J. Korean Soc. Ind. Appl. Math. Vol.25, No.4, 173–195, 2021 https://doi.org/10.12941/jksiam.2021.25.173

DISCONTINUOUS GALERKIN SPECTRAL ELEMENT METHOD FOR ELLIPTIC PROBLEMS BASED ON FIRST-ORDER HYPERBOLIC SYSTEM

DEOKHUN KIM[1] AND HYUNG TAEK AHN[1†]

[1]SCHOOL OF NAVAL ARCHITECTURE AND OCEAN ENGINEERING, UNIVERSITY OF ULSAN, ULSAN 44610, KOREA
E-mail address: †htahn@ulsan.ac.kr

ABSTRACT. A new implicit discontinuous Galerkin spectral element method (DGSEM) based on the first order hyperbolic system(FOHS) is presented for solving elliptic type partial different equations, such as the Poisson problems. By utilizing the idea of hyperbolic formulation of Nishikawa[1], the original Poisson equation was reformulated in the first-order hyperbolic system. Such hyperbolic system is solved implicitly by the collocation type DGSEM. The steady state solution in pseudo-time, which is the solution of the original Poisson problem, was obtained by the implicit solution of the global linear system. The optimal polynomial orders of $O(\mathcal{h}^{p+1})$) are obtained for both the solution and gradient variables from the test cases in 1D and 2D regular grids. Spectral accuracy of the solution and gradient variables are confirmed from all test cases of using the uniform grids in 2D.

1. INTRODUCTION

Ever since the first introduction of the discontinuous Galerkin (DG) method for the steady solution of neutron transport by Reed and Hill in 1973[2] at Los Alamos National Laboratory, the method has been gaining tremendous amount of popularity and becomes a candidate for the next generation high-order numerical method. In early stage of the research, such effort was leaded by the frontiers, such as Cockburn and Shu[3-6] and their collaborators[7] and followed by others wider applications[8-13]. The popularity of the DG method is mostly attributed to its hybrid nature of combining the standard Galerkin finite element within the interior of the element for high-order accuracy and the finite volume with strict conservative property across its neighboring elements. The combination of such ideas brought the DG method an arbitrarily high-order solution on unstructured grids in a stable manner. In addition to the superior accuracy, the method enjoys flexibility as well. Because of the relaxed continuity requirement across the element interface, many advanced features can be incorporated into the DG framework with less complexity, such as using different polynomial approximation from element to element, and/or dealing with the hanging nodes from mesh refinements. In that sense, the DG method can be considered as the frontier among various

Received September 7 2021; Revised November 11 2021; Accepted in revised form November 12 2021; Published online December 25 2021.

2000 *Mathematics Subject Classification*. 93B05.

Key words and phrases. Discontinuous Galerkin(DG) method, Spectral element method, Poisson equation, First order hyperbolic system(FOHS).

† Corresponding author.

kinds of continuously evolving numerical methods for PDE, especially for the hyperbolic type conservation laws.

Like all numerical methods, the DG method is not free of disadvantages. Two distinctive huddles of the DG method from a wider spread than it would be, are first due to the extra degree of freedom compared to the continuous Galerkin finite element method, and second the issues of dealing with the elliptic PDE's. The first issue of the extra memory overhead is certainly be a concern for a relatively low-order solutions however once a higher-order solution is being sought, such as $p > n_{dim}$ where p is the degree of the polynomial and n_{dim} the number of spatial dimensions, relative overhead diminishes[14, 15]. In fact, for a very high-order method is employed, such as the spectral element method, the memory overhead penalty quickly approaches comparable to the continuous counterpart.

Application of the DG method towards the elliptic or parabolic problem, which is a prerequisite for solving more challenging problems such as Navier-Stokes equations, is much less obvious compared to the hyperbolic type equations, and it has been an active research topic. A first comprehensive report for the elliptic problem was presented by Arnold et al.[16, 17]. Among others, a few representative method for solving elliptic problem would be the local discontinuous Galerkin method[18], compact discontinuous Galerkin method[19], symmetric interior penalty method[20], and the BR2 method proposed in for solving Navier-Stokes equations[21, 22].

Recently, Nishikawa[1] proposed a novel idea of reformulating the elliptic/parabolic equation into a set of first order hyperbolic system with pseudo-time evolution. This idea may reminds ones the aforementioned DG methods[18, 19, 21, 22] because of the introduction of the additional auxiliary variables for the solution gradients, or fluxes. The idea is, however, fundamentally different in the sense that the hyperbolic formulation starts at the very first of the PDE level and even before the derivation of the weak form or any spatial discretization. Solving the elliptic problem in the hyperbolized form, various kinds of numerical schemes well developed for the hyperbolic PDE can be utilized. The expected benefits of this approach would be, for example, the stable and accurate high-order solutions, efficient implicit time-marching, equal order accuracy of auxiliary gradient variables, slowing growing condition number of the discretized system, and so on. Additional difference of the hyperbolic method is that it does not involve any mesh size dependent parameter, rather it just utilizes a universal relaxation length scale $\mathrm{L_r}$, or time scale $\mathrm{T_r}$, for the fast approach to the steady state solution in pseudo-time. This makes the formulation being general and free of h-dependent tuning parameters, which would be cumbersome for the users to check its sensitivity on the solution.

Since the first introduction of the hyperbolic method, the idea have been applied to various problems including pure diffusion problems[1, 23], convection-diffusion problems[24], and also Navier-Stokes problems[25, 26] mostly in cell-vertex finite volume methods. High-order (>2^{nd}) solution was also attempted in cell-vertex finite volume method[1, 23, 24, 25, 26], cell-centered finite volume method[27], and residual-distribution method[28], but has not been extended to a DG method[29] until recently, the reconstructed discontinuous Galerkin (rDG) method[30-32] for steady and unsteady convection-diffusion equations.

In this paper, we present a very high-order spectral DG method for solving the Poisson problem via. hyperbolic formulation. This is based on the collocation types spectral element method[14, 33] inside of element within the DG framework, i.e. being called as discontinuous Galerkin spectral element method (DGSEM)[34-37]. Since the first introduction of the method by Kopriva and his collaborators[38], the method applied to various problems, including Navier-Stokes equations[38-40] using quadrilaterals and hexahedrons with efficient tensor-product basis function mostly driven by explicit time-stepping schemes. Here, we like to present a fully implicit scheme in pseudo-time for solving the hyperbolically re-formulated Poisson equation within the DGSEM framework, aiming the designed spectral accuracy for both the solution and gradient variables.

The paper is organized in the following contents. In section 2, the hyperbolic formulation of the original Poisson problem in presented in PDE level. In the following section 3, the weak form is derived from the first order hyperbolic system, and DGSEM discretization is explained. In the section 4, two types of test results are presented. First, 1D test cases were presented with emphasis on the implicit boundary condition application technique. Secondly the problem was extended to 2D with regular Uniform quadrilateral grids. For both cases, optimal order of accuracy and the spectral accuracy were confirmed for both the solution and gradient variables. An interesting observation of matrix conditioning is also reported. The findings are summarized in section 5 and conclusions are made.

2. Hyperbolic formulation of diffusion equation

The governing equation for the current study is a Poisson equation in two dimensional space as follows,

$$\begin{aligned} -\nabla\cdot(\upsilon\nabla \mathrm{u}) = f \quad & in \quad \Omega \\ \mathrm{u} = g_D \quad & on \quad \partial\Omega \end{aligned}$$

where u is the solution and υ is the diffusion coefficient, f is the solution independent source, and g_{D} is the solution prescribed on the Dirichlet type boundary. By the idea of Nishikawa[1], auxiliary variables representing the solution gradients are introduced, and the original Poisson equation was transformed into a system of three coupled first-order hyperbolic system (FOHS), as follows

$$\begin{cases} \dfrac{\partial u}{\partial \tau} = \upsilon\left(\dfrac{\partial p}{\partial x} + \dfrac{\partial q}{\partial y}\right) + f \\ \dfrac{\partial p}{\partial \tau} = \dfrac{1}{T_r}\left(\dfrac{\partial u}{\partial x} - p\right) \qquad in \quad \Omega \\ \dfrac{\partial q}{\partial \tau} = \dfrac{1}{T_r}\left(\dfrac{\partial u}{\partial y} - q\right) \end{cases} .$$
$$\mathrm{u} = g_D \qquad on \quad \partial\Omega$$

Here, τ is the artificial time to reach the steady state, T_r is the parameter called relaxation time, which is to be chosen for accelerated convergence to the steady state in the artificial time[23], and $(p, q) = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)$ are the auxiliary variables representing solution gradients introduced to make the original system to be first order. It is obvious that when the solution reaches the steady state at pseudo-time τ, the steady-state solution recovers the solution of the original Poisson equation in the steady state.

If the unsteady solution is necessary, certainly a time-accurate formulation of the first order system can be constructed by, for example, using a dual time-stepping formulation. However, our emphasis is on steady state solution which is also the solution for the original Poisson problem.

