Commun. Comput. Phys.
doi: 10.4208/cicp.050810.090611a

Vol. **11**, No. 5, pp. 1503-1524
May 2012

# Extension of the High-Order Space-Time Discontinuous Galerkin Cell Vertex Scheme to Solve Time Dependent Diffusion Equations

Shuangzhang Tu*, Gordon W. Skelton and Qing Pang

*Department of Computer Engineering, Jackson State University, Jackson, MS 39217, USA.*

Received 5 August 2010; Accepted (in revised version) 9 June 2011

Communicated by Chi-Wang Shu

Available online 23 December 2011

**Abstract.** In this paper, the high-order space-time discontinuous Galerkin cell vertex scheme (DG-CVS) developed by the authors for hyperbolic conservation laws is extended for time dependent diffusion equations. In the extension, the treatment of the diffusive flux is exactly the same as that for the advective flux. Thanks to the Riemann-solver-free and reconstruction-free features of DG-CVS, both the advective flux and the diffusive flux are evaluated using continuous information across the cell interface. As a result, the resulting formulation with diffusive fluxes present is still consistent and does not need any extra ad hoc techniques to cure the common "variational crime" problem when traditional DG methods are applied to diffusion problems. For this reason, DG-CVS is conceptually simpler than other existing DG-typed methods. The numerical tests demonstrate that the convergence order based on the $L_2$-norm is optimal, i.e. $\mathcal{O}(h^{p+1})$ for the solution and $\mathcal{O}(h^p)$ for the solution gradients, when the basis polynomials are of odd degrees. For even-degree polynomials, the convergence order is sub-optimal for the solution and optimal for the solution gradients. The same odd-even behaviour can also be seen in some other DG-typed methods.

**AMS subject classifications**: 65M99, 76M25

**Key words**: High-order method, space-time method, discontinuous Galerkin (DG) method, cell-vertex scheme (CVS), diffusion equations.

## 1 Introduction

Recently, the authors developed a compact high order space-time scheme for hyperbolic conservation laws [1–4]. The method integrates the best features of the space-time Conservation Element/Solution Element (CE/SE) [5] method and the discontinuous Galerkin

*Corresponding author. *Email addresses:* shuangzhang.tu@jsums.edu (S. Tu), gordon.skelton@jsums.edu (G. Skelton), qing.pang@jsums.edu (Q. Pang)

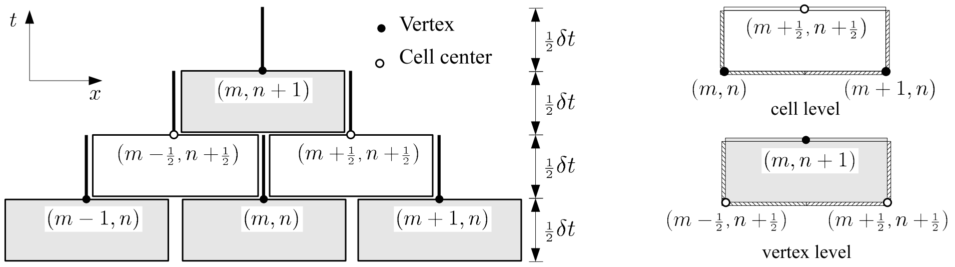

Figure 1: Solution elements (SEs) and conservation elements (CEs) in the $x$-$t$ domain. Left: solution elements; right: conservation elements.

(DG) method. The core idea is to construct a staggered space-time mesh through alternate cell-centered CEs and vertex-centered CEs (cf. Fig. 1 (right)) within each time step. Inside each SE (cf. Fig. 1 (left)), the solution is approximated using high-order space-time DG basis polynomials. The space-time flux conservation is enforced inside each CE using the DG discretization. The solution is updated successively at the cell level and at the vertex level within each physical time step. For this reason and the method's DG ingredient, the method was named as the space-time discontinuous Galerkin cell-vertex scheme (DG-CVS) [3].

DG-CVS equally works on higher dimensions on arbitrary grids. Fig. 2 shows the conservation elements and solution elements on quadrilateral meshes and triangular meshes. Obviously, the definitions of CEs and SEs on higher dimensions are analogous to that for 1-D meshes (cf. Fig. 1). Fig. 3 demonstrates the resulting dual mesh at the cell level and the vertex level for both rectangular meshes and triangular meshes, respectively.

A summary of the main features of DG-CVS is given as follows:

- *Based on space-time formulation.* The space-time formulation is advantageous in handling moving boundary problems since it automatically satisfies the so-called Geometric Conservation Law.
- *High-order accuracy in both space and time.*
- *Riemann solver free.* In contrast to the traditional DG methods, DG-CVS does not need any numerical flux. The Riemann-solver-free feature offers two-fold advantages. First, this Riemann-solver-free approach eliminates some pathological behaviours associated with some Riemann solvers. Second, it is suitable for some hyperbolic PDE systems whose eigenstructures are not explicitly known.
- *Reconstruction free.* DG-CVS solves for the solution and its all spatial and temporal derivatives simultaneously at each space-time node, thus eliminating the need of reconstruction.
- *Suitable for arbitrary spatial meshes.*
- *Highly compact regardless of order of accuracy.*

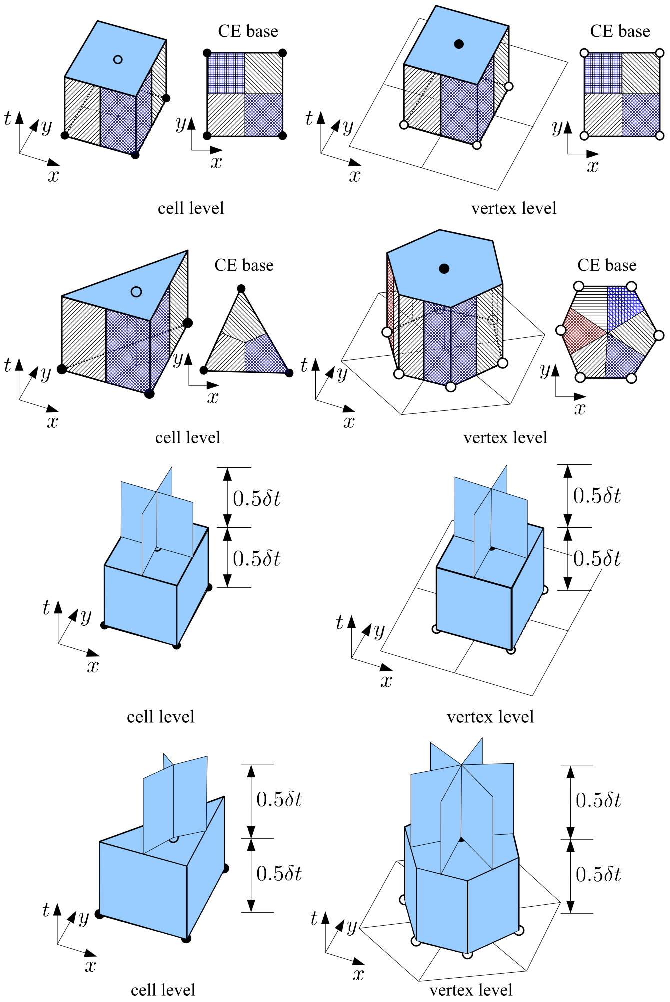

Figure 2: Conservation elements (CEs) and solution elements (SEs) in the $x$-$y$-$t$ domain. First row: CEs for rectangular meshes; second row: CEs for triangular meshes; third row: SEs for rectangular meshes; fourth row: SEs for triangular meshes.

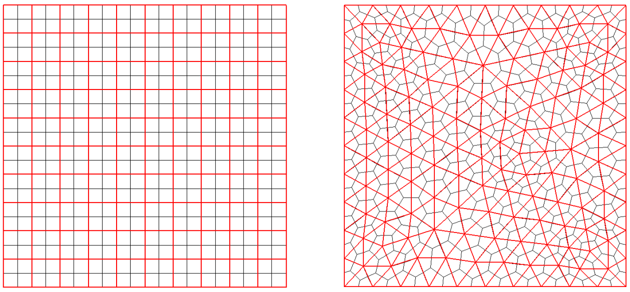

Figure 3: Dual meshes for the solution updating at the cell level (in red) and the vertex level (in black). Left: rectangular mesh; right: triangular mesh.

In this paper, we aim to show that DG-CVS for hyperbolic conservation laws can be straightforwardly extended to time dependent diffusion equations by simply incorporating the diffusive flux into the space-time flux. This method appears conceptually simpler than existing DG-based diffusion equation solvers.

It is well known that it is non-trivial to solve diffusion equations using the traditional DG method. In traditional DG methods, the solution between two adjacent cell interfaces is discontinuous. Integration by parts leads to the surface integral where a numerical flux must be provided. For the first order advective flux, the numerical flux can be obtained by considering the left and the right states of the solution across the cell interface. This approach is in spirit the same as the one for the finite volume method (e.g., Riemann solvers) and is a standard approach in many DG methods. However, for the second order diffusive flux, similar integration by parts alone has been shown to result in an inconsistent formulation since it suffers from the so-called "variational crime". In other words, the solution which satisfies the weak form does not satisfy the original differential equation. Actually, significant errors result from such simple "integration by parts" on diffusive terms.

