Isı Bilimi ve Tekniği Dergisi, 42, 1, 91-100, 2022
J. of Thermal Science and Technology

ISSN 1300-3615
https://doi.org/10.47480/isibted.1107459

AN ACCELERATED NODAL DISCONTINUOUS GALERKIN METHOD FOR THERMAL CONVECTION ON UNSTRUCTURED MESHES: FORMULATION AND VALIDATION

Ali KARAKUS
Department of Mechanical Engineering, Middle East Technical University
06800 Cankaya Ankara
akarakus@metu.edu.tr, ORCID: 0000-0002-9659-6712

(Geliş Tarihi: 04.10.2021, Kabul Tarihi: 04.04.2022)

Abstract: We present a GPU-accelerated method for large scale, coupled incompressible fluid flow and heat transfer problems. A high-order, nodal discontinuous Galerkin method is utilized to discretize governing equations on unstructured triangular meshes. A semi-implicit scheme with explicit treatment of the advective terms and implicit treatment of the split Stokes operators are used for time discretization. The pressure system is solved with a conjugate gradient method together with a fully GPU-accelerated multigrid preconditioner. The code is built on scalable libParanumal solver which is a library of high-performance kernels for high-order discretizations. Performance portability is achieved by using the open concurrent compute abstraction, OCCA. A set of numerical experiments including free and mixed convection problems indicate that our approach experimentally reaches design order of accuracy.
Keywords: discontinuous Galerkin, GPU, parallel, incompressible, heat transfer, high-order.

ISI TAŞINIMI İÇİN YAPISAL OLMAYAN AĞLARDA BİR HIZLANDIRILMIŞ SÜREKSİZ GALERKİN METDU: FORMÜLASYONU VE DOĞRULANMASI

Öz: Bu çalışmada, büyük ölçekli, birleştirilmiş sıkıştırılamaz akış ve ısı transferi problemleri için GPU ile hızlandırılmış bir yöntem sunulmuştur. Yapılandırılmamış üçgen ağlar üzerinde tanımlayıcı denklemlerini ayrıklaştırmak için yüksek dereceli, nodal süreksiz Galerkin yöntemi kullanılmıştır. Zaman ayrıklaştırması, taşınım terimlerinin açık bir şekilde ele alındığı, Stokes operatörlerinin ise örtük olarak çözüldüğü bir yarı-örtük şema kullanılarak elde edilmiştir. Basınç sistemi, tamamen GPU ile hızlandırılmış bir çoklu-ağ ön koşullandırıcının kullanıldığı, eşlenik gradyan yöntemiyle çözülmüştür. Kod, yüksek dereceli ayrıklaştırmalar için yüksek performanslı kerneller sağlayan, ölçeklenebilir kütüphane olan libParanumal üzerine yazılmıştır. Platformdan bağımsız performans taşınabilirliği OCCA (open concurrent compute abstraction) dili ile elde edilmiştir. Yapılan serbest ve karışık taşınım problemlerini içeren bir dizi sayısal test ile sunulan metodun deneysel olarak beklenen, spektral doğruluğa ulaştığını gösterilmiştir.
Anahtar Kelimler: süreksiz Galerkin, GPU, paralel, sıkıştırılamaz akış, ısı transferi, yüksek seviyeli hesaplama.

NOMANCLATURE

Abbreviations

Gr	Grashof Number $\left[= g\beta(T - T_r){L_r}^3/\nu^2\right]$
Pr	Prandtl Number $[= \nu/\alpha]$
Ra	Rayleigh Number $[= GrPr]$
Re	Reynolds Number $[= U_r L_r/\nu]$
Ri	Richardson Number $[= Gr/Re^2]$
St	Strouhal Number $[= fL_r/U_r]$
DG	Discontinuous Galerkin
DNS	Direct Numerical Simulation
GPU	Graphical Processing Unit
HPC	High Performance Computing
NS	Navier-Stokes
SIPDG	Symmetric Interior Penalty DG

Subscripts

u	Velocity Related Value
θ	Temperature Related Value
D	Dirichlet Boundary
N	Neumann Boundary
r	Reference Value