The system of equations can be written in a differential vector form as follows,

$$\frac{\partial}{\partial\tau}\boldsymbol{U} + \nabla\cdot\boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}) + \boldsymbol{f}$$

where $\boldsymbol{U}$ is the solution vector, $\boldsymbol{F}(\boldsymbol{U})$ contains the flux vector along all spatial direction, and $\boldsymbol{S}(\boldsymbol{U})$ is the source term depending on the solution, and $\boldsymbol{f}$ is the forcing term of the original Poisson problem. Each of the vectors can be defined in 2D as follows,

$$\boldsymbol{U} = \begin{Bmatrix} u \\ p \\ q \end{Bmatrix},\ \boldsymbol{F} = F\boldsymbol{i} + G\boldsymbol{j} = \begin{Bmatrix} -\nu p \\ -\frac{u}{T_r} \\ 0 \end{Bmatrix}\boldsymbol{i} + \begin{Bmatrix} -\nu q \\ 0 \\ -\frac{u}{T_r} \end{Bmatrix}\boldsymbol{j},\quad \boldsymbol{S} = \begin{Bmatrix} 0 \\ -\frac{p}{T_r} \\ -\frac{q}{T_r} \end{Bmatrix},\quad \boldsymbol{f} = \begin{Bmatrix} f \\ 0 \\ 0 \end{Bmatrix}$$

The hyperbolicity of the above system can be easily confirmed as follows. That differential form of the system can be written in an integral form for an arbitrarily closed domain Ω and its boundary $\partial\Omega$ as follows,

$$\frac{d}{d\tau}\int_{\Omega} \boldsymbol{U}dV + \oint_{\partial\Omega}(\boldsymbol{F}\cdot\boldsymbol{n})dS = \int_{\Omega} \boldsymbol{S}dV + \int_{\Omega} \boldsymbol{f}dV,$$

where the integrand of the second term is the flux term of the hyperbolic formulation, which is defined as

$$(\boldsymbol{F}\cdot\boldsymbol{n}) = (F\boldsymbol{i} + G\boldsymbol{j})\cdot\left(n_x\boldsymbol{i} + n_y\boldsymbol{j}\right)$$

where $\left(n_x, n_y\right)$ is the outward unit normal vector to the boundary $\partial\Omega$, and i and j are unit vectors along x and y coordinate directions. Based on that definition of flux, the flux Jacobian is defined as follows,

$$J = \frac{\partial}{\partial\boldsymbol{U}}(\boldsymbol{F}\cdot\boldsymbol{n}) = \frac{\partial}{\partial\boldsymbol{U}}\begin{Bmatrix} -\nu\left(pn_x + qn_y\right) \\ -un_x/T_r \\ -un_y/T_r \end{Bmatrix} = \begin{bmatrix} 0 & -\nu n_x & -\nu n_y \\ -n_x/T_r & 0 & 0 \\ -n_y/T_r & 0 & 0 \end{bmatrix}$$

whose eigenvalues are found as follows,

$$\lambda_1 = -\sqrt{\nu/T_r},\ \lambda_2 = \sqrt{\nu/T_r},\ \lambda_3 = 0$$

Indeed, all is real and distinct, and results in truly hyperbolic system inherited from the original Poisson equation. Note that the first order hyperbolic system(FOHS) now includes a free parameter T_r to be determined prior to the actual solution. By the analysis of Nishikawa[23], the following time scale $T_r = {L_r}^2/\nu$ and length scale $L_r = 1/2\pi$ are used for the acceleration to the steady state in pseudo-time, which is the ultimate solution. Note that this parameter does not depends on a specific discretization methods or mesh size, and this makes the FOHS to be a general system substituting the original elliptic equation and being ready for various numerical methods for PDE's.

3. Discontinuous Galerkin spectral element method for first order hyperbolic formulation of Poisson problem

In this section the DGSEM formulation of the FOHS is presented and some implementation issues are discussed with focus on memory requirement and sparsity patterns of global stiffness matrix.

3.1. **DGSEM formulation of the hyperbolic diffusion problem**. The starting point of the discontinuous Galerkin method would be the derivation of the weak from the strong form, the FOHS. We assume that the original domain Ω is decomposed into a collection of finite number of non-overlapping subdomains such that

$$\Omega \cong \Omega_h = \cup_{e=1}^{n_{el}} \Omega_e,$$

Where n_{el} is total number of elements, and Ω_e is the e^{th} sub-domain occupied by the element. Consider the following broken Sobolev space $\boldsymbol{\mathcal{V}}_{\boldsymbol{h}}^{\boldsymbol{p}}$,

$$\boldsymbol{\mathcal{V}}_{\boldsymbol{h}}^{\boldsymbol{p}} := \left\{ \mathcal{v}_h \in [L^2(\Omega_h)]^k \quad : \quad \mathcal{v}_h|_{\Omega_I} \in \left[\mathcal{Q}_p^k\right] \quad \forall \Omega_e \in \Omega_h \right\},$$

which is composed of discontinuous k dimensional vector polynomials of degree p, and $\mathcal{Q}_p$ is the set of polynomials of degree $\leq p$. Multiplying a test function $\boldsymbol{V}_h$ to the strong form and integrating over the domain Ω_e followed by the integration by parts results in the following weak form: find $\boldsymbol{U}_h \in \boldsymbol{\mathcal{V}}_{\boldsymbol{h}}^{\boldsymbol{p}}$ such that

$$\frac{d}{d\tau}\int_{\Omega_e} \boldsymbol{V}_h{}^T \boldsymbol{U}_h d\Omega - \int_{\Omega_e} \nabla \boldsymbol{V}_h{}^T \cdot \boldsymbol{F}(\boldsymbol{U}_h) d\Omega + \oint_{\partial\Omega_e} \boldsymbol{V}_h{}^T \boldsymbol{\mathcal{F}}\left(\boldsymbol{U}_h{}^-, \boldsymbol{U}_h{}^+\right) \cdot \boldsymbol{n} d\Gamma =$$

$$\int_{\Omega_e} \boldsymbol{V}_h{}^T \boldsymbol{S}(\boldsymbol{U}_h) d\Omega + \int_{\Omega_e} \boldsymbol{V}_h{}^T \boldsymbol{f} d\Omega \qquad \forall \ \boldsymbol{V}_h \in \ \mathcal{V}_h^p.$$

Note that due to the allowed solution discontinuity between the interior solution $\boldsymbol{U}_h{}^-$ and the exterior $\boldsymbol{U}_h{}^+$ across the element interface, the surface integral term appears and hence a consistent and stable numerical flux has to be provided. For actual construction of the trial and test functions, the collocation type nodal basis functions are utilized. By locating the Lagrange basis functions at the Gauss-Legendre quadrature points along the reference coordinate direction $\xi \in [-1,1]$, a set of discretely orthogonal basis functions with respect to the chosen quadrature rule are constructed as follows

$$\ell_i(\xi) = \prod_{\substack{j=1 \\ j \neq i}}^{N} \frac{(\xi - \xi_j)}{(\xi_i - \xi_j)}, \quad i = 1, \cdots, N.$$

Note that N is number of basis function along each coordinate direction and $= \mathcal{p} + 1$, where $\mathcal{p}$ is the degree of polynomial. The i_{th} basis function located at ξ_i has interpolation property, i.e. $\ell_i(\xi_j) = \delta_{ij}$ for $i, j = 1, \cdots, N$. Hence this method is called as discontinuous Galerkin(DG) across elements and spectral element method(SEM) within each element[33-37].

Once the basis in 1D was constructed, by using quadrilateral elements in 2D, or hexahedrons in 3D, those in multi-dimensions can be simply constructed by the tensor product between each reference coordinate directions. For example in 2D, the basis functions are constructed by the tensor-product of the discretely orthogonal Lagrange polynomials as follows

$$\psi_{I(i,j)} = \ell_i(\xi)\ell_j(\eta).$$

By using the local index map of $id(i,j) = N \times (j-1) + i$, the 2D basis function can be listed in a row vector as follows,

$$\boldsymbol{\psi} = [\psi_{id=1} \quad \cdots \quad \psi_{id=N^2}]$$

The solution points are collocated at the quadrature points are displayed in Fig.1 and whose corresponding shape functions are displayed in Fig.2. The solution points are marked by red-filled markers and flux-points by hollow markers, whose sizes are scaled by their weights of the quadrature rule, i.e. the bigger weight the larger point.