To cure this, various methods have been proposed. In the method of Baumann-Oden [6], extra jump terms are added to the element boundaries to eliminate the inconsistency. In the Local Discontinuous Galerkin (LDG) of Cockburn and Shu [7] method which was motivated by the DG method of Bassi and Rebay [8], the second order diffusion equation is rewritten into a first order system which is further solved by the standard DG method. Refer to the unified analysis of various DG methods for elliptic equations by Arnold et al. [9] for a comprehensive comparison between various DG methods including the earlier interior penalty (IP) method. The space-time DG method of [10, 11] et al. is conceptually similar to LDG except that it makes no distinction between space and time in its DG discretization. More recently, the compact DG method of Peraire and Persson [12] is a variant of the LDG method with improved compactness. The recovery DG method of van Leer [13] uses the locally recovered solution to construct the diffusive fluxes. The central local discontinuous Galerkin methods by Liu et al. [14] which are also variants of the LDG method are built on the overlapping grids to avoid the need of numerical fluxes.

In the current DG-CVS, as will be shown in this paper, the diffusive flux is treated in exactly the same way as that for the advective fluxes. The resulting formulation does not suffer from the variational crime. The reason is simple. In DG-CVS, the space-time flux conservation is enforced in the staggered space-time mesh. Both the advective and diffusive fluxes across the spatial cell interface are evaluated using the smooth solution at the previous time level. Therefore, all spatial fluxes are continuous and unique across the cell interface. As a result, no extra reconstruction or recovery or ad hoc penalty and coupling terms are needed to ensure the consistency of the variational form for diffusive terms. For this reason, DG-CVS is conceptually simper than other existing DG methods for diffusion equations.

This paper aims to demonstrate the efficacy and performance, especially the convergence rates, of DG-CVS for transient diffusion equations. In [3], DG-CVS has been shown to exhibit optimal convergence rates based on discrete $l$-norms for hyperbolic equations, i.e. DG-CVS is $(p+1)$th order accurate when the basis polynomial is of degree $p$. In this paper, we find that, if the same $l$-norms are used to evaluate the errors for diffusion equations, inconsistent convergence behavior will be observed. To attain the optimal convergence rate for both advection and diffusion equations, the $L_2$-norm which is based on the continuous integration should be used instead. We will also find that this $L_2$ optimality only holds when the degree of basis polynomials is odd. For even degree approximation, the convergence is suboptimal. In addition, the $L_2$ optimality for odd degree approximation is found to be mesh independent. Similar odd-even behavior for odd and even approximations have also been observed in some other DG-typed methods [15–19].

The rest of the paper is organized as follows. Section 2 describes the basic discretization formulation based on DG-CVS. Section 3 presents some test results by solving time dependent advection-diffusion equations, heat equations and Burgers equation on both structured and unstructured meshes using DG-CVS. In that section, the convergence performance of DG-CVS will be presented and discussed. Finally, general concluding remarks are given in Section 4.

## 2 DG-CVS formulation for advection-diffusion equations

To illustrate the formulation of DG-CVS, we consider the following one-dimensional scalar advection-diffusion problem with appropriate initial and boundary conditions

$$\frac{\partial u(x,t)}{\partial t}+\frac{\partial f(u)}{\partial x}-\nu\frac{\partial^2 u}{\partial x^2}=0,\quad (x,t)\in[x_L,x_R]\times[0,T], \tag{2.1}$$

where $u$ is the unknown solution, $f$ is the advective flux and $\nu$ is the constant diffusion coefficient.

### 2.1 Space-time discontinuous Galerkin formulation

Following the idea of the discontinuous Galerkin (DG) method, an approximate solution $u^h$ is sought within each space-time solution element (SE), denoted as $K$. When restricted to the SE, $u^h$ belongs to the finite dimensional space $\mathcal{U}(K)$ such that

$$u^h(x,t)=\sum_{j=1}^{N}\phi_j(x,t)s_j, \tag{2.2}$$

where $\{\phi_j(x,t)\}_{j=1}^N$ are some type of space-time polynomial basis functions defined within the solution element, $\{s_j\}_{j=1}^N$ are the unknowns to be determined and $N$ is the number of basis functions depending on the degree of the polynomial function.

The first spatial derivative of the solution can be expressed in terms of the basis functions as follows.

$$\frac{\partial u^h(x,t)}{\partial x}=\sum_{j=1}^{N}\frac{\partial \phi_j}{\partial x}s_j. \tag{2.3}$$

Galerkin orthogonality states that for all test functions $v^h \in \mathcal{U}(K)$

$$\int_{\Omega_K} v^h\left(\frac{\partial u^h}{\partial t}+\frac{\partial f^h}{\partial x}-\nu\frac{\partial^2 u^h}{\partial x^2}\right)\mathrm{d}\Omega=0, \tag{2.4}$$

where $\Omega_K$ is the space-time conservation element (CE) corresponding to the solution element $K$. Note that the conservation element is identical to the solution element except for the volumeless vertical spike in the solution element as seen in Fig. 1. The space-time flux conservation in weak form as in (2.4) is for each individual space-time conservation element. Therefore, the current method can be considered as a space-time discontinuous Galerkin method.

It is sufficient to replace $v^h$ in Eq. (2.4) by each of the basis functions $\{\phi_i\}_{i=1}^N$. Integrating the resulting weak form by parts yields

$$\int_{\Omega_K}\left[\frac{\partial \phi_i}{\partial t}u^h+\frac{\partial \phi_i}{\partial x}\left(f^h-\nu\frac{\partial u^h}{\partial x}\right)\right]\mathrm{d}\Omega=\int_{\Gamma}\phi_i F^h_{\mathbf{n}}\mathrm{d}\Gamma, \tag{2.5}$$

where

$$F^h_{\mathbf{n}}=F^h\cdot\mathbf{n}=\left(f^h-\nu\frac{\partial u^h}{\partial x},u^h\right)\cdot(n_x,n_t) \tag{2.6}$$

is the space-time flux normal to the CE boundary. $\mathbf{n}=(n_x,n_t)$ is the outward unit normal of the boundary of the space-time conservation element. $\Gamma=\partial\Omega_k$ is the boundary of the CE. Note that the partial integration is also performed on the time-dependent term, which is a salient difference between space-time DG methods and semi-discrete DG methods. As can be seen, the formulation in (2.5) contains both the volume integral and the surface integral.

## 2.2 Fluxes across the space-time cell interface

The advective flux $f$ in (2.5) can be a linear or nonlinear function of the advected quantity depending on the specific problem. Here the linear case is presented for simplicity's sake. When the flux $f$ is a linear function of $u$, e.g., $f=au$ where $a$ is the constant advection speed. Then Eq. (2.5) becomes

$$\int_{\Omega_K}\left[\frac{\partial \phi_i}{\partial t}u^h+\frac{\partial \phi_i}{\partial x}\left(au^h-\nu\frac{\partial u^h}{\partial x}\right)\right]\mathrm{d}\Omega=\int_{\Gamma}\phi_i F^h_{\mathbf{n}}\mathrm{d}\Gamma. \tag{2.7}$$

From here on, for notational simplicity, the superscript '$h$' is omitted.

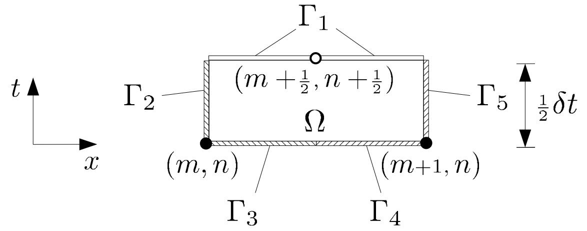

Figure 4: Illustration of space-time flux conservation element (explicit scheme).

To further illustrate the DG-CVS formulation, especially the flux terms on the right hand side of Eq. (2.7), a cell level CE shown in Fig. 4 is taken as a specific example. In Fig. 4, $m$ is the spatial index and $n$ is the temporal index. Here the solution at $(m+\frac{1}{2},n+\frac{1}{2})$ is being updated using the known information at nodes $(m,n)$ and $(m+1,n)$ at the previous half time level. As shown in Fig. 4, the CE boundary $\Gamma$ is divided into five sections $\Gamma_1$, $\Gamma_2$, $\Gamma_3$, $\Gamma_4$ and $\Gamma_5$ where

- $\Gamma_1$ belongs to the SE associated with the cell-level node $(m+\frac{1}{2},n+\frac{1}{2})$ where the solution is to be determined.
- $\Gamma_2$ and $\Gamma_3$ belong to the SE associated with the vertex-level node $(m,n)$ where the solution is already known and
- $\Gamma_4$ and $\Gamma_5$ belong to the SE associated with the vertex-level node $(m+1,n)$ where the solution is also already known.