Greek Symbols

α	Thermal Diffusivity $[= m^2/s]$
β	Expansion Coefficient $[= 1/^\circ C]$
$\nabla \cdot$	Divergence Operator
∇	Gradient Operator
Δ	Laplace Operator
ν	Kinematic Viscosity $[= m^2/s]$
τ	Stabilization Parameter
θ	Non-dimensional Temperature

Superscripts

e	Element Id

ex	Exact Solution
f	Face Id
$n+1$	Next Time Level
n	Current Time Level
Symbols	
s_u	Source Term of Momentum Equation
u	Velocity Vector
g_D	Dirichlet Boundary Condition
g_N	Neumann Boundary Condition
h	Characteristics Mesh Size
K	Number of Elements
L_2	L_2 Vector Norm of Error
N	Approximation Order
N_p	Number of Interpolation Nodes
p	Non-dimensional Pressure
s_θ	Source Term of Energy Equation
t	Non-dimensional Time

INTRODUCTION

One of the primary challenges in fluid mechanics and heat transfer is to reflect physical interactions that occurred in a large range of spatial and temporal scales. This type of analysis for real-life applications might require extreme degrees of freedom and high-order discretizations to minimize numerical dissipation. Simulation tools developed for these problems can require large compute times which cannot be provided even by the fastest supercomputers. The principal objective of high-performance computing (HPC) is to bring the extreme runtimes to reasonable levels. Recently, HPC systems use accelerator based on-node parallelism using Graphical Processing Units (GPUs). On the other hand, developing a high-order finite element based flow and heat transfer analysis suit that benefits modern accelerators is complicated due to the need of achieving multi-threaded parallelism with effectively exploiting nonuniform memory hierarchies (Karakus et al., 2019b).

Discontinuous Galerkin (DG) method (we refer Hesthaven and Warburton (2008) and the references therein) is a class of high-order finite element methods. DG method uses completely discontinuous, piecewise polynomial approximations for spatial discretization that leads to weak elemental connectivity and local stencil. These properties of the DG method together with high-order approximations yields highly parallel operators with high arithmetic intensity that makes the method well-suited for GPUs. Because of its attractive properties, DG method has been used to analyze various physical problems (Gandham et al., 2015; Modave et al., 2016; Chan et al., 2016; Karakus et al., 2016a,b, 2019a). The implementation of DG methods on GPUs is well documented for first-order hyperbolic systems with explicit time integrators, but only a few of papers report GPU-optimized DG discretizations for incompressible flows (Roca et al., 2011) or the operators required for incompressible flows (Swirydowicz et al., 2019). In Karakus et al. (2019b), we recently proposed a GPU-optimized nodal DG method for incompressible flows on unstructured meshes. We extend this efficient approach to thermal convection problems in the present study.

There have been only a couple of studies dealing with incompressible thermal convection using discontinuous or continuous high-order finite element methods. Hossain et al. (Hossain et al., 2021) developed a spectral/hp element method for the Direct Numerical Simulation (DNS) of incompressible thermal convective flows by considering Boussinesq type thermal body-forcing with periodic boundary conditions and enforcing a constant volumetric flow rate. In Kumar and Pothérat (2020), the authors studied the convective patterns that arise in a nearly semi-cylindrical cavity placed in a fluid heated at the upper boundary and bounded by a cold and porous semicircular boundary. They used spectral element method to obtain results and performed a linear stability analysis of the fluid flow. In Saha et al. (2015), a high-order spectral element model is developed for forced thermal convection problems considering incompressible flows and avoiding thermal body-forcing effects.

In this work, we focus on a GPU accelerated, high-order discontinuous Galerkin (DG) based approach for coupled fluid flow and heat transfer problems which is suitable for large scale engineering simulations. In particular, we deal with the formulation of presented method and validation of developed solver for different thermal convection regimes. To the best of our knowledge, there has been no published research on incompressible thermal convection that fully benefits high-order, nodal DG discretization and modern computational architectures relying on-node parallelization.

The remainder of this paper is organized as follows. In the first section, we present the mathematical formulation for the DG scheme to approximate the incompressible Navier-Stokes equations coupled with energy equations, including the spatial discretizations and the temporal splitting scheme. Then, linear solvers and GPU-accelerated p-multigrid preconditioner are briefly explained which is followed by numerical validation test cases. Final section is dedicated to concluding remarks and comments on the future works.