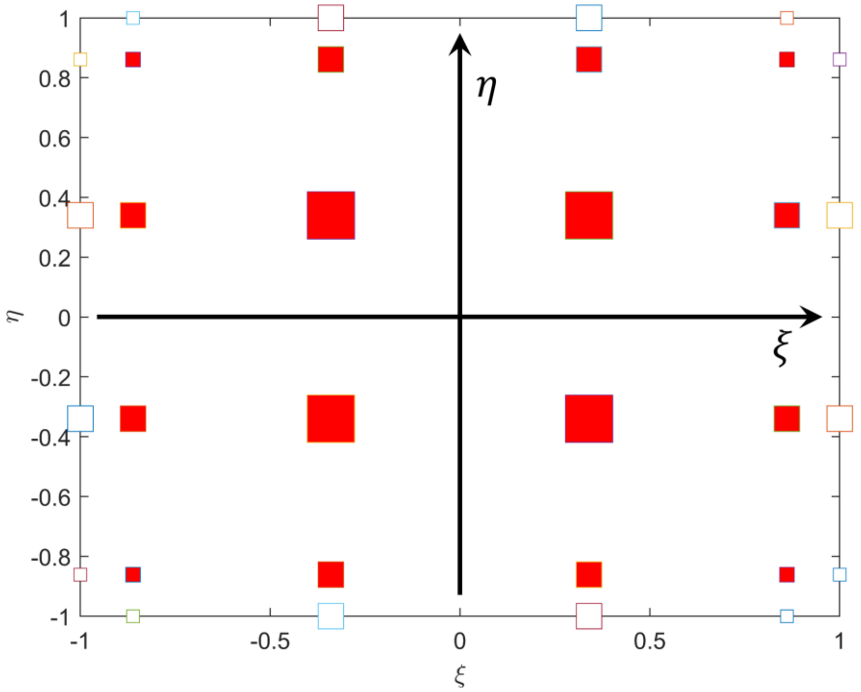


FIGURE 1. Solution and flux point arrangement on the reference domain of a 2D quadrilateral element $(N = 4, \wp = 3)$. Red squares denote the solution point and white hollow squares the location for the flux computation. All markers scaled by relative quadrature weights.

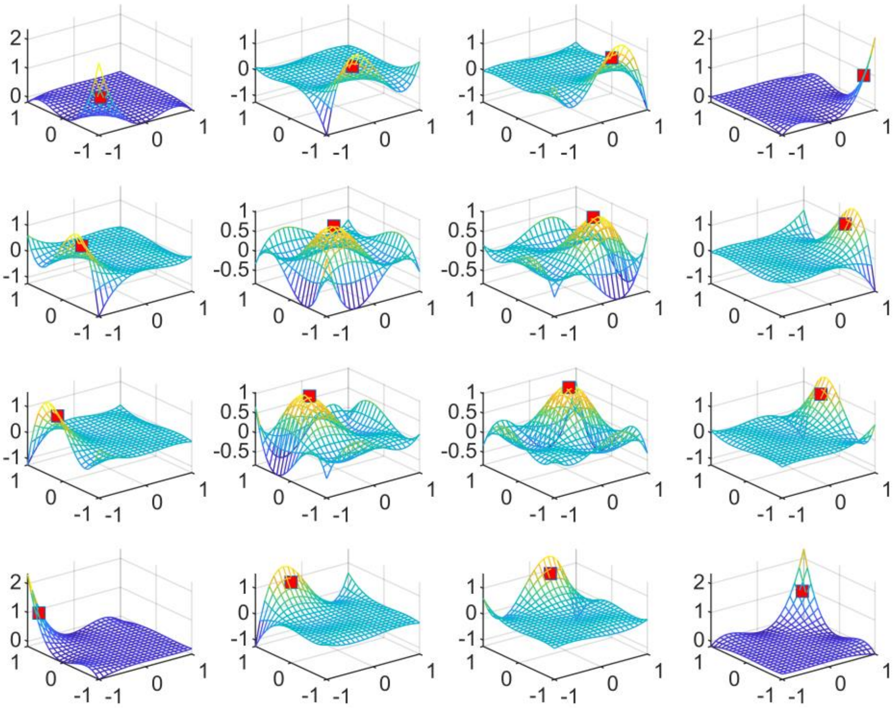


FIGURE 2. Basis functions for $N = 4$ within a reference element. Each interpolate a unit value at the quadrature point.

But the virtue of the flexibility of the DG method, the polynomial order can be different from element to element and also from variable to variable. For the simplicity of the implementation, all element and solution variables are approximation by the same pre-determined solution order at the start of each test case. Once the polynomial orders, or the basis functions are constructed, the approximated solution within an element can be expressed as follows

$$\boldsymbol{U}_h = \boldsymbol{\Psi}\boldsymbol{u}_h = [\boldsymbol{I}_{3\times3}\otimes\boldsymbol{\psi}]\begin{Bmatrix}\boldsymbol{u}\\ \boldsymbol{p}\\ \boldsymbol{q}\end{Bmatrix},$$

where $\boldsymbol{I}_{3\times3}$ is the 3×3 identity matrix and $\otimes$ is the Kronecker product. Since the current number of variables for the FOHS is three (one solution and two gradients), the number of unknown coefficients for each element is $3N^2$ as follows

$$\begin{gathered}\boldsymbol{u} = [u_1 \quad \cdots \quad u_{N^2}]^T\\ \boldsymbol{p} = [p_1 \quad \cdots \quad p_{N^2}]^T.\\ \boldsymbol{q} = [q_1 \quad \cdots \quad q_{N^2}]^T\end{gathered}$$

By following Galerkin's idea, the test function is also chosen to be the same as the trial functions of the solution approximation and can be expressed as follows

$$\boldsymbol{V}_h = \boldsymbol{\Psi} = [\boldsymbol{I}_{3\times3}\otimes\boldsymbol{\psi}]$$

Once the solution and test functions are approximated by the set of basis polynomials, the weak form for a sub-domain Ω_e, i.e. the L^2-orthogonal projection of the residual on the test function space can be expressed as follows,

$$\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{\Psi}d\Omega\frac{d}{d\tau}\boldsymbol{u}^h-\int_{\Omega_e}\left(\boldsymbol{\nabla\Psi}^{\boldsymbol{T}}\right)\cdot\boldsymbol{F}(\boldsymbol{U}_h)d\Omega+\oint_{\partial\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\left(\boldsymbol{\mathcal{F}}\left(\boldsymbol{U}_h{}^-,\boldsymbol{U}_h{}^+\right)\cdot\boldsymbol{n}\right)d\Gamma=\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{S}(\boldsymbol{U}_h)d\Omega+\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{f}d\Omega,$$

Since the continuity requirement of the solution at the element interface is relaxed to be discontinuous, the solution jump is inevitable and this has to be treated by the introduction of a unique numerical flux function $\mathbb{F}$ as follows

$$\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{\Psi}d\Omega\frac{d}{d\tau}\boldsymbol{u}^h-\int_{\Omega_e}\left(\boldsymbol{\nabla\Psi}^{\boldsymbol{T}}\right)\cdot\boldsymbol{\mathcal{F}}(\boldsymbol{U}_h)d\Omega+\oint_{\partial\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\left(\mathbb{F}\left(\boldsymbol{U}_h{}^-,\boldsymbol{U}_h{}^+;\boldsymbol{n}\right)\right)d\Gamma=\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{S}_{\boldsymbol{u}}(\boldsymbol{U}_h)d\Omega+\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{f}d\Omega.$$

Note that the flux function employed here is based on the standard upwind flux which is also used in previous studies[24, 27] and defined as follows,

$$\mathbb{F}\left(\boldsymbol{U}_h{}^-,\boldsymbol{U}_h{}^+;\boldsymbol{n}\right)=\frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{F}\left(\boldsymbol{U}_h{}^-;\boldsymbol{n}\right)+\boldsymbol{F}\left(\boldsymbol{U}_h{}^+;\boldsymbol{n}\right)\right)-\frac{\mathbf{1}}{\mathbf{2}}|\boldsymbol{A}|\left(\boldsymbol{U}_h{}^+-\boldsymbol{U}_h{}^-\right).$$

Since this flux function couples the particular element of interest and only its immediate neighbors sharing faces, i.e. von Neumann neighbors, the current DG method results in indeed a compact scheme. By considering a particular element Ω_I, the actual implementation of the weak form including the interface flux term can be further clarified as follows,

$$\mathbf{M}\frac{d}{d\tau}\boldsymbol{u}_I-\mathbf{G}\boldsymbol{u}_I+\sum_{J=1}^{nei}\left[\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}\left(\partial\mathbb{F}/\partial\boldsymbol{U}_h{}^-\right)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_I+\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}\left(\partial\mathbb{F}/\partial\boldsymbol{U}_h{}^+\right)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_J\right]-\mathbf{S}_{\mathbf{u}}\boldsymbol{u}_I=\ \boldsymbol{f}_I,$$

where $\boldsymbol{u}_I$ and $\boldsymbol{u}_J$ is the unknown coefficients vectors at the cell Ω_I and its neighbor Ω_J respectively, and $\partial\Omega_{I,J}$ refers to interface of $\partial\Omega_I\cap\partial\Omega_J$. Each coefficient matrices evaluated by utilizing the same quadrature rule of N Gauss-Legendre quadrature points along each coordinate direction, which results in exact integration of polynomials degree up to $2N-1=2\wp+1$. This quadrature rule will result in exact mass matrix for constant and linear geometric mappings. There may be slight mass lumping induced error, if a bi-linear or higher-order geometric mapping is used. However in order to keep the simplicity of the implementation, the same quadrature rule was employed for all test cases, which still results in spectral accuracy for all cases.