In addition, the interior volume of the CE belongs to the SE $(m+\frac{1}{2},n+\frac{1}{2})$. Within the SEs associated with space-time nodes $(m,n)$ and $(m+1,n)$, the solution and its gradient have been obtained in the previous time level and can be evaluated according to Eqs. (2.2) and (2.3). The fluxes across $\Gamma_2$, $\Gamma_3$, $\Gamma_4$ and $\Gamma_5$ can be correspondingly computed. As a result, (2.7) becomes

$$\begin{aligned}&\int_{\Omega_K}\left[\frac{\partial\phi_i}{\partial t}u^{(m+\frac{1}{2},n+\frac{1}{2})}+\frac{\partial\phi_i}{\partial x}\left(au^{(m+\frac{1}{2},n+\frac{1}{2})}-\nu\frac{\partial u^{(m+\frac{1}{2},n+\frac{1}{2})}}{\partial x}\right)\right]\mathrm{d}\Omega-\int_{\Gamma_1}\phi_i F_{\mathbf{n}}^{(m+\frac{1}{2},n+\frac{1}{2})}\mathrm{d}\Gamma\\&=\int_{\Gamma_2+\Gamma_3}\phi_i F_{\mathbf{n}}^{(m,n)}\mathrm{d}\Gamma+\int_{\Gamma_4+\Gamma_5}\phi_i F_{\mathbf{n}}^{(m+1,n)}\mathrm{d}\Gamma,\end{aligned}\tag{2.8}$$

where the left hand side contains the unknowns at the time level $t^{n+1/2}$ and the right hand side contains the known solution at the time level $t^n$. In (2.8), the superscripts represent the space-time indices in the space-time domain, as shown in Fig. 4.

As can be seen here, the diffusive flux is treated exactly the same way as that for the advective flux. Both the advective flux and the diffusive flux across the space-time boundaries of two adjacent conservation elements are evaluated using the continuous information provided by the nodes at the previous half time level. In the derivation of the formulation, no Riemann solvers are employed for the advective flux and no techniques like penalty terms for the diffusive flux. This is the most essential feature distinguishing the current method from other DG methods.

### 2.3 Further simplification of the formulation

On boundary $\Gamma_1$, the outward unit normal $\mathbf{n}=(0,1)$. Therefore,

$$F_{\mathbf{n}}^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}=u^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)} \quad \text{on } \Gamma_1. \tag{2.9}$$

Similarly, on boundaries $\Gamma_3$ and $\Gamma_4$, the outward unit normal $\mathbf{n}=(0,-1)$, leading to

$$F_{\mathbf{n}}^{(m,n)}=-u^{(m,n)} \quad \text{on } \Gamma_3 \quad \text{and} \quad F_{\mathbf{n}}^{(m+1,n)}=-u^{(m+1,n)} \quad \text{on } \Gamma_4. \tag{2.10}$$

For stationary meshes, the side boundaries of the CE are vertical and thus have outward unit normals $\mathbf{n}=(\pm 1,0)$. As a result,

$$F_{\mathbf{n}}^{(m,n)}=-\left(au^{(m,n)}-\nu\frac{\partial u^{(m,n)}}{\partial x}\right) \quad \text{on } \Gamma_2, \tag{2.11a}$$

$$F_{\mathbf{n}}^{(m+1,n)}=au^{(m+1,n)}-\nu\frac{\partial u^{(m+1,n)}}{\partial x} \quad \text{on } \Gamma_5. \tag{2.11b}$$

Substituting all these into (2.8) to obtain

$$\begin{aligned}
&\int_{\Omega_K}\left[\frac{\partial\phi_i}{\partial t}u^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}+\frac{\partial\phi_i}{\partial x}\left(au^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}-\nu\frac{\partial u^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}}{\partial x}\right)\right]\mathrm{d}\Omega-\int_{\Gamma_1}\phi_i u^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}\mathrm{d}\Gamma\\
=&-\int_{\Gamma_3}\phi_i u^{(m,n)}\mathrm{d}\Gamma-\int_{\Gamma_2}\phi_i\left(au^{(m,n)}-\nu\frac{\partial u^{(m,n)}}{\partial x}\right)\mathrm{d}\Gamma\\
&-\int_{\Gamma_4}\phi_i u^{(m+1,n)}\mathrm{d}\Gamma+\int_{\Gamma_5}\phi_i\left(au^{(m+1,n)}-\nu\frac{\partial u^{(m+1,n)}}{\partial x}\right)\mathrm{d}\Gamma.
\end{aligned} \tag{2.12}$$

Eq. (2.12) is for updating the solution at the space-time node $(m+\frac{1}{2},n+\frac{1}{2})$ which is at the cell level. For vertex level nodes, similar formulation can be derived.

A linear equation system can be obtained from (2.12). If we substitute (2.2) and (2.3) into (2.12), the left hand side of (2.12) becomes

$$\begin{aligned}
&\int_{\Omega_K}\frac{\partial\phi_i}{\partial t}\left(\sum_{j=1}^{N}\phi_j s_j^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}\right)\mathrm{d}\Omega+a\int_{\Omega_K}\frac{\partial\phi_i}{\partial x}\left(\sum_{j=1}^{N}\phi_j s_j^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}\right)\mathrm{d}\Omega\\
&-\nu\int_{\Omega_K}\left[\frac{\partial\phi_i}{\partial x}\left(\sum_{j=1}^{N}\frac{\partial\phi_j}{\partial x}s_j^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}\right)\right]\mathrm{d}\Omega-\int_{\Gamma_1}\phi_i\left(\sum_{j=1}^{N}\phi_j s_j^{\left(m+\frac{1}{2},n+\frac{1}{2}\right)}\right)\mathrm{d}\Gamma.
\end{aligned} \tag{2.13}$$

Replacing the left hand side of (2.12) by (2.13), we obtain the following matrix-vector form

$$\mathbf{Ms}=\mathbf{b}, \tag{2.14}$$

where **M** is an $N\times N$ matrix, **s** is an $N\times 1$ vector containing the unknowns, $\{s_j\}_{j=1}^N$, and **b** is another $N\times 1$ vector containing the right hand side of (2.12) which is known. The elements of **M** and **b**, $M_{ij}$ and $b_i$ are explicitly given as

$$M_{ij}=\int_{\Omega_K}\left[\frac{\partial\phi_i}{\partial t}\phi_j+a\left(\frac{\partial\phi_i}{\partial x}\phi_j\right)-\nu\left(\frac{\partial\phi_i}{\partial x}\frac{\partial\phi_j}{\partial x}\right)\right]\mathrm{d}\Omega-\int_{\Gamma_1}\phi_i\phi_j\mathrm{d}\Gamma, \tag{2.15a}$$

$$\begin{aligned}b_i=&-\int_{\Gamma_3}\phi_i u^{(m,n)}\mathrm{d}\Gamma-\int_{\Gamma_2}\phi_i\left(au^{(m,n)}-\nu\frac{\partial u^{(m,n)}}{\partial x}\right)\mathrm{d}\Gamma\\&-\int_{\Gamma_4}\phi_i u^{(m+1,n)}\mathrm{d}\Gamma+\int_{\Gamma_5}\phi_i\left(au^{(m+1,n)}-\nu\frac{\partial u^{(m+1,n)}}{\partial x}\right)\mathrm{d}\Gamma.\end{aligned} \tag{2.15b}$$

The linear system (2.14) is solved for each space-time node. No coupling exists between adjacent nodes. Therefore, the formulation in (2.12) is globally explicit.

**Remark 2.1.** In DG-CVS, unknowns are stored at both vertices and cell centroids of the spatial mesh. However, the solutions at vertices and cell centroids are updated at different time levels within each time step in an alternate fashion. At the beginning of each physical time step ($t^n$), the solution is assumed known at the vertices of the mesh, either given as the initial condition or obtained from the previous time step. Inside each new time step, the solution is updated in two successive steps. The first step updates the solution at cell centroids at the half-time level ($t^{n+1/2}$) based on the known vertex solutions at the previous time level ($t^n$). The second step updates the solution at vertices at the new time level ($t^{n+1}$) based on the known cell solutions at the previous half-time level ($t^{n+1/2}$). The same process is repeated for new time steps.

**Remark 2.2.** The integrals in Eq. (2.12) are implemented quadrature-free. Refer to our earlier paper [3] for details. The surface integrals in Eq. (2.12) are evaluated using the Gaussian quadrature rule.

**Remark 2.3.** If the advective flux $f$ is a nonlinear function of $u$, the standard Newton-Raphson method is used to solve the local nonlinear equation system at each space-time node.

## 2.4 Quick verification

To verify that the DG-CVS formulation based on Eq. (2.12) produces correct solutions and does not suffer from the "variational crime", we solve the 1-D heat equation on $[-1,1]$ with initial solution $u_0=\sin(\pi x)$ and periodic boundary conditions. The domain is divided by 10 evenly-spaced cells. Fig. 5 shows the $p1$ and $p3$ solution at $t=0.1$. Here the local solution variation is also plotted together with the solution at the vertex. As can be seen, the $p1$ solution is fairly accurate on this coarse mesh and the $p3$ solution visually sits on top of the underlying exact solution. This simple test confirms that the DG-CVS formulation is consistent in the sense that it does not produce wrong solutions.