FORMULATION

We consider a closed two-dimensional domain $\Omega \subset R^2$ and denote the boundary of Ω by $\partial\Omega$. Following the notation presented in Karakus et al. (2019b), we assume that $\partial\Omega$ can be partitioned into two non-overlapping regions denoted by $\partial\Omega_D$ and $\partial\Omega_N$ referring prescribed Dirichlet or Neumann boundary conditions, respectively.

We are interested in the approximation of non-isothermal incompressible Navier-Stokes equations coupled by the energy equations through Boussinesq approximation which reads

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1.1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \frac{1}{Re}\Delta \boldsymbol{u} + \boldsymbol{s_u}, \tag{1.2}$$

$$\frac{\partial \theta}{\partial t} + (\boldsymbol{u} \cdot \nabla)\theta = \frac{1}{Re\, Pr}\Delta\theta + s_\theta, \tag{1.3}$$

in non-dimensional form and space-time domain $\Omega \times (0, \mathcal{T}]$ subject to the initial conditions

$$\boldsymbol{u} = \boldsymbol{u_0}, \theta = \theta_0 \text{ for } t = 0, \boldsymbol{x} \in \Omega, \tag{2}$$

and the boundary conditions

$$\boldsymbol{u} = \boldsymbol{g_D} \text{ on } \boldsymbol{x} \in \partial\Omega_D^u, t \in (0, \mathcal{T}], \tag{3.1}$$

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} = 0, p = 0 \text{ on } \boldsymbol{x} \in \partial\Omega_N^{\boldsymbol{u}}, t \in (0, \mathcal{T}], \tag{3.2}$$

$$\theta = g_D \text{ on } \boldsymbol{x} \in \partial\Omega_D^\theta, t \in (0, \mathcal{T}], \tag{3.3}$$

$$\frac{\partial \theta}{\partial n} = g_N \text{ on } \boldsymbol{x} \in \partial\Omega_N^\theta, t \in (0, \mathcal{T}]. \tag{3.4}$$

Here u, p and θ are non-dimensional velocity, static pressure and temperature fields, respectively. In the equation, following parameters are used to get dimensionless quantities,

$$\boldsymbol{x} = \frac{\boldsymbol{x}^*}{L_r}, t = \frac{t^*}{L_r/U_r}, \boldsymbol{u} = \frac{\boldsymbol{u}^*}{U_r}, p = \frac{p^*}{\rho_r U_r^2}$$
$$\rho = \frac{\rho^*}{\rho_r}, \nu = \frac{\nu^*}{\nu_r}, \alpha = \frac{\alpha^*}{\alpha_r}, \theta = \frac{T - T_r}{T_s}, \tag{4}$$

where superscript $*$ denotes the dimensional parameter and the subscript r refers to corresponding reference value i.e. reference length scale L_r, velocity U_r, density ρ_r, viscosity ν_r, thermal diffusivity α_r and temperature T_r. One should note that the temperature scale T_s is problem dependent i.e. in the case of a channel flow with uniformly heated walls, T_s can be defined as the maximum temperature difference between the hot and cold walls in the system.

The non-dimensional Reynolds and Prandtl numbers in the eq. (1) are defined as $Re = U_r L_r/\nu_r$ and $Pr = \nu_r/\alpha_r$. Also, $\boldsymbol{s_u} = (\boldsymbol{g}\beta(T - T_r)L_r/U_r)\theta$ is the forcing term for Navier-Stokes, where $\boldsymbol{g}$ is the gravitational acceleration, β is the expansion coefficient. In free convection problems, the reference velocity is selected as $U_r = g\beta(T - T_r)L_r$. $s_\theta = s_\theta(\theta, \nabla\theta, \boldsymbol{u})$ is the generic generation term for the energy equation written in terms of temperature. We would like to emphasize that superscripts u and θ in boundary representation separate the Dirichlet and Neumann conditions on the physical boundary set for flow and heat transfer equations.

The governing PDE system is discretized by first constructing the spatial discretization using the nodal DG method, then by the temporal discretization using a pressure correction scheme that are covered in the following sections.