By the virtue of the orthogonality of the basis, the mass matrix can be conveniently evaluated as follows

$$\mathbf{M}=\int_{\Omega_e}\boldsymbol{\Psi}^{\boldsymbol{T}}\boldsymbol{\Psi}d\Omega=\boldsymbol{I}_{\mathbf{3\times3}}\otimes\mathrm{diag}\left(\omega_i\omega_j\right),$$

and the Galerkin volume integral term can be evaluated as follows,

$$\mathbf{G}=\int_{\Omega_e}\left(\boldsymbol{\nabla\Psi}^{\boldsymbol{T}}\right)\cdot\left(\boldsymbol{\partial F}/\boldsymbol{\partial U}_h\right)\boldsymbol{\Psi}d\Omega=\int_{\Omega_e}\left(\boldsymbol{\partial\Psi}^{\boldsymbol{T}}/\boldsymbol{\partial}x_k\right)\left(\boldsymbol{\partial F}_k/\boldsymbol{\partial U}_h\right)\boldsymbol{\Psi}d\Omega.$$

Note that the sparsity pattern of flux Jacobian $\boldsymbol{\partial F}_k/\boldsymbol{\partial U}_h$, defined as follows

$$\boldsymbol{\partial F}_k/\boldsymbol{\partial U}_h=\begin{bmatrix}0 & -\nu n_1\delta_{k1} & -\nu n_2\delta_{k2}\\ -n_1\delta_{k1}/T_r & 0 & 0\\ -n_2\delta_{k2}/T_r & 0 & 0\end{bmatrix},$$

is directly reflected to that of the diagonal block matrix which is be confirmed in the following section.

The next boundary integral of the flux term is the most important term of most of the DG methods. The evaluation of the flux term produces coupling off-diagonal blocks between the element of interest and the immediate neighbor elements. The stable, consistent, and compact upwind numerical flux was utilized like in previous finite volume method[27] and reconstructed discontinuous Galerkin methods[30, 32], as follows

$$\mathbf{M}\frac{d}{d\tau}\boldsymbol{u}_I \;-\mathbf{G}\boldsymbol{u}_I \;+\sum_{J=1}^{nei}\frac{1}{2}\left[\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}(\boldsymbol{J}+|\boldsymbol{A}|)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_I \;+\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}(\boldsymbol{J}-|\boldsymbol{A}|)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_J\;\right]- \mathbf{S}_{\mathbf{u}}\boldsymbol{u}_I \;=\boldsymbol{f}_I\;,$$

Again the sparsity pattern of the system matrix affected by the sparsity of the flux Jacobian $\boldsymbol{J}$ associated with a particular normal direction as follows,

$$\boldsymbol{J}=\boldsymbol{\partial F}/\boldsymbol{\partial U}^h=\begin{bmatrix}0 & -\nu n_x & -\nu n_y\\ -n_x/T_r & 0 & 0\\ -n_y/T_r & 0 & 0\end{bmatrix},$$

and its absolute Jacobian $|\boldsymbol{A}|$ derived by the eigen-decomposition of $\boldsymbol{J}$ as follows,

$$|\boldsymbol{A}|=R|\Lambda|R^{-1}=\frac{\nu}{L_r}\begin{bmatrix}1 & 0 & 0\\ 0 & {n_x}^2 & n_xn_y\\ 0 & n_yn_x & {n_y}^2\end{bmatrix}.$$

Note that the sparsity pattern of $\boldsymbol{J}$ and $|\boldsymbol{A}|$ are complementary. Hence depending on the surface normal vector, the assembled system matrix would result in the coupling block matrix to be full. Although the formulation is presented with time evolution term in pseudo-time τ, the temporal accuracy means no sense until it researches the steady state. Unless the steady state solution is being sought by explicit time-marking scheme which is easy to be inefficient due to the restrictive timestep of high-order method, the unsteady terms can be neglected and the steady state solution was being sought by the implicit solution process, as follows

$$-\mathbf{G}\boldsymbol{u}_I^h-\mathbf{S}_{\mathbf{u}}\boldsymbol{u}_I^h+\sum_{J=1}^{nei}\frac{1}{2}\left[\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}(\boldsymbol{J}+|\boldsymbol{A}|)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_I^h\right]+\sum_{J=1}^{nei}\frac{1}{2}\left[\int_{\partial\Omega_{I,J}}\boldsymbol{\Psi}^{\boldsymbol{T}}(\boldsymbol{J}-|\boldsymbol{A}|)\boldsymbol{\Psi}d\Gamma\,\boldsymbol{u}_J^h\right]= \boldsymbol{f}_I^h$$

Once the above equation is being assembled into the global system, it results in the global linear system whose solution is the steady state solution and gradients in pseudo-time. A direct solution of the global linear system is sufficient for the steady state solution, because that is the ultimate solution of the original Poisson problem.

4. Results

For the simplicity of the problem and explanation of the boundary condition, the first test case is presented in 1D, as follows

$$\begin{aligned} -u_{xx}=f \quad & in \quad x\in(0,1)\\ u=g_D \quad & on \quad x=0,1\text{ ’}\end{aligned}$$

where the source is given as $f=\pi^2 sin(\pi x)$ and $g_D=0$ which results in the exact solution of $\mathrm{u}(x)=sin(\pi x)$. The original equation was reformulated into a set of hyperbolic PDE as follows,

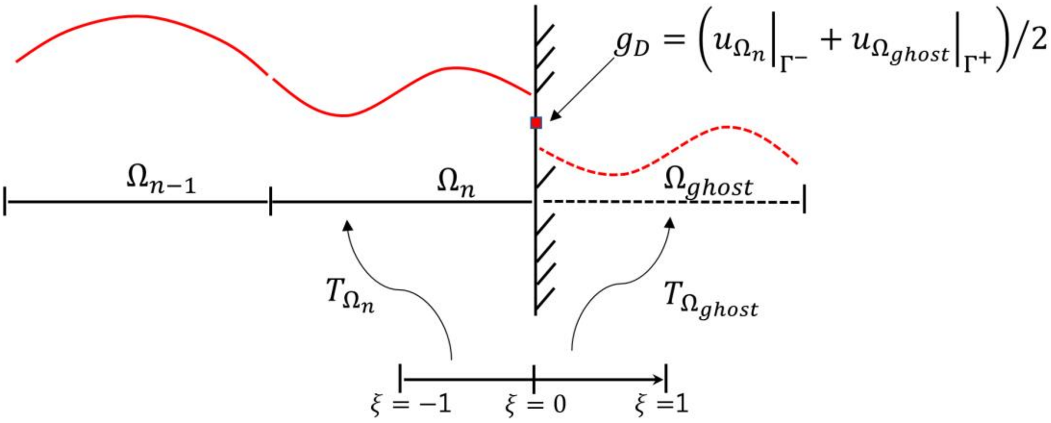


FIGURE 3. Implicit boundary condition imposition using ghost element. Ω_n is the element on boundary and Ω_{ghost} is the ghost element whose solution is constrained to be the mirror image of the boundary element. T_{Ω_n} and $T_{\Omega_{ghost}}$ are the geometric mappings from the reference domain to the boundary and ghost element in physical domain, respectively.

The Dirichlet boundary conditions at the both ends are imposed in a weak sense through the flux computation procedure. On the flux evaluation stage, solution state on both sides of the element interface is necessary, but for the element on boundary however is lack of counterpart element over the boundary. This absent element is constructed by the ghost-element idea as shown in Fig.3. It is direct mirror image of the boundary element Ω_n but its solution is constrained to be the shifted mirror image to that of the boundary element. In this sense, the boundary element acts like the master and the ghost behaves as a slave to the master. In order to achieve this idea, first a ghost element is constructed whose solution order and geometric quantity is same as the boundary cell it itself, then its solution is constrained to be the shifted anti-symmetric image of the solution whose average meets the prescribed Dirichlet condition, as follows

$$\frac{1}{2}\left(u_{\Omega_n}(\xi) + u_{\Omega_{ghost}}(-\xi)\right) = g_D, \tag{4.1}$$

where $\xi \in [-1,1]$ is the coordinate direction in the reference domain, which is normal to the boundary in physical domain. Note that the boundary condition is never to be assigned strongly, rather it is imposed by weakly through the flux computation procedure. The current imposition of the boundary condition is a high-order extension of a implicit boundary condition imposition which described in finite volume method[41].

Hyperbolic system is also being solved for the gradient variable, p, and a proper boundary condition is needed for it as well. Same ghost strategy is utilized for the gradient variables, but with Neumann type boundary condition as follows

$$p_{\Omega_n}(\xi) = p_{\Omega_{ghost}}(-\xi).$$

Note that the above condition is directly derivable by differentiating the Eq. (4.1) along the boundary normal direction, ξ. This condition implies that the solution curvature to be zero in Dirichlet boundary, i.e. being the inflection point if the solution is extended towards the ghost cell. For multi-dimensional case, the same strategy applies for additional gradient variables along the boundary normal direction, which may be either ξ or η of the reference coordinate depending on the element orientation in the physical domain.

Now the above FOHS is discretized by the DGSEM and solved for the solution (u, p).

A sample solution of this 1D problem is displayed in Fig. 4 for two different orders of solution approximation. As shown in the figure, the $\mathscr{p} = 2$ solution shows noticeable discrepancy not only inside of the domain but also on the boundary value, i.e. $u \quad = 0$ on both ends, but such discrepancy disappears at least visually for higher order solution of $\mathscr{p} = 4$.