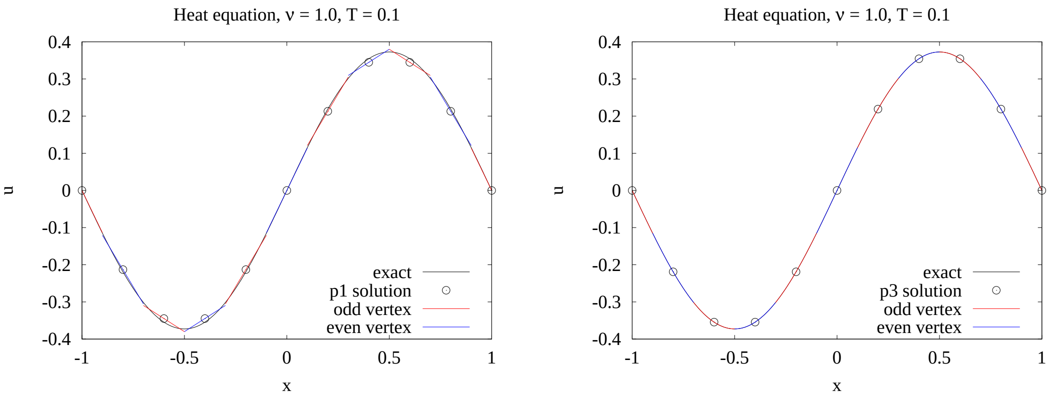

Figure 5: Solution of 1-D heat equation at $t=0.1$. Left: $p1$ solution. Right: $p3$ solution.

# 3 Numerical tests and discussions

In this section, more numerical results and discussions are presented to demonstrate the accuracy and convergence performance of DG-CVS for time dependent scalar diffusion equations.

## 3.1 Grid convergence study

An important measurement of the performance of any high-order method is its convergence order of accuracy. In the context of finite element-based methods, when basis polynomials of degree $p$ is used, and if the solution $u$ and the solution gradient $u_x$ are approximated using Eqs. (2.2) and (2.3), respectively, then the optimal convergence orders are $p+1$ for the solution $u$ and $p$ for the solution gradient. Theoretical analysis of convergence orders of high-order methods is difficult, if not impossible. However, the convergence order can be easily determined numerically via the grid convergence study with appropriate error norms. In this sub-section, the grid convergence study is conducted on the 1-D advection-diffusion equation and the 2-D heat equation. The convergence behavior of DG-CVS for the advection equation and the diffusion equation will be compared.

To determine the numerical convergence order, the following error norms are defined:

$$l_\infty(\epsilon)=\max_{i=1}^{n_\text{v}}|u^h(\mathbf{x}_i)-u^\text{exact}(\mathbf{x}_i)|, \tag{3.1a}$$

$$l_2(\epsilon)=\sqrt{\frac{1}{n_\text{v}}\sum_{i=1}^{n_\text{v}}\left(u^h(\mathbf{x}_i)-u^\text{exact}(\mathbf{x}_i)\right)^2}, \tag{3.1b}$$

$$L_2(\epsilon)=\sqrt{\frac{1}{|\Omega|}\sum_{i=1}^{n_\text{v}}\int_{\Omega_i}\left(u_i^h(\mathbf{x})-u_i^\text{exact}(\mathbf{x})\right)^2\text{d}\Omega}, \tag{3.1c}$$

where $n_V$ is the number of vertices in the computational domain, $u^h(\mathbf{x}_i)$ is the computed numerical solution at $i$th vertex and $u^{\text{exact}}(\mathbf{x}_i)$ is the analytical solution at the $i$th vertex, $\Omega_i$ is the spatial domain associated with the $i$th vertex and $|\Omega|$ is the size of the entire computational domain. Here, $l_\infty$ and $l_2$ norms are evaluated at the discrete location of vertices and $L_2$ norm is obtained by integrating the continuous solution within the spatial domain associated with each vertex.

### 3.1.1 Convergence orders on 1-D meshes

The first test is to solve the following 1-D linear scalar advection-diffusion equation

$$\frac{\partial u}{\partial t}+a\frac{\partial u}{\partial x}-\nu\frac{\partial^2 u}{\partial x^2}=0, \qquad -1\le x\le 1, \tag{3.2a}$$

$$u(x,0)=u_0(x)=\sin(\pi x), \quad \text{periodic b.c.} \tag{3.2b}$$

The analytical solution is given as

$$u_{\text{exact}}=e^{-\pi^2\nu t}\sin(\pi(x-at)).$$

The following two cases are tested:

- Pure advection equation. $a=1.0$, $\nu=0$.
- Heat equation. $a=0$, $\nu=1.0$.

For each case, four meshes (number of cells nc $=10,20,40$, and 80) are used for varying degrees of basis polynomials ($p=1,2,3$, and 4).

For the advection case, the time step is chosen as $\delta t=\sigma\delta x/a$ where $\delta x$ is the cell interval and the Courant number $\sigma=$ 0.5, 0.3125, 0.25 and 0.25 for $p1$, $p2,p3$ and $p4$ cases, respectively. For the stability limits in the case of advection equations, one can refer to our earlier paper [3]. All cases are computed up to $t=1.0$. The numerical convergence orders using the $l_\infty$, $l_2$ and $L_2$ norms are recorded in Tables 1, 2 and 3. All three tables show that the numerical convergence orders are $p+1$ for $u$ and $p$ for $u_x$ for all $p$'s. The convergence orders are optimal.

For the diffusion case, the time step is chosen as $\delta t=0.1\delta x^2/\nu$ for all $p1-p4$ cases. All cases are computed up to $t=0.1$. The numerical convergence orders based on various norms are recorded in Tables 4 and 5 and 6. Comparing with the optimal convergence rates for the advection equation, one can see that the convergence rates for the diffusion equation appear inconsistent. For clarity's sake, the following observations from Tables 4 and 5 and 6 are summarized:

- When $p$ is odd,
  - the convergence rates based on the $l_\infty$ and $l_2$ norms are optimal for $u$ and super-optimal for $u_x$;
  - the convergence rates based on the $L_2$ norm are optimal for both $u$ and $u_x$.

Table 1: 1-D advection equation. $a=1.0$, $t=1.0$. Numerical convergence order determined by $l_\infty$ norm.

| | | nc = 10 | nc = 20 | | nc = 40 | | nc = 80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $l_\infty(\epsilon)$ | $l_\infty(\epsilon)$ | order | $l_\infty(\epsilon)$ | order | $l_\infty(\epsilon)$ | order | comments |
| 1 | $u$ | 2.11E-02 | 4.88E-03 | 2.113 | 1.13E-03 | 2.112 | 2.70E-04 | 2.064 | optimal |
| | $u_x$ | 1.08E-01 | 4.38E-02 | 1.301 | 2.09E-02 | 1.071 | 1.03E-02 | 1.016 | optimal |
| 2 | $u$ | 2.13E-04 | 2.77E-05 | 2.946 | 3.51E-06 | 2.979 | 4.41E-07 | 2.995 | optimal |
| | $u_x$ | 2.89E-02 | 7.18E-03 | 2.008 | 1.79E-03 | 2.001 | 4.48E-04 | 2.000 | optimal |
| 3 | $u$ | 2.56E-05 | 1.58E-06 | 4.019 | 9.71E-08 | 4.023 | 6.04E-09 | 4.007 | optimal |
| | $u_x$ | 4.04E-04 | 6.03E-05 | 2.742 | 7.80E-06 | 2.951 | 9.83E-07 | 2.989 | optimal |
| 4 | $u$ | 4.18E-07 | 1.28E-08 | 5.031 | 3.90E-10 | 5.037 | 1.25E-11 | 4.965 | optimal |
| | $u_x$ | 5.59E-05 | 3.51E-06 | 3.992 | 2.21E-07 | 3.991 | 1.39E-08 | 3.990 | optimal |

Table 2: 1-D advection equation. $a=1.0$, $t=1.0$. Numerical convergence order determined by $l_2$ norm.

| | | nc = 10 | nc = 20 | | nc = 40 | | nc = 80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $l_2(\epsilon)$ | $l_2(\epsilon)$ | order | $l_2(\epsilon)$ | order | $l_2(\epsilon)$ | order | comments |
| 1 | $u$ | 1.48E-02 | 3.36E-03 | 2.134 | 7.88E-04 | 2.095 | 1.90E-04 | 2.055 | optimal |
| | $u_x$ | 7.71E-02 | 3.07E-02 | 1.330 | 1.46E-02 | 1.071 | 7.25E-03 | 1.010 | optimal |
| 2 | $u$ | 1.61E-04 | 2.02E-05 | 2.994 | 2.52E-06 | 3.006 | 3.14E-07 | 3.006 | optimal |
| | $u_x$ | 2.13E-02 | 5.20E-03 | 2.037 | 1.28E-03 | 2.018 | 3.19E-04 | 2.009 | optimal |
| 3 | $u$ | 1.76E-05 | 1.09E-06 | 4.013 | 6.79E-08 | 4.009 | 4.24E-09 | 3.999 | optimal |
| | $u_x$ | 2.86E-04 | 4.18E-05 | 2.774 | 5.45E-06 | 2.940 | 6.91E-07 | 2.981 | optimal |
| 4 | $u$ | 3.12E-07 | 9.28E-09 | 5.069 | 2.79E-10 | 5.057 | 8.93E-12 | 4.964 | optimal |
| | $u_x$ | 4.13E-05 | 2.54E-06 | 4.022 | 1.58E-07 | 4.008 | 9.89E-09 | 3.999 | optimal |