Preliminaries

The computational domain Ω is partitioned into K triangular elements $\mathcal{E}^e$, $e = 1, \ldots, K$, such that

$$\Omega = \cup_{e=1}^{K} \mathcal{E}^e$$

The boundary of the element $\mathcal{E}^e$ is denoted by $\partial\mathcal{E}^e$. Two elements, $\mathcal{E}^{e+}$ and $\mathcal{E}^{e-}$, are neighbours if they have a common face, i.e., $\partial\mathcal{E}^{e-} \cap \partial\mathcal{E}^{e+} \neq \emptyset$. The unit outward normal vector of $\partial\mathcal{E}$ is showed as n.

We consider a finite element space on each element $\mathcal{E}^e$, denoted $V_N^e = \mathcal{P}_{\mathcal{N}}(\mathcal{E}^e)$ where $\mathcal{P}_{\mathcal{N}}(\mathcal{E}^e)$ is the space of polynomial functions of degree N on element $\mathcal{E}^e$. As a basis of the finite element spaces, we take a set of $N_p = |V_n^e|$ Lagrange polynomials $\{l_n^e\}_{n=0}^{n=N_p}$, interpolating at the Warp & Blend nodes (Warburton, 2006) mapped to the element $\mathcal{E}^e$.

We have introduced the inner product $(u, v)_{\mathcal{E}^e}$ to denote the integration of the product of arbitrary variables u and v computed over the element $\mathcal{E}^e$ and the inner product $(u, v)_{\partial\mathcal{E}^e}$ to denote the integration along the element boundary $\partial\mathcal{E}^e$. We also define internal and external trace values of a scalar-valued function u^e as u^- and u^+, respectively. Note that we suppress the use of the e superscript when it is clear which element is the local trace. According to this definition, average and jump operators can be defined as:

$$\{\{u\}\} = \frac{u^+ + u^-}{2}, \quad [[u^e]] = u^+ - u^-. \tag{5}$$

When u is a vector-valued function, the above operators act on it component wise.

Spatial Discretization

The polynomial approximation of the velocity field u, the pressure p and the temperature θ fields are defined on each element as

$$\boldsymbol{u}^e = \sum_{n=0}^{N_p} \boldsymbol{u}_n^e\, l_n^e(\boldsymbol{x}), p^e = \sum_{n=0}^{N_p} p_n^e\, l_n^e(\boldsymbol{x}), \theta^e = \sum_{n=0}^{N_p} \theta_n^e\, l_n^e(\boldsymbol{x}),$$

for all $\boldsymbol{x} = (x, y) \in \mathcal{E}^e$. Then, using the polynomial representation of u^e, p^e and θ^e, the semi-discrete form of PDE system given by the eq. (1) is defined on each element $\mathcal{E}^e$ as

$$D^e \boldsymbol{u}^e = 0 \tag{6.1}$$

$$\frac{d\boldsymbol{u}^e}{dt} + \boldsymbol{N}^e(\boldsymbol{u}^e) = \boldsymbol{G}^e p^e + \frac{1}{Re} L^e \boldsymbol{u}^e + \boldsymbol{S}_{\boldsymbol{u}}^e \tag{6.2}$$

$$\frac{d\theta^e}{dt} + N^e(\theta^e) = \frac{1}{RePr} L^e \theta^e + S_\theta^e \tag{6.3}$$

where the operators are defined as follow, $\boldsymbol{N}^e: (V_N^e)^2 \to (V_N^e)^2$, $N^e: (V_N^e) \to (V_N^e)$, $\boldsymbol{S}_{\boldsymbol{u}}^e: (V_N^e)^2 \to (V_N^e)^2$, $S_\theta^e: (V_N^e) \to (V_N^e)$, $\boldsymbol{G}^e: V_N^e \to (V_N^e)^2$, $L^e: V_N^e \to V_N^e$ and $D^e: (V_N^e)^2 \to V_N^e$. These terms are discrete versions of the nonlinear term $\boldsymbol{u} \cdot \nabla \boldsymbol{u}$, advection term $\boldsymbol{u} \cdot \nabla \theta$, source terms s_u and s_θ, gradient operator ∇, Laplacian Δ, and the divergence operator $\nabla \cdot$, in the same order. In the subsequent parts, we define these operators in the nodal DG framework.