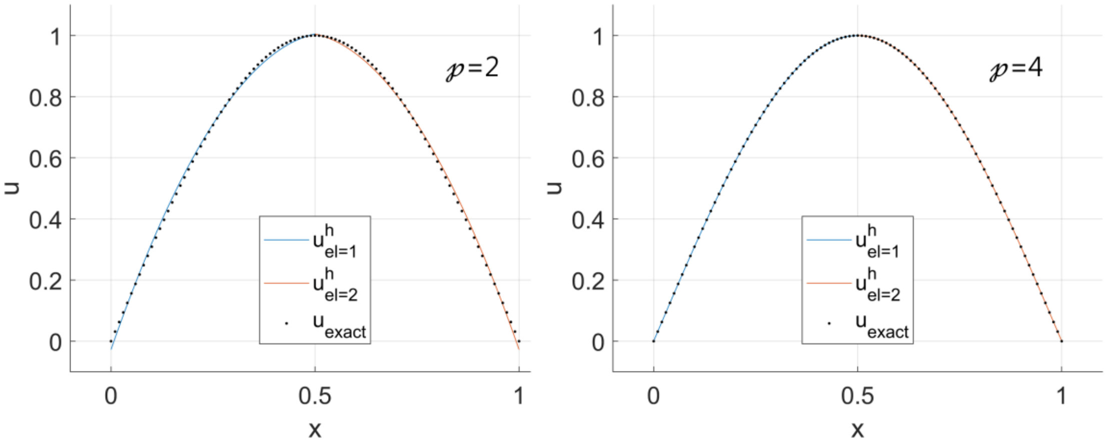


FIGURE 4. Sample solution of the 1D test problem using $n_{el} = 2$. For $\mathscr{p} = 2$, discrepancy between the exact is noticeable within the domain and also on the boundary. For higher order case of $\mathscr{p} = 4$, numerical solution almost interpolate the exact solution even on the boundary.

The results from the $\mathscr{h}$ and $\mathscr{p}$ refinement are displayed in the following Fig. 5. As shown in the figure, the expected order of accuracy is obtained for the degree of polynomials of $\mathscr{p} = 2, \cdots, 9$ results in solution orders of above $N = 3, \cdots, 10$ or above the order of $O(\mathscr{h}^{p+1})$, which is the expected optimal order. It should also be emphasized that such optimal order of accuracy is not only for the solution variable but also for the gradient variable, which is one of many advantages of the hyperbolic formulation.

For $\mathrm{N} > 10$, the polynomial order of accuracy is difficult to measure because a mesh refinement immediately results in the near machine zero error. Instead, the spectral convergence of the error is display by fixing the element number $n_{el} = 2$ but increasing the polynomial orders. As shown in Fig. 6 displayed in semi-log scale error with respect to the solution order, the spectral convergence is clearly observable not only for the solution variable but also for the gradient variable.

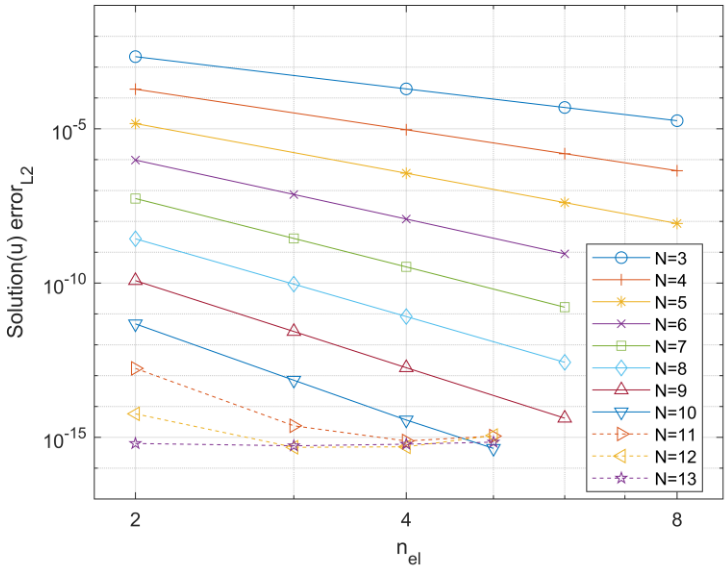
The summary of the verification study is presented in Table 1 and 2. It is clear that the error is decaying in super-convergent manner, i.e. $O(h^{p+1+\alpha})$ with $\alpha > 0$ for all orders of $N = 3,\cdots,10$. For example, the polynomial degree $p = 9$ corresponding to the $N = 10$ case results in the error decaying in the order of $O(h^{10.35})$ for the solution variable and $O(h^{10.45})$ for the gradient variables. Such super-convergence is observed for all orders of approximation for both the solution and the gradient variable simultaneously.

TABLE 1. The L2 error of the solution for different orders of solution approximations. The rate of convergence was estimated by using least-squares fit.

$N = p + 1$	$n_{el} = 2$	$n_{el} = 3$	$n_{el} = 4$	$n_{el} = 6$	$n_{el} = 8$	Rate
3	2.18e-03	–	1.97e-04	4.91e-05	1.83e-05	3.44
4	1.92e-04	–	9.32e-06	1.56e-06	4.39e-07	4.38
5	1.48e-05	–	3.63e-07	4.06e-08	8.53e-09	5.37
6	9.72e-07	7.45e-08	1.19e-08	8.85e-10	–	6.37
7	5.50e-08	2.80e-09	3.34e-10	1.65e-11	–	7.38
8	2.72e-09	9.22e-11	8.24e-12	2.70e-13	–	8.39
9	1.20e-10	2.69e-12	1.80e-13	4.18e-15	–	9.34
10	4.76e-12	7.09e-14	3.64e-15	*4.37e-16*	–	10.35

TABLE 2. The L2 error of the gradient, i.e. $p = \partial u/\partial x$ for different orders of solution approximations. The rate of convergence was estimated by using least-squares fit.

$N = p + 1$	$n_{el} = 2$	$n_{el} = 3$	$n_{el} = 4$	$n_{el} = 6$	$n_{el} = 8$	Rate
3	2.60e-02	–	2.55e-03	6.39e-04	2.37e-04	3.38
4	2.61e-03	–	1.24e-04	2.06e-05	5.72e-06	4.41
5	2.07e-04	–	4.88e-06	5.35e-07	1.11e-07	5.43
6	1.37e-05	1.01e-06	1.59e-07	1.16e-08	–	6.43
7	7.72e-07	3.80e-08	4.46e-09	2.16e-10	–	7.44
8	3.80e-08	1.24e-09	1.09e-10	3.53e-12	–	8.45
9	1.66e-09	3.62e-11	2.38e-12	5.09e-14	–	9.46
10	6.56e-11	9.49e-13	4.70e-14	4.92e-15	–	10.45



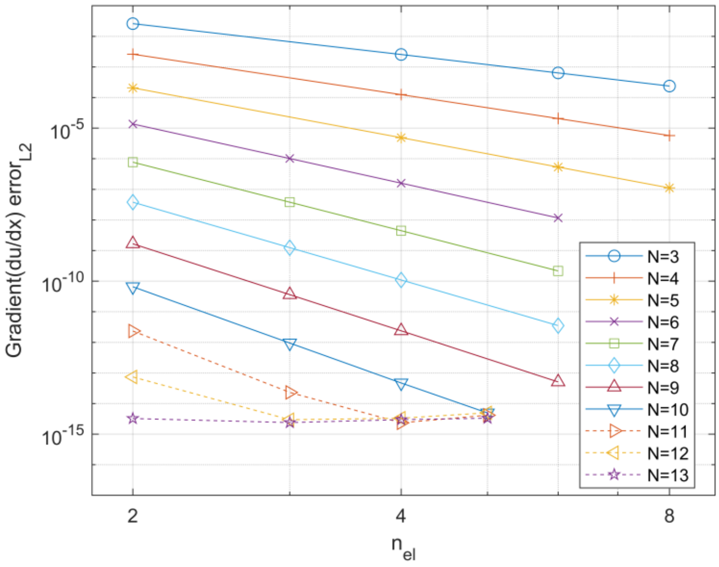


FIGURE 5. Error convergence by elevating the solution orders. Notice that for very high orders, e.g. $N > 10$ indicated by dashed lines, initial mesh refinement results in absolute errors near the machine zero.