Table 3: 1-D advection equation. $a=1.0$, $t=1.0$. Numerical convergence order determined by $L_2$ norm.

| | | nc = 10 | nc = 20 | | nc = 40 | | nc = 80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 1.17E-02 | 2.79E-03 | 2.064 | 6.87E-04 | 2.020 | 1.71E-04 | 2.006 | optimal |
| | $u_x$ | 4.04E-01 | 2.03E-01 | 0.993 | 1.02E-01 | 0.998 | 5.09E-02 | 0.999 | optimal |
| 2 | $u$ | 5.73E-04 | 7.21E-05 | 2.990 | 9.03E-06 | 2.997 | 1.13E-06 | 2.999 | optimal |
| | $u_x$ | 3.66E-02 | 9.21E-03 | 1.990 | 2.31E-03 | 1.998 | 5.77E-04 | 1.999 | optimal |
| 3 | $u$ | 2.55E-05 | 1.64E-06 | 3.961 | 1.03E-07 | 3.988 | 6.47E-09 | 3.997 | optimal |
| | $u_x$ | 2.31E-03 | 2.94E-04 | 2.975 | 3.69E-05 | 2.992 | 4.62E-06 | 2.998 | optimal |
| 4 | $u$ | 8.02E-07 | 2.50E-08 | 5.005 | 7.80E-10 | 5.002 | 2.46E-11 | 4.986 | optimal |
| | $u_x$ | 1.00E-04 | 6.25E-06 | 3.999 | 3.90E-07 | 4.002 | 2.45E-08 | 3.996 | optimal |

Table 4: 1-D heat equation. $\nu=1.0$, $t=0.1$. Numerical convergence order determined by $l_\infty$ norm.

| | | nc$=$10 | nc$=$20 | | nc$=$40 | | nc$=$80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $l_\infty(\epsilon)$ | $l_\infty(\epsilon)$ | order | $l_\infty(\epsilon)$ | order | $l_\infty(\epsilon)$ | order | comments |
| 1 | $u$ | 9.72E-03 | 2.62E-03 | 1.894 | 6.58E-04 | 1.991 | 1.65E-04 | 1.998 | optimal |
| | $u_x$ | 3.04E-02 | 7.82E-03 | 1.959 | 1.97E-03 | 1.990 | 4.93E-04 | 1.997 | super-optimal |
| 2 | $u$ | 1.32E-03 | 3.70E-04 | 1.838 | 9.40E-05 | 1.977 | 2.36E-05 | 1.994 | sub-optimal |
| | $u_x$ | 3.17E-02 | 8.28E-03 | 1.936 | 2.09E-03 | 1.983 | 5.25E-04 | 1.996 | optimal |
| 3 | $u$ | 7.29E-05 | 5.10E-06 | 3.836 | 3.24E-07 | 3.976 | 2.03E-08 | 3.994 | optimal |
| | $u_x$ | 5.08E-05 | 3.66E-06 | 3.792 | 2.37E-07 | 3.948 | 1.50E-08 | 3.987 | super-optimal |
| 4 | $u$ | 5.14E-06 | 3.23E-07 | 3.993 | 1.99E-08 | 4.020 | 1.24E-09 | 4.006 | sub-optimal |
| | $u_x$ | 2.79E-05 | 1.77E-06 | 3.974 | 1.11E-07 | 3.992 | 6.98E-09 | 3.998 | optimal |

Table 5: 1-D heat equation. $\nu=1.0$, $t=0.1$. Numerical convergence order determined by $l_2$ norm.

| | | nc$=$10 | nc$=$20 | | nc$=$40 | | nc$=$80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $l_2(\epsilon)$ | $l_2(\epsilon)$ | order | $l_2(\epsilon)$ | order | $l_2(\epsilon)$ | order | comments |
| 1 | $u$ | 6.89E-03 | 1.81E-03 | 1.933 | 4.60E-04 | 1.974 | 1.16E-04 | 1.989 | optimal |
| | $u_x$ | 2.25E-02 | 5.66E-03 | 1.988 | 1.41E-03 | 2.006 | 3.51E-04 | 2.006 | super-optimal |
| 2 | $u$ | 9.38E-04 | 2.55E-04 | 1.877 | 6.57E-05 | 1.959 | 1.66E-05 | 1.985 | sub-optimal |
| | $u_x$ | 2.34E-02 | 5.99E-03 | 1.965 | 1.50E-03 | 2.000 | 3.74E-04 | 2.004 | optimal |
| 3 | $u$ | 5.16E-05 | 3.52E-06 | 3.875 | 2.26E-07 | 3.959 | 1.43E-08 | 3.985 | optimal |
| | $u_x$ | 3.75E-05 | 2.65E-06 | 3.822 | 1.70E-07 | 3.964 | 1.07E-08 | 3.995 | super-optimal |
| 4 | $u$ | 3.65E-06 | 2.23E-07 | 4.032 | 1.39E-08 | 4.002 | 8.71E-10 | 3.997 | sub-optimal |
| | $u_x$ | 2.06E-05 | 1.28E-06 | 4.003 | 7.98E-08 | 4.008 | 4.97E-09 | 4.006 | optimal |

Table 6: 1-D heat equation. $\nu=1.0$, $t=0.1$. Numerical convergence order determined by $L_2$ norm.

| | | nc$=$10 | nc$=$20 | | nc$=$40 | | nc$=$80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 4.90E-03 | 1.24E-03 | 1.982 | 3.11E-04 | 1.995 | 7.78E-05 | 1.999 | optimal |
| | $u_x$ | 1.49E-01 | 7.50E-02 | 0.994 | 3.75E-02 | 0.999 | 1.88E-02 | 1.000 | optimal |
| 2 | $u$ | 1.22E-03 | 2.78E-04 | 2.130 | 6.75E-05 | 2.042 | 1.68E-05 | 2.011 | sub-optimal |
| | $u_x$ | 1.54E-02 | 3.93E-03 | 1.965 | 9.90E-04 | 1.991 | 2.48E-04 | 1.998 | optimal |
| 3 | $u$ | 3.70E-05 | 2.45E-06 | 3.918 | 1.55E-07 | 3.978 | 9.76E-09 | 3.994 | optimal |
| | $u_x$ | 6.50E-04 | 8.09E-05 | 3.006 | 1.01E-05 | 3.000 | 1.26E-06 | 3.000 | optimal |
| 4 | $u$ | 3.92E-06 | 2.30E-07 | 4.094 | 1.41E-08 | 4.025 | 8.77E-10 | 4.007 | sub-optimal |
| | $u_x$ | 3.75E-05 | 2.15E-06 | 4.128 | 1.31E-07 | 4.034 | 8.14E-09 | 4.008 | optimal |

- When $p$ is even, the convergence rates based on all norms are sub-optimal for $u$ and optimal for $u_x$.

Since $l$-norms do not produce the same convergence orders for the heat equation as those for the advection equation no matter whether $p$ is odd or even, while the convergence orders based on the $L_2$-norm are optimal for both the heat equation and the advection equation when $p$ is odd, this indicates that $L_2$-norm is a more appropriate norm for determining the convergence order. We will focus on $L_2$-norm in the remaining tests.

The inconsistent convergence behavior between odd degree and even degree approximations has also been reported in the methods of many other researcher [15–19]. It is also interesting to note that the first version of the central LDG method by Liu et al. [14] is sub-optimal first order accurate for $p=1$ and optimal $(p+1)$th order accurate for $p>1$.

### 3.1.2 Convergence orders on 2-D structured and unstructured meshes

To investigate the convergence behavior of DG-CVS for 2-D diffusion equations, rectangular and unstructured triangular meshes with various resolutions are used. Fig. 6 shows the coarsest rectangular and unstructured triangular meshes used in the test. The coarsest rectangular mesh is composed of $10\times 10$ rectangular cells and is designated as "qua-10". The coarsest triangular mesh is designated as "tri-10" whose edge resolution is comparable to that of mesh qua-10. The meshes will be refined isotropically several times in the grid convergence study, resulting in a series of meshes designated as "qua-20", "qua-40", "qua-80", "tri-20", "tri-40" and "tri-80", respectively.

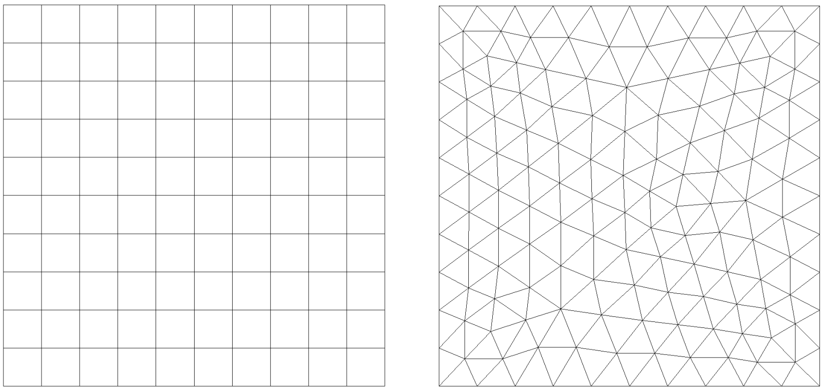

Figure 6: Two-dimensional meshes used in the tests. Left: rectangular mesh. Right: unstructured triangular mesh.