We begin with the discretization of nonlinear term, $\boldsymbol{u} \cdot \nabla \boldsymbol{u}$ and the advection term $\boldsymbol{u} \cdot \nabla \theta$. Using the incompressibility condition eq. (1.1), the advection terms are written in divergence form i.e., $\boldsymbol{u} \cdot \nabla \boldsymbol{u} = \nabla \cdot \boldsymbol{F}(\boldsymbol{u})$ and $\boldsymbol{u} \cdot \nabla \theta = \nabla \cdot F(\boldsymbol{u}, \theta)$, where $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{u} \otimes \boldsymbol{u}$ and $F(\theta, \boldsymbol{u}) = \boldsymbol{u}\theta$. Multiplying $\boldsymbol{u} \cdot \nabla \boldsymbol{u}$ and $\boldsymbol{u} \cdot \nabla \theta$ by a test function $v \in V_N^e$, integrating over the element $\mathcal{E}^e$, and performing integration by parts, we define the discrete advective terms $\boldsymbol{N}^e(\boldsymbol{u})$ and $N^e(\theta, \boldsymbol{u})$ via the following variational forms

$$\left(v, \boldsymbol{N}^e(\boldsymbol{u}^e)\right)_{\mathcal{E}^e} = -\left(\nabla v, \boldsymbol{F}(\boldsymbol{u}^e)\right)_{\mathcal{E}^e} + (v, \boldsymbol{n} \cdot \boldsymbol{F}^*)_{\partial \mathcal{E}^e},$$
$$\left(v, N^e(\theta^e, \boldsymbol{u}^e)\right)_{\mathcal{E}^e} = -\left(\nabla v, F(\theta^e, \boldsymbol{u}^e)\right)_{\mathcal{E}^e} + (v, \boldsymbol{n} \cdot F^*)_{\partial \mathcal{E}^e}$$

Since DG uses discontinuous approximation space, the flux functions $\boldsymbol{F}$ and F are not uniquely defined on the element boundaries and hence, they are replaced by local Lax-Friedrichs numerical flux functions $\boldsymbol{F}^*$ and F^* which depend on the local and neighboring trace values,

$$\boldsymbol{F}^* = \{\{\boldsymbol{F}(\boldsymbol{u})\}\} + \frac{1}{2} \boldsymbol{n} \Lambda_u^e [[\boldsymbol{u}]], \tag{7.1}$$

$$F^* = \{\{F(\boldsymbol{u}\theta)\}\} + \frac{1}{2} \Lambda_\theta^e [[\theta]], \tag{7.2}$$

The parameters Λ in the eq. (7) introduce artificial diffusion required to stabilize the numerical discretization of advective terms and are defined to be the maximum eigenvalue of the flux Jacobians in absolute value, i.e.,

$$\Lambda_u = \max_{u \in [u^-, u^+]} \left| \boldsymbol{n} \cdot \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}} \right|, \; \Lambda_\theta = \max_{\theta \in [\theta^-, \theta^+]} \left| \frac{\partial F}{\partial \theta} \right|.$$

Boundary conditions are enforced weakly in flux functions by adjusting exterior trace values i.e. in the case of Dirichlet boundaries $\partial \mathcal{E}^e \cap \Omega_D \neq \emptyset$, the boundary conditions, eq. (3.1) and (3.3) lead $\boldsymbol{u}^+ = \boldsymbol{g}_D$ and $\theta^+ = g_D$. For Neumann boundaries eq. (3.2) and (3.4) where $\partial \mathcal{E}^e \cap \Omega_N \neq \emptyset$, we simply choose $\boldsymbol{u}^+ = \boldsymbol{u}^-$ and $\theta^+ = \theta^-$.