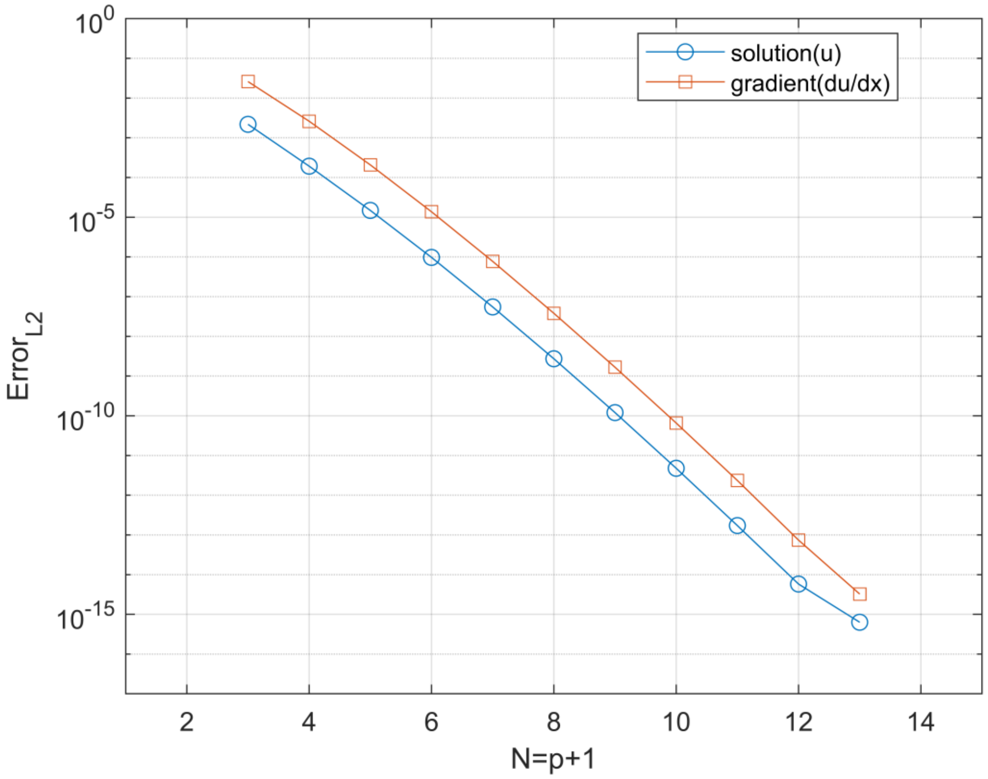


FIGURE 6. Spectral convergence of the solution and a gradient variable. Only $n_{el} = 2$ was used for all orders of solution approximation.

4.2 **Accuracy in 2D and conditioning of stiffness matrix**. In this section Poisson equation in 2D is test as follows,

$$\begin{array}{lll} -\left(u_{xx} + u_{yy}\right) = f & in & x, y \in (0,1)^2 \\ u = g_D & on & x, y = 0,1 \end{array},$$

where the source is given as $f = \pi^2 sin(\pi x) sin(\pi y)$ and $g_D = 0$ which results in the exact solution of $u(x) = sin(\pi x) sin(\pi y)$. As a direct extension from the 1D case, a regular Cartesian grid is considered. Sample solutions of $n_{el} = 2 \times 2$ elements with $N = \wp + 1 = 3$ and 5 are displayed in Fig. 7. As solution order increases, solution continuity across the element interfaces and also the Dirichlet boundary condition, $u = g_D$, on the domain boundaries are improves, and this can be confirmed in the corresponding contour lines as well.

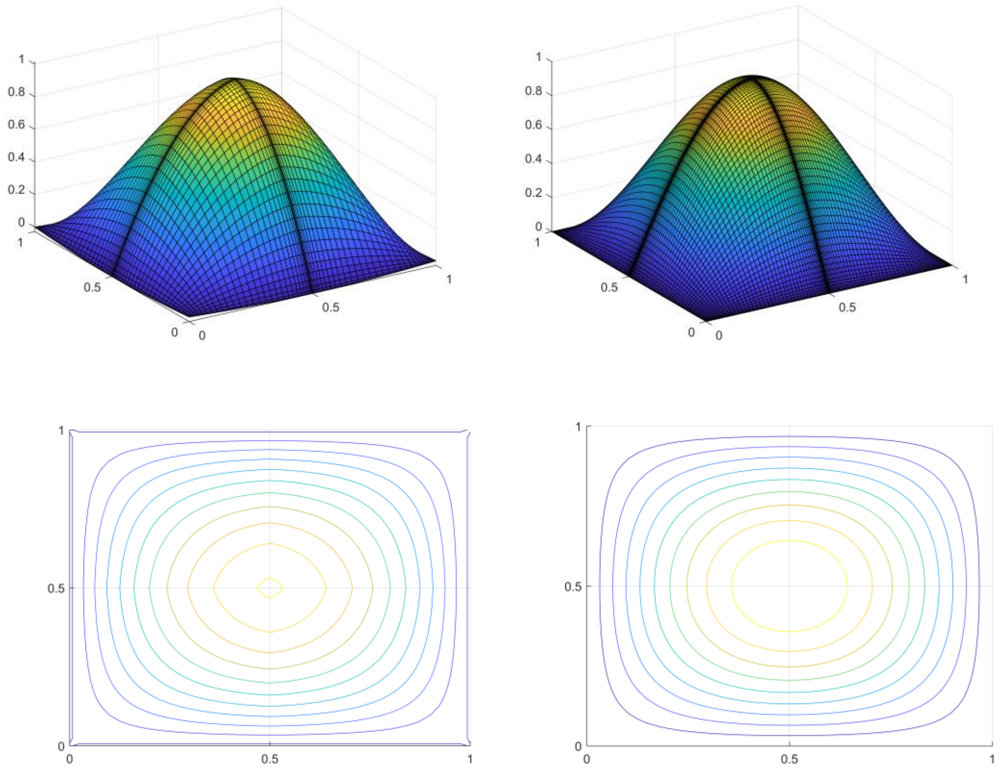


FIGURE 7. Sample solution of 2D test case, with $n_{el} = 2 \times 2$. Left column is for $\mathscr{p} = 2$ solution, and right for $\mathscr{p} = 4$. By increasing the solution order, contour becomes smoother across the elements and also near the boundaries.

The $\mathscr{h}$ and $\mathscr{p}$ refinement study is presented in Fig. 8. As mesh refines, the expected polynomial order of accuracy is obtained, i.e. for degree $\mathscr{p}$ solution approximations, almost $N = \mathscr{p} + 1$ order of accuracy is obtained. This optimal behavior is not only for the solution variables but also for the gradient variables, i.e. p and q. Due to the symmetry of the solution with respect to the $y = x$, convergence histories of p and q towards $\partial u/\partial x$ and $\partial u/\partial y$ are essentially identical, hence only the history of the gradient variable, p, is displayed.

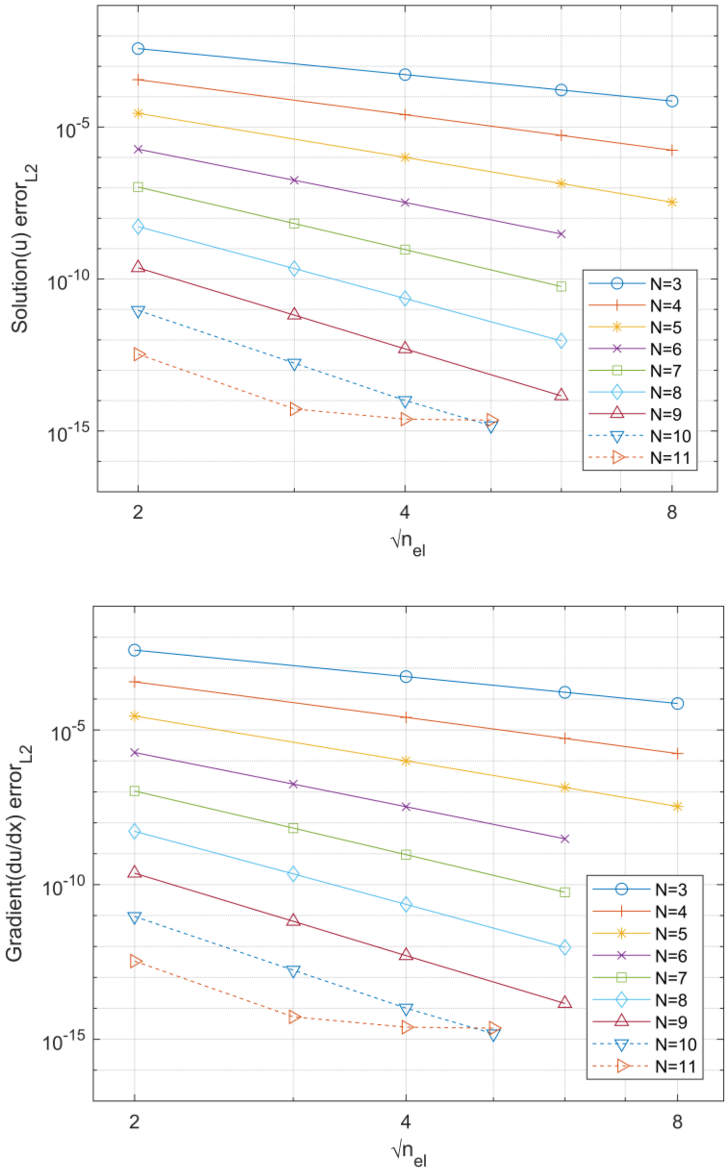


FIGURE 8. Convergence of the error for different solution orders. Optimal order of accuracy was obtained not only for the solution variable but also the gradient variables. The second gradient variable $q = \partial u/\partial y$ was omitted because it shows essential identical convergence history as the first.

For very high order cases, e.g. $N = \wp + 1 = 11$, initial refinement results in the absolute error near the machine zero order and this makes the measuring the polynomial order difficult.

Spectral convergence of the error is also confirmed in Fig. 9 where the error is displayed in log-linear scale with respect to the solution order N. Like in the previous 1D results, the error converges exponentially not only for the solution variables but also the gradient variables. Note that the error of gradient variable of p is essentially identical to that of q.

TABLE 3. The L2 error of the solution variable for different orders of solution approximations. The rate of convergence was estimated by using least-squares fit.