The following 2-D heat equation with sinusoidal initial solution is solved using DG-CVS.

$$\frac{\partial u}{\partial t}-\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)=0, \quad -1\le x,y\le 1, \tag{3.3a}$$

$$u(x,y,0)=\sin(\pi(x+y)), \quad \text{periodic b.c.} \tag{3.3b}$$

Table 7: 2-D heat equation with sinusoidal initial solution on rectangular meshes. $\nu=1.0$, $t=0.1$. Numerical convergence order determined by the $L_2$ norm.

| | | qua-10 | qua-20 | | qua-40 | | qua-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 4.27e-03 | 1.09e-03 | 1.970 | 2.73e-04 | 1.997 | 6.84e-05 | 1.997 | optimal |
| | $u_x$ and $u_y$ | 7.81e-02 | 3.94e-02 | 0.987 | 1.98e-02 | 0.993 | 9.89e-03 | 1.001 | optimal |
| 2 | $u$ | 8.37e-04 | 2.02e-04 | 2.051 | 5.00e-05 | 2.014 | 1.25e-05 | 2.000 | sub-optimal |
| | $u_x$ and $u_y$ | 1.24e-02 | 3.21e-03 | 1.950 | 8.11e-04 | 1.985 | 2.03e-04 | 1.998 | optimal |
| 3 | $u$ | 5.43e-05 | 3.73e-06 | 3.864 | 2.39e-07 | 3.964 | 1.51e-08 | 3.984 | optimal |
| | $u_x$ and $u_y$ | 1.28e-03 | 1.58e-04 | 3.018 | 1.97e-05 | 3.004 | 2.47e-06 | 2.996 | optimal |
| 4 | $u$ | 2.92e-06 | 1.68e-07 | 4.119 | 1.05e-08 | 4.000 | 6.54e-10 | 4.005 | sub-optimal |
| | $u_x$ and $u_y$ | 1.01e-04 | 6.06e-06 | 4.059 | 3.75e-07 | 4.014 | 2.34e-08 | 4.002 | optimal |

Table 8: 2-D heat equation with sinusoidal initial solution on triangular meshes. $\nu=1.0$, $t=0.1$. Numerical convergence order determined by the $L_2$ norm.

| | | tri-10 | tri-20 | | tri-40 | | tri-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 4.82e-03 | 1.25e-03 | 1.947 | 3.22e-04 | 1.957 | 8.12e-05 | 1.988 | optimal |
| | $u_x$ | 7.26e-02 | 3.64e-02 | 0.996 | 1.82e-02 | 1.000 | 9.13e-03 | 0.995 | optimal |
| | $u_y$ | 7.25e-02 | 3.64e-02 | 0.994 | 1.82e-02 | 1.000 | 9.13e-03 | 0.995 | optimal |
| 2 | $u$ | 1.37e-03 | 3.46e-04 | 1.985 | 8.65e-05 | 2.000 | 2.17e-05 | 1.995 | sub-optimal |
| | $u_x$ | 9.07e-03 | 2.30e-03 | 1.979 | 5.80e-04 | 1.988 | 1.46e-04 | 1.990 | optimal |
| | $u_y$ | 9.03e-03 | 2.29e-03 | 1.979 | 5.78e-04 | 1.986 | 1.45e-04 | 1.995 | optimal |
| 3 | $u$ | 3.14e-05 | 2.02e-06 | 3.958 | 1.28e-07 | 3.980 | 8.07e-09 | 3.987 | optimal |
| | $u_x$ | 7.30e-04 | 9.23e-05 | 2.983 | 1.16e-05 | 2.992 | 1.46e-06 | 2.990 | optimal |
| | $u_y$ | 7.30e-04 | 9.23e-05 | 2.983 | 1.16e-05 | 2.992 | 1.46e-06 | 2.990 | optimal |
| 4 | $u$ | 2.15e-06 | 1.33e-07 | 4.015 | 8.35e-09 | 3.994 | 5.27e-10 | 3.986 | sub-optimal |
| | $u_x$ | 4.62e-05 | 2.96e-06 | 3.964 | 1.88e-07 | 3.977 | 1.19e-08 | 3.982 | optimal |
| | $u_y$ | 4.62e-05 | 2.96e-06 | 3.964 | 1.88e-07 | 3.977 | 1.19e-08 | 3.982 | optimal |

The analytical solution is given as

$$u_{\text{exact}}=e^{-2\pi^2\nu t}\sin(\pi(x+y)).$$

The time step is chosen to be $\delta t=\sigma h^2/\nu$ where $h$ is the local characteristic size of the element and $\sigma=0.2$ for $p1$ cases and $\sigma=0.1$ for all other cases. Table 7 and Table 8 show the convergence orders on rectangular meshes and triangular meshes, respectively. Both tables show the same convergence rates which indicates the convergence orders are independent on the type of the spatial mesh. In addition, the same convergence rates as those in Table 6 for the 1-D heat equation are observed, that is, the convergence orders are optimal for both $u$ and its gradient when $p$ is odd, and sub-optimal for $u$ and optimal for $u$'s gradients when $p$ is even.

## 3.2 More tests of 2D cases

In this sub-section, we provide two more test cases. Both cases involve time dependent boundary conditions. The first case is the 2-D heat equation with the delta initial solution, and the second case is the 2-D nonlinear viscous Burgers equation.

From the previous tests, one can conclude that the convergence orders based on the $L_2$ norm is optimal for both the solution and its gradients when $p$ is odd. Besides, this optimality holds for both the advection and diffusion equations. Since in practice, the governing equations often involve both advection and diffusion simultaneously, it justifies to employ basis polynomials of odd degrees for best and consistent convergence rates. Therefore, in the remaining test cases, only results of odd $p$ will be presented.

### 3.2.1 2-D heat equation with the delta initial solution

This case is to solve the following 2-D heat equation using DG-CVS.

$$\frac{\partial u}{\partial t}-\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)=0, \qquad -1\le x,y\le 1, \tag{3.4a}$$

$$u(x,y,0)=u_0, \tag{3.4b}$$

where $u_0$ is the delta function at the origin of the domain $(0,0)$. The solution to Eq. (3.4) with such an initial condition is called the fundamental solution of the heat equation [20]. The analytical solution is given as

$$u_{\text{exact}}=\frac{1}{4\pi t}e^{\left(-\frac{x^2+y^2}{4t}\right)}.$$

The boundary conditions on the four boundaries are time varying depending on the above analytical solution formula.

The time step is chosen to be $\delta t=\sigma h^2$ where $h$ is the local characteristic size of the element and $\sigma=0.2$ for the $p1$ case and $\sigma=0.1$ for the $p3$ case. Fig. 7 shows the carpet view of the solution at three instants, $t=0.01$, $t=0.05$, and $t=0.2$, on both the rectangular and triangular meshes, respectively.

Tables 9 and 10 show the convergence rates of $p1$ and $p3$ approximations on rectangular and triangular meshes, respectively. Not surprisingly, the convergence rates are optimal for all cases.

In Fig. 8, the solution $u$ and its gradient $u_x$ along the horizontal line $y=0$ are shown together with the analytical solutions at $t=0.05$. To visually compare the accuracy between $p1$ and $p3$ results, we intentionally choose a coarse $20\times 20$ mesh. As can be seen, the $p1$ solution is not accurate enough to resolve the local extrema. By contrast, the $p3$ solution lies on top of the analytical solution.

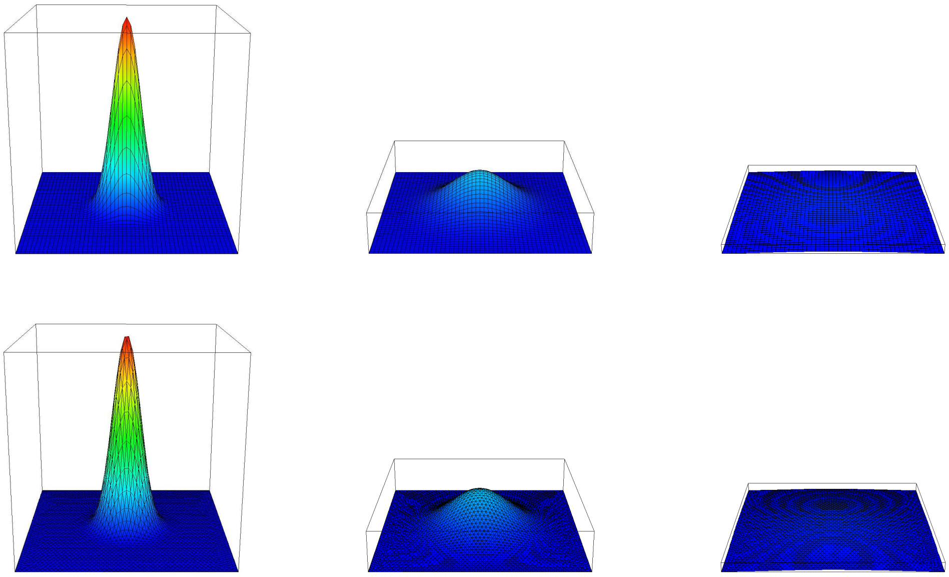

Figure 7: Solution of 2-D heat equation with the delta initial solution. Top row: solutions on the rectangular mesh qua-50. Bottom row: solutions on the triangular mesh tri-50. Left column: solution at $t=0.01$, middle column: solution at $t=0.05$. Right column: solution at $t=0.2$.