To obtain discrete gradient and divergence operators, $\boldsymbol{G}^e$ and D^e, we multiply the pressure gradient ∇p^e and the velocity divergence $\nabla \cdot \boldsymbol{u}^e$ by a test function $v \in V_N^e$, integrate over the element $\mathcal{E}^e$. Different from the advection operators, we use integration by parts twice to get strong variational forms and simply utilize central fluxes $p^* = \{\{p\}\}$ and $\boldsymbol{u}^* = \{\{\boldsymbol{u}\}\}$ leading

$$(v, \boldsymbol{G}^e p^e)_{\mathcal{E}^e} = (v, \nabla p^e)_{\mathcal{E}^e} + \left(v, \boldsymbol{n}(p^* - p^-)\right)_{\partial \mathcal{E}^e}, \tag{8}$$

$$(v, D^e \boldsymbol{u}^e)_{\mathcal{E}^e} = (v, \nabla \cdot \boldsymbol{u}^e)_{\mathcal{E}^e} + \left(v, \boldsymbol{n} \cdot (\boldsymbol{u}^* - \boldsymbol{u}^-)\right)_{\partial \mathcal{E}^e}, \tag{9}$$

The boundary conditions are imposed for these operators slightly differently than for $\boldsymbol{N}^e(\boldsymbol{u})$ and $N(\theta, \boldsymbol{u})$. For Dirichlet boundaries central fluxes become $\boldsymbol{u}^* = \boldsymbol{g}_D$ and $p^* = p^-$, and for zero Neumann boundaries, we choose $\boldsymbol{u}^* = \boldsymbol{u}^-$ and $p^* = 0$.

To discretize the Laplacian operator L^e, we follow follow the Symmetric Interior Penalty DG (SIPDG) approach (Wheeler, 1978; Arnold, 1982) which reads the following variational definition of the operator for velocity and temperature

$$(v, L^e \boldsymbol{u}^e)_{\mathcal{E}^e} = -(\nabla v, \nabla \boldsymbol{u}^e)_{\mathcal{E}^e} + (v, \boldsymbol{n} \cdot \nabla \boldsymbol{u})_{\partial \mathcal{E}^e} - \frac{1}{2}(\boldsymbol{n} \cdot \nabla v, [[\boldsymbol{u}]])_{\partial \mathcal{E}^e} + (v, \tau [[\boldsymbol{u}]])_{\partial \mathcal{E}^e} \tag{10.1}$$

$$(v, L^e \theta^e)_{\mathcal{E}^e} = -(\nabla v, \nabla \theta^e)_{\mathcal{E}^e} + (v, \boldsymbol{n} \cdot \nabla \theta)_{\partial \mathcal{E}^e} - \frac{1}{2}(\boldsymbol{n} \cdot \nabla v, [[\theta]])_{\partial \mathcal{E}^e} + (v, \tau [[\theta]])_{\partial \mathcal{E}^e} \tag{10.2}$$

where τ is the penalty parameter and must be chosen to be sufficiently large in order to enforce coercivity. On the other hand, selecting large τ increases the condition number of the Laplacian operator and degrades the performance of linear solvers. Along each face of an element shared by its neighbor element, $\partial \mathcal{E}^{ef} = \mathcal{E}^{e+} \cap \mathcal{E}^{e-}$, penalty parameter τ^{ef} is selected using the lower bound estimate derived in in Shahbazi (2005) as

$$\tau^{ef} = \frac{(N+1)(N+2)}{2} max\left(\frac{1}{h_+^{ef}}, \frac{1}{h_-^{ef}}\right) \tag{11}$$

where h_+^{ef} and h_-^{ef} are characteristic length scales of the elements $\mathcal{E}^{e+}$ and $\mathcal{E}^{e-}$ on either side of the face $\partial \mathcal{E}^{ef}$ and

are defined as $h_{+}^{ef} = \frac{|\mathcal{E}^{e+}|}{|\partial\mathcal{E}^{ef}|}$ and $h_{-}^{ef} = \frac{|\mathcal{E}^{e-}|}{|\partial\mathcal{E}^{ef}|}$. Boundary conditions for the discretized Laplacian operator are imposed in a way analogous to that described for the gradient and divergence operators for pressure and velocity systems. For the heat equation, non-homogeneous temperature Neumann data is imposed as $\nabla\theta^{+} = \nabla\theta^{-} - 2\mathrm{g_N}$.

With the definitions of discrete operators $\boldsymbol{N}^e$, N^e