$N = \wp + 1$	$\sqrt{n_{el}} = 2$	$\sqrt{n_{el}} = 3$	$\sqrt{n_{el}} = 4$	$\sqrt{n_{el}} = 6$	$\sqrt{n_{el}} = 8$	Rate
3	3.82e-03	–	5.31e-04	1.65e-04	7.18e-05	2.86
4	3.61e-04	–	2.55e-05	5.32e-06	1.73e-06	3.85
5	2.82e-05	–	1.00e-06	1.38e-07	3.37e-08	4.85
6	1.87e-06	1.78e-07	3.30e-08	3.02e-09	–	5.85
7	1.06e-07	6.72e-09	9.32e-10	5.66e-11	–	6.86
8	5.30e-09	2.21e-10	2.29e-11	9.29e-13	–	7.87
9	2.34e-10	6.50e-12	5.04e-13	1.43e-14	–	8.83
10	9.32e-12	1.71e-13	1.01e-14	–	–	9.84

TABLE 4. The L2 error of the first gradient variable, i.e. $p = \partial u/\partial x$ for different orders of solution approximations. The error of the second gradient variable $q = \partial u/\partial y$, which is essentially identical to the first, was omitted. The rate of convergence was estimated by using least-squares fit.

$N = \wp + 1$	$\sqrt{n_{el}} = 2$	$\sqrt{n_{el}} = 3$	$\sqrt{n_{el}} = 4$	$\sqrt{n_{el}} = 6$	$\sqrt{n_{el}} = 8$	Rate
3	3.69e-02	–	5.19e-03	1.60e-03	6.89e-04	2.87
4	3.73e-03	–	2.55e-04	5.18e-05	1.66e-05	3.90
5	2.98e-04	–	1.00e-05	1.34e-06	3.23e-07	4.92
6	1.97e-05	1.80e-06	3.26e-07	2.92e-08	–	5.93
7	1.11e-06	6.75e-08	9.16e-09	5.44e-10	–	6.94
8	5.50e-08	2.21e-09	2.24e-10	8.89e-12	–	7.95
9	2.40e-09	6.43e-11	4.89e-12	1.30e-13	–	8.95
10	9.49e-11	1.68e-12	9.75e-14	–	–	9.93

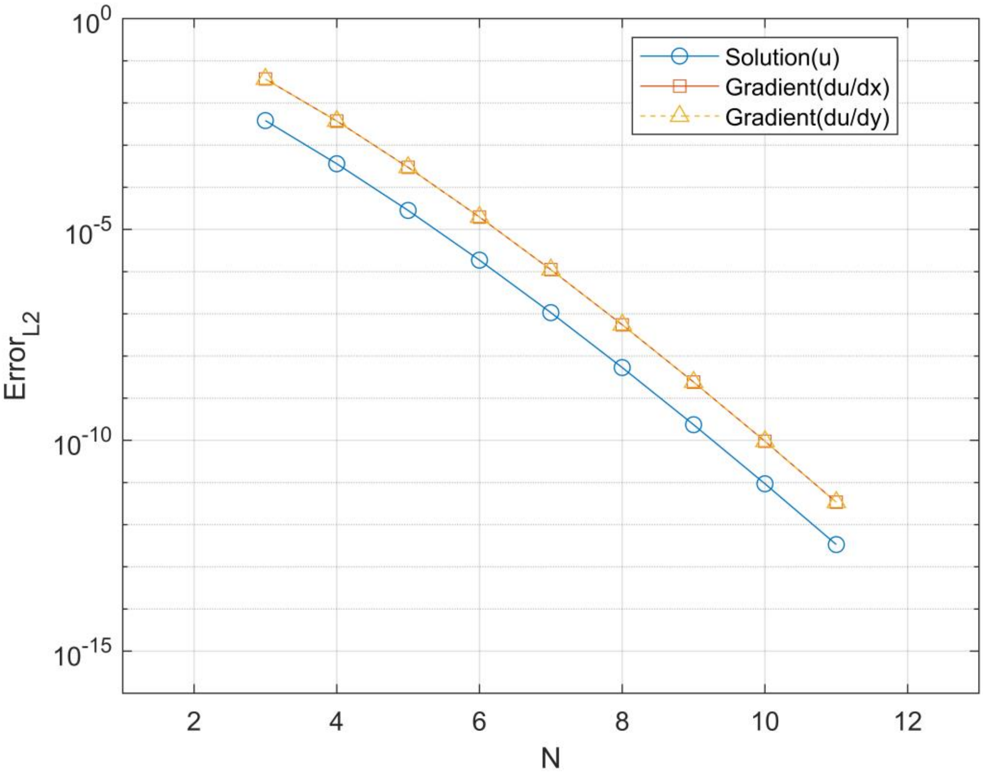


FIGURE 9. Spectral convergence of the solution and gradient variables. Only $\boldsymbol{n_{el} = 2 \times 2}$ was employed for all orders of solution

Another very interesting fact about the current method is about its conditioning of the global stiffness matrix. It is well known that, for grids of a certain degree of uniformity, the spectral condition number of the system grows in the order of $\mathcal{O}(\hbar^{-2})$ where $\hbar$ is the element length scale. In other words, if mesh refines by a half along each direction, then the condition number of the global stiffness matrix grows quadruple, which is almost universal for all discretization methods for Poisson problems. Surprisingly, the conditioning number of the current DGSEM hyperbolic Poisson problem grows only linearly, i.e. in the order of $\mathcal{O}(\hbar^{-1})$. This could be an indication that the current DGSEM can be an efficient method in term of CPU time as well, because the well-conditioned linear system shows faster convergence by iterative linear solvers, especially for large scale problems. Such phenomenon may be attributed to the fact that our problem is truly first order hyperbolic system although it was originated from the second order elliptic problem. Note that a similar characteristic, related to the spectral radius of the stiffness matrix, is already observed in the previous studies in explicit time marching schemes[25], and it was further confirmed in implicit formulation here.

The linear growth of the condition number can be demonstrated by scaling the original matrix condition number, $\kappa(A)$, as follows

$$\kappa^*(A) = \kappa(A) \times \left(\frac{\mathscr{h}}{\mathscr{p}^2}\right) \tag{4.2}$$

Where $\kappa^*(A)$ is the scaled condition number of global stiffness matrix A and $\mathscr{h} = 1/\sqrt{n_{el}}$ is the element size in 2D Cartesian grids and $\mathscr{p}$ is the degree of polynomial, i.e. $N-1$. As summarized in Table 5 and displayed in Fig. 10, once it is scaled the condition number is bounded by a constant even if the mesh is refined. Furthermore, if it is scaled by the square of the polynomial degree, the normalized condition number κ^* decreases monotonically as the solution order is elevated.

TABLE 5. The condition number of the global stiffness matrix, $\kappa^*(A)$, normalized by Eq.(4.2). Notice that the condition number is growing in the order of $O(\mathscr{h}^{-1})$, rather than $O(\mathscr{h}^{-2})$.

$N = \mathscr{p} + 1$	$\sqrt{n_{el}} = 2$	$\sqrt{n_{el}} = 3$	$\sqrt{n_{el}} = 4$	$\sqrt{n_{el}} = 5$	$\sqrt{n_{el}} = 6$	$\sqrt{n_{el}} = 8$
3	26.52	–	29.95	–	30.12	30.10
4	18.29	–	20.85	–	21.15	21.23
5	14.59	–	16.54	–	16.81	16.89
6	12.22	13.38	13.75	–	14.00	–
7	10.62	11.54	11.86	–	12.07	–
8	9.43	10.19	10.46	–	10.64	–
9	8.52	9.15	9.39	–	9.55	–
10	7.79	8.34	8.55	8.64	–	–

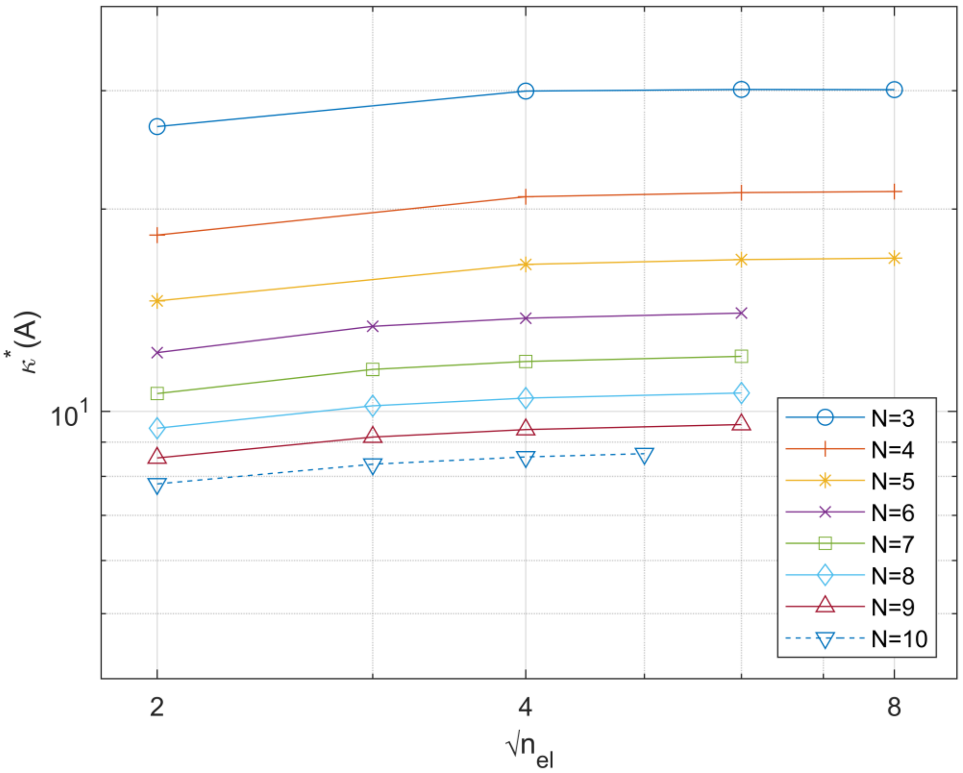


FIGURE 10. Normalized condition number by the element size h and polynomial order p, as shown in Eq. (4.2). Once it is normalized by h/p^2, the matrix conditioning number becomes actually bounded by a constant, which is inverse proportional to the solution order.

