bieler, H., Deconinck, H., Couallier, V., Van der Ven, H., Sorensen, K. (eds.) ADIGMA—A European Initiative on the Development of Adaptive Higer-Order Variational Methods for Aerospace Applications. Notes on Numerical Fluid Mechanics and Multidisciplinary Design, vol. 113, pp. 257–269. Springer, Berlin (2010)

29. Sollie, W.E.H., Bokhove, O., van der Vegt, J.J.W.: Space-time discontinuous Galerkin finite element method for two-fluid flows. J. Comput. Phys. **230**, 789–817 (2011)
30. Sudirham, J.J., van der Vegt, J.J.W., van Damme, R.M.J.: Space-time discontinuous Galerkin method for advection-diffusion problems on time-dependent domains. Appl. Numer. Math. **56**, 1491–1518 (2006)
31. Tassi, P.A., Rhebergen, S., Vionnet, C.A., Bokhove, O.: A discontinuous Galerkin finite element model for river bed evolution under shallow flows. Comput. Methods Appl. Mech. Eng. **197**, 2930–2947 (2008)
32. van der Vegt, J.J.W., Sudirham, J.J.: A space-time discontinuous Galerkin method for the time-dependent Oseen equations. Appl. Numer. Math. **58**, 1892–1917 (2008)
33. van der Vegt, J.J.W., van der Ven, H.: Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows I. General formulation. J. Comput. Phys. **182**, 546–585 (2002)
34. van der Vegt, J.J.W., Rhebergen, S.: hp-Multigrid as smoother algorithm for higher order discontinuous Galerkin discretizations of advection dominated flows. Part I. Multilevel analysis. J. Comput. Phys. **231**(22), 7537–7563 (2012)
35. van der Vegt, J.J.W., Rhebergen, S.: hp-Multigrid as smoother algorithm for higher order discontinuous Galerkin discretizations of advection dominated flows. Part II. Optimization of the Runge-Kutta smoother. J. Comput. Phys. **231**(22), 7564–7583 (2012)