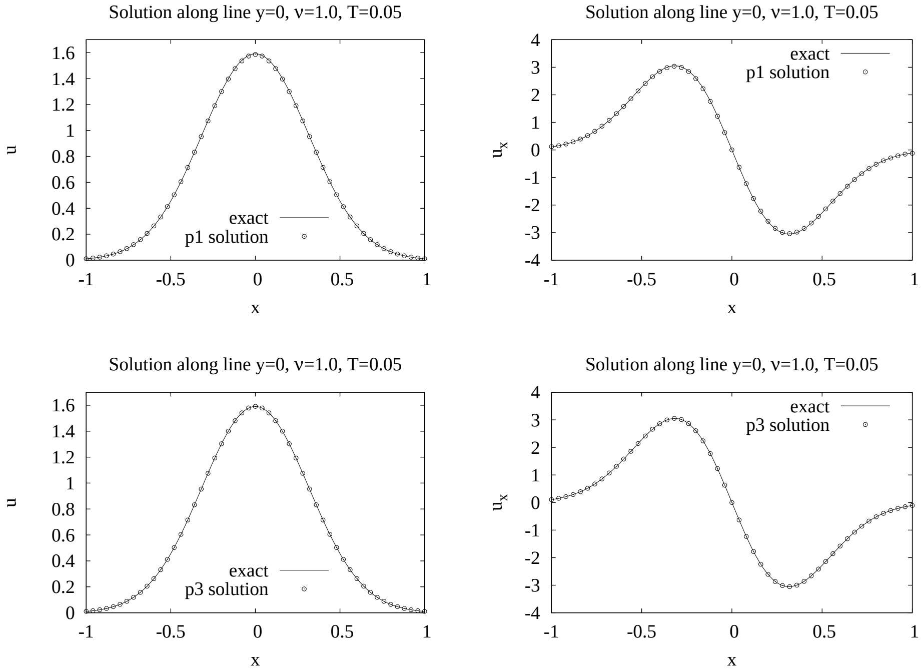

Figure 8: Comparison between $p1$ and $p3$ solutions on the $20\times20$ mesh at $y=0$ and $t=0.05$ for the case of the fundamental solution of the 2-D heat equation.

Table 9: 2-D heat equation with delta initial solution on rectangular meshes. $t=0.05$. Numerical convergence order determined by $L_2$ norm.

| | | qua-10 | qua-20 | | qua-40 | | qua-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 1.20e-02 | 3.12e-03 | 1.943 | 7.88e-04 | 1.985 | 1.98e-04 | 1.993 | optimal |
| | $u_x$ and $u_y$ | 2.53e-01 | 1.28e-01 | 0.983 | 6.43e-02 | 0.993 | 3.22e-02 | 0.998 | optimal |
| 3 | $u$ | 4.53e-04 | 3.33e-05 | 3.766 | 2.19e-06 | 3.927 | 1.38e-07 | 3.988 | optimal |
| | $u_x$ and $u_y$ | 6.82e-03 | 7.78e-04 | 3.132 | 9.38e-05 | 3.052 | 1.15e-05 | 3.028 | optimal |

Table 10: 2-D heat equation with delta initial solution on triangular meshes. $t=0.05$. Numerical convergence order determined by $L_2$ norm.

| | | tri-10 | tri-20 | | tri-40 | | tri-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 1.15e-02 | 3.02e-03 | 1.929 | 7.71e-04 | 1.970 | 1.95e-04 | 1.983 | optimal |
| | $u_x$ | 2.23e-01 | 1.12e-01 | 0.994 | 5.64e-02 | 0.990 | 2.83e-02 | 0.995 | optimal |
| | $u_y$ | 2.32e-01 | 1.17e-01 | 0.988 | 5.85e-02 | 1.000 | 2.93e-02 | 0.998 | optimal |
| 3 | $u$ | 1.72e-04 | 1.11e-05 | 3.954 | 6.99e-07 | 3.989 | 4.37e-08 | 4.000 | optimal |
| | $u_x$ | 4.12e-03 | 5.00e-04 | 3.043 | 6.17e-05 | 3.019 | 7.69e-06 | 3.004 | optimal |
| | $u_y$ | 4.46e-03 | 5.42e-04 | 3.041 | 6.71e-05 | 3.014 | 8.35e-06 | 3.006 | optimal |

### 3.2.2 2-D viscous Burgers equation

Finally, the following 2-D viscous Burgers equation is solved using DG-CVS.

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+u\frac{\partial u}{\partial y}-\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)=0, \qquad 0\le x,y\le 25, \tag{3.5}$$

with the analytical solution given as

$$u_{\text{exact}}=\frac{2}{e^{\frac{x-x_c+y-y_c-2t}{\nu}}+1},$$

where $(x_c,y_c)=(0,0)$ is a constant location.

The 1-D version of this case was presented in [21]. This case is constructed such that the original wave is propagated without changing shape under the effect of both nonlinear advection and linear diffusion. The initial solution at $t=0$ and the boundary conditions on all four boundaries are provided by the analytical solution. Therefore the boundary conditions are time dependent.

We first conduct the grid convergence study on this nonlinear advection-diffusion case. In the study, $\nu=2.5$ is chosen and the simulation is run up to $t=10$. In the current study, the time step is chosen as $\delta t=\sigma\min(h/a,h^2/\nu)$ where $h$ is the local mesh size and $\sigma=0.2$ for the $p1$ case and $\sigma=0.1$ for the $p3$ case. For the purpose of determining the convergence rates, this choice of time steps may not be appropriate since advection and

Table 11: Solution of 2-D viscous Burgers equation on rectangular meshes. $\nu$=2.5, $t$=10. Numerical convergence order determined by $L_2$ norm.

| | | qua-10 | qua-20 | | qua-40 | | qua-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 1.38e-02 | 3.15e-03 | 2.131 | 7.71e-04 | 2.031 | 1.92e-04 | 2.006 | optimal |
| | $u_x$ and $u_y$ | 1.74e-02 | 8.52e-03 | 1.030 | 4.21e-03 | 1.017 | 2.09e-03 | 1.010 | optimal |
| 3 | $u$ | 2.83e-04 | 1.69e-05 | 4.066 | 1.11e-06 | 3.928 | 7.88e-08 | 3.816 | optimal |
| | $u_x$ and $u_y$ | 7.11e-04 | 9.12e-05 | 2.963 | 1.19e-05 | 2.938 | 1.61e-06 | 2.886 | optimal |

Table 12: Solution of 2-D viscous Burgers equation on triangular meshes. $\nu$=2.5, $t$=10. Numerical convergence order determined by $L_2$ norm.

| | | tri-10 | tri-20 | | tri-40 | | tri-80 | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | variable | $L_2(\epsilon)$ | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | $L_2(\epsilon)$ | order | comments |
| 1 | $u$ | 8.08e-03 | 2.19e-03 | 1.883 | 5.49e-04 | 1.996 | 1.38e-04 | 1.992 | optimal |
| | $u_x$ | 1.54e-02 | 7.79e-03 | 0.983 | 3.86e-03 | 1.013 | 1.93e-03 | 1.000 | optimal |
| | $u_y$ | 1.54e-02 | 7.79e-03 | 0.983 | 3.86e-03 | 1.013 | 1.93e-03 | 1.000 | optimal |
| 3 | $u$ | 1.70e-04 | 1.21e-05 | 3.812 | 1.00e-06 | 3.597 | 1.03e-07 | 3.279 | optimal |
| | $u_x$ | 4.16e-04 | 5.31e-05 | 2.970 | 6.44e-06 | 3.044 | 7.96e-07 | 3.016 | optimal |
| | $u_y$ | 4.16e-04 | 5.34e-05 | 2.962 | 6.48e-06 | 3.043 | 8.02e-07 | 3.014 | optimal |

diffusion have different time scales. A more appropriate approach may be the operator splitting method where the advection and the diffusion are treated with different time steps. Tables 11 and 12 show the convergence rates of $p1$ and $p3$ on rectangular and triangular meshes, respectively. The convergence rates, though not as neat as those for the pure diffusion equation, are still close to optimal.

To further demonstrate the accuracy of DG-CVS for various values of $\nu$, the following three cases are simulated on both qua-50 and tri-50 meshes where the size of the mesh is indicated by $\delta x = 0.5$:

- short wave ($\delta x/\nu = 10$, i.e. $\nu = 0.05$);
- medium wave ($\delta x/\nu = 1$, i.e. $\nu = 0.5$);
- long wave ($\delta x/\nu = 0.2$, i.e. $\nu = 2.5$).