5. CONCLUSIONS

Discontinuous Galerkin spectral element method is presented for the elliptic PDE reformulated in the first order hyperbolic system. Optimal order of accuracy $O(h^{p+1})$ is obtained with degree p polynomial for both the solution and gradient variables. The expected spectral convergence was obtained for both the solution and gradient variables. The conditioning of the global stiffness matrix is compared. Surprisingly but expectedly, the matrix condition number grows only linearly with $O(h^{-1})$ in contrast to $O(h^{-2})$ for conventional Poisson solvers, which sheds a light on the faster convergence of the iterative linear solvers. The proposed method is being extended to the convection-diffusion problems and towards more challenging nonlinear problems, such as Navier-Stokes equations for incompressible flows, whose results are to be presented in our subsequent papers. As a conclusion, the DGSEM could be a very attractive method for solving the hyperbolized elliptic problems.

ACKNOWLEDGMENTS

This work was supported by the National Supercomputing Center with supercomputing resources including technical support (KSC-2020-CRE-0119).

REFERENCES

[1] H. Nishikawa, *A first-order system approach for diffusion equation. I: Second-order residual-distribution schemes*, Journal of Computational Physics, 227(1) (2007) 315-352.

[2] W. Reed, T. Hill, *Triangular mesh methods for the neutron transport equation*, Technical Report, Los Alamos Scientific Laboratory/LA-UR-73-479 (1973)

[3] B. Cockburn, C.W. Shu, *Runge–Kutta discontinuous Galerkin methods for convection-dominated problems,* Journal of Scientific Computing 16 (3) (2001), 173-261.

[4] B. Cockburn, C.W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework*, Mathematics of computation 52 (186) (1989), 411-435.

[5] B. Cockburn, S.Y. Lin, C.W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems*, Journal of Computational Physics 84(1) (1989), 90-113.

[6] B. Cockburn, S. Hou, C.W. Shu, *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case*, Mathematics of Computation 54 (190) (1990), 545-581.

[7] B. Cockburn, C.W. Shu, *The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems*, Journal of Computational Physics 141 (2) (1998), 199-224.

[8] D. Xiu, J.S. Hesthaven, *High-order collocation methods for differential equations with random inputs*, SIAM Journal on Scientific Computing 27 (3) (2005), 1118-1139.

[9] J.S. Hesthaven, T. Warburton, *Nodal high-order methods on unstructured grids: I. Time-domain solution of Maxwell's equations*, Journal of Computational Physics 181 (1) (2002) 186-221.

[10] D. Gottlieb, J.S. Hesthaven, *Spectral methods for hyperbolic problems*, Journal of Computational and Applied Mathematics 128 (1-2) (2001), 83-131.

[11] H. Luo, J.D. Baum, R. Löhner, *A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids*, Journal of Computational Physics 227 (20) (2008) 8875-8893.

[12] H. Luo, L. Luo, R. Nourgaliev, V.A. Mousseau, N. Dinh, *A reconstructed discontinuous Galerkin method for the compressible Navier–Stokes equations on arbitrary grids*, Journal of Computational Physics 229 (19) (2010) 6961-6978.

[13] Y. Liu, M. Vinokur, Z.J. Wang, *Spectral difference method for unstructured grids I: basic formulation*, Journal of Computational Physics 216 (2) (2006) 780-801.

[14] J.S. Hesthaven, T. Warburton, *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, Springer Science & Business Media, 2007.

[15] J.S. Hesthaven, *Numerical methods for conservation laws: From analysis to algorithms*, Society for Industrial and Applied Mathematics, 2017.

[16] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, *Discontinuous Galerkin methods for elliptic problems*, Discontinuous Galerkin Methods, Springer, Berlin, Heidelberg, 2000, 89-101.

[17] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM journal on numerical analysis 39 (5) (2002), 1749-1779.

[18] B. Cockburn, C.W. Shu, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM Journal on Numerical Analysis 35 (6) (1998), 2440-2463.

[19] J. Peraire, P.O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, SIAM Journal on Scientific Computation, 30 (4) (2008), 1806-1824.

[20] D. Arnold, *An interior penalty finite element method with discontinuous elements*, SIAM Journal on Numerical Analysis 19(4) (1982) 742-760.

[21] F. Bassi, S. Rebay, *A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations*, Journal of Computational Physics 131 (1997), 267-279.

[22] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, *A high-order accurate discontinuous finite element method fir inviscid an viscous turbomachinery flows*, 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics, Technologisch Instituut, Antwerpen, Belgium, 1997. 99-109.

[23] H. Nishikawa, *First-, second-, and third-order finite-volume schemes for diffusion*, Journal of Computational Physics 256 (2014) 791–805.

[24] H. Nishikawa, *First-, second-, and third order finite-volume schemes for advection‒diffusion*, Journal of Computational Physics 273 (2014) 287‒309.

[25] H. Nishikawa, *New-Generation hyperbolic Navier-Stokes schemes: O(1/h) speed-up and accurate viscous/heat fluxes,* 20th AIAA Computational Fluid Dynamics Conference, Honolulu, Hawaii, USA, 2011, 2011-3043.

[26] Y. Nakashima, N. Watanabe, H. Nishikawa, *Hyperbolic Navier–Stokes solver for three-dimensional flows*, 54th AIAA Aerospace Science Meeting, San Diego, California, USA, 2016, 2016-1101.

[27] E. Lee, H.T. Ahn, H. Luo, *Cell-centered high-order hyperbolic finite volume method for diffusion equation on unstructured grids,* Journal of Computational Physics 335 (2018) 464-491.

[28] H. Nishikawa, A first-order system approach for diffusion equation. II: Unification of advection and diffusion, Journal of Computational Physics, 229(11) (2010) 3989-4019

[29] A. Mazaheri, H. Nishikawa, First-Order Hyperbolic System Method for Time-Dependent Advection-Diffusion Problems, NASA Technical Reports, NASA/TM-2014-218175 (2014)

[30] J. Lou, L. Li, H. Luo, H. Nishikawa, *Reconstructed discontinuous Galerkin methods for linear advection–diffusion equations based on first-order hyperbolic system*, Journal of Computational Physics 369 (2018) 103-124.

[31] L. Li, J. Lou, H. Luo, H. Nishikawa, *High-order hyperbolic Navier-Stokes reconstructed discontinuous Galerkin method*, Scitech 2019 Forum, San Diego, California, USA, 2019, 2019-1150

[32] L. Li, J. Lou, H. Luo, H. Nishikawa, *High-order hyperbolic Navier-Stokes reconstructed discontinuous Galerkin method for unsteady flows*, AIAA Aviation 2019 Forum, Dallas, Texas, USA, 2019, 2019-3060

[33] D.A. Kopriva, *Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers*, Springer Science & Business Media, 2009.

[34] D.A. Kopriva, J.H. Kolias, *A conservative staggered-grid Chebyshev multidomain method for compressible flows*, Journal of Computational Physics 125 (1) (1996) 244-261.

[35] D.A. Kopriva, *A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method*, Journal of Computational Physics 128 (2) (1996) 475-488.

[36] D.A. Kopriva, *A staggered-grid multidomain spectral method for the compressible Navier–Stokes equations*, Journal of Computational Physics 143 (1) (1998) 125-158.

[37] D.A. Kopriva, G.J. Gassner, *On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods*, SIAM Journal on Scientific Computation. 44 (2) (2010) 136-155.

[38] D.A. Kopriva, G.J. Gassner, *An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems*, SIAM Journal on Scientific Computation. 36 (4) (2014) A2076-A2099.

[39] G.J. Gassner, A.R. Winters, D.A. Kopriva, *Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations*, J. Comput. Phys. 327 (2016) 39-66.

[40] F. Hindenlang, G.J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C.D. Munz, *Explicit discontinuous Galerkin methods for unsteady problems*, Comput. Fluids 61 (2012) 86-93.

[41] H. Luo, J. Baum, R. Lohner, J. Cabello, *Implicit schemes and boundary conditions for compressible flows on unstructured grids,* 32nd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA, 1994, 1994-816.