Fig. 9 plots the $p1$ solution along the diagonal line $x-y=0$ together with the exact solution. As can be seen, all types of waves have been captured accurately in terms of both location and magnitude. In the case of the short wave, no limiter is applied and therefore slight overshoot and undershoot can be seen. The $p3$ solution is not shown since no much visual difference can be seen between the $p3$ solution and the $p1$ solution. Fig. 10 compares the computed solution gradients ($u_x$ or $u_y$) with the exact solution for the case of $\nu = 0.5$. As can be seen, the $p3$ solution is superior to the $p1$ solution in resolving the local sharp extremum of the solution gradient.

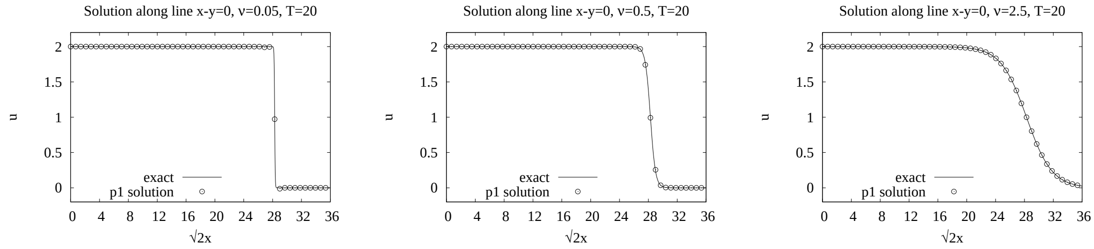

Figure 9: $p1$ solution of 2-D viscous Burgers equation along the diagonal line $x-y=0$ at $t=20$. Left: $\nu=0.05$; middle: $\nu=1.0$; right: $\nu=2.5$.

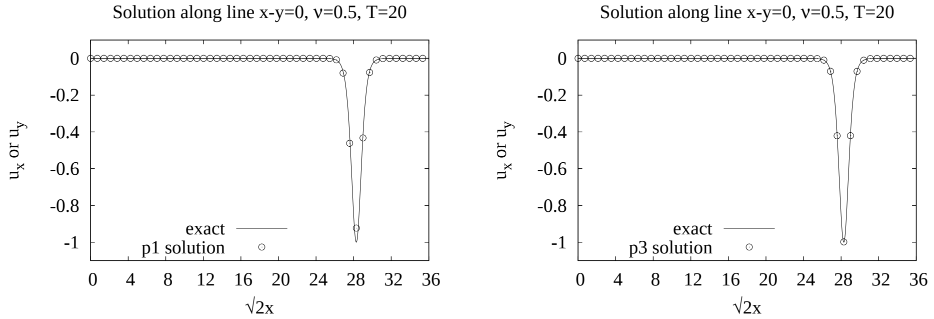

Figure 10: $u_x$ or $u_y$ of the 2-D viscous Burgers equation with $\nu=0.5$ along the diagonal line at $t=20$. Left: $p1$ solution; right: $p3$ solution.

# 4 Conclusions

The paper extends the authors' high-order space-time discontinuous Galerkin cell-vertex scheme (DG-CVS) for hyperbolic conservation laws to time dependent scalar diffusion equations. The extension is straightforward by simply incorporating the diffusive flux into the space-time flux. The space-time flux conservation is locally enforced using the space-time discontinuous Galerkin approach over staggered space-time conservation elements. With DG-CVS, the diffusive flux is treated in exactly the same way as the advection flux without any ad hoc techniques. DG-CVS updates the solution at the cell level and the vertex level successively within each physical time step. The inter-element advective and diffusive fluxes for updating the solution at a vertex (cell) are provided by the information of the adjacent cells (vertices) where the solutions have been determined at the previous half time level. Therefore, both the advective and diffusive flux is continuous and unique across the spatial cell interface. For this reason, the DG-CVS formulation for diffusion equations is consistent and does not suffer from the "variational crime". DG-CVS appears conceptually simpler than traditional DG methods for diffusion equations.

The grid convergence study reveals that DG-CVS is $L_2$ optimal for both advection and diffusion equations when the degree of basis polynomials, $p$, is odd. In other words, DG-CVS is $(p+1)$th order accurate for the solution and $p$th order accurate for the solution gradients for odd $p$. The $L_2$ optimality for odd $p$ holds for arbitrary meshes. When $p$ is even, the convergence rate of DG-CVS is still optimal for advection equations but sub-optimal for diffusion equations. In practice, one can choose the odd $p$ for best performance when solving advection-diffusion equations.

The numerical tests verify that DG-CVS accurately solves the time dependent diffusion equations as well as advection equations. Our ongoing work is to extend DG-CVS further to solve the compressible Navier-Stokes equations.

## Acknowledgments

This work is supported by the U.S. Air Force Office of Scientific Research (AFOSR) Computational Mathematics Program under the Award No. FA9550-08-1-0122 and the Award No. FA9550-10-1-0045. The authors are also grateful to the School of Engineering and the Department of Computer Engineering at Jackson State University for their support.

### References

[1] Tu, S., A high order space-time Riemann-solver-free method for solving compressible Euler equations, January 2009, AIAA Paper 2009–1335.

[2] Tu, S., A solution limiting procedure for an arbitrarily high order space-time method, June 2009, AIAA Paper 2009–3983.

[3] Tu, S., Skelton, G., and Pang, Q., A compact high order space-time method for conservation laws, Communications in Computational Physics, Vol. 9, No. 2, 2011, pp. 441–480.

[4] Tu, S. and Tian, Z., Preliminary Implementation of a High Order Space-time Method on Overset Cartesian/Quadrilateral Grids, January 2010, AIAA Paper 2010-0544.

[5] Chang, S.-C. and To, W., A new numerical framework for solving conservation laws: the method of space-time conservation element and solution element, 1991, NASA TM 1991-104495.

[6] Baumann, C. E. and Oden, J. T., A discontinuous *hp* finite element method for convection-diffusion problems, Comput. Methods Appl. Mech. Engrg., Vol. 175, 1999, pp. 311–341.

[7] Cockburn, B. and Shu, C.-W., The local Discontinuous Galerkin Method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal., Vol. 35, 1998, pp. 2440–2463.

[8] Bassi, F. and Rebay, S., A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, J. Comput. Phys., Vol. 131, No. 2, 1997, pp. 267–279.

[9] Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D., Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal., Vol. 39, No. 5, 2001, pp. 1749–1779.

[10] Sudirham, J. J., van der Vegt, J. J. W., and van Dam R. M. J., Space-time discontinuous Galerkin method for advection-diffusion problems on time-dependent domains, Appl. Numer. Math., Vol. 56, December 2006, pp. 1491–1518.

[11] Klaij, C. M., van der Vegt, J. J. W., and van der Ven, H., Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations, J. Comput. Phys., Vol. 217, September 2006, pp. 589–611.

[12] Peraire, J. and Persson, P.-O., The compact discontinuous Galerkin (CDG) method for elliptic problems, SIAM J. Sci. Comput., Vol. 30, No. 4, 2008, pp. 1806–1824.

[13] van Leer, B. and Lo, M., A Discontinuous Galerkin Method for Diffusion Based on Recovery, June 2007, AIAA Paper 2007-4083.

[14] Liu, Y., Shu, C.-W., Tadmor, E., and Zhang, M., Central Local Discontinuous Galerkin Methods on Overlapping Cells for Diffusion Equations, 2010, submitted to Mathematical Modelling and Numerical Analysis and under revision.

[15] Baumann, C. and Oden, J., A discontinuous hp finite element method for convection-diffusion problems, Comput. Methods Appl. Mech. Engrg., Vol. 175, 1999, pp. 311–341.

[16] Babuška, I., Baumann, C., and Oden, J., A discontinuous hp finite element method for diffusion problems: 1-D analysis, Computers and Mathematics with Applications, Vol. 37, No. 9, 1999, pp. 103–122.

[17] Larson, M. G. and Niklasson, A. J., Analysis of a family of discontinuous Galerkin methods for elliptic problems: the one dimensional case, Numer. Math., Vol. 99, November 2004, pp. 113–130.

[18] Romkes, A., Prudhomme, S., and Oden, J. T., Convergence analysis of a discontinuous finite element formulation based on second order derivatives, Comput. Meth. Appl. Mech. Eng., Vol. 195, 2006, pp. 3461–3482.

[19] Dolejší, V. and Havle, O., The L2-Optimality of the IIPG Method for Odd Degrees of Polynomial Approximation in 1D, J. Sci. Comput., Vol. 42, 2010, pp. 122–143.

[20] Eriksson, K., Estep, D., Hansbo, P., and Johnson, C., Computational Differential Equations, Cambridge University Press, 1996.

[21] Popescu, M., Shyy, W., and Garbey, M., Finite volume treatment of dispersion-relation-preserving and optimized prefactored compact schemes for wave propagation, J. Comput. Phys., Vol. 210, No. 2, 2005, pp. 705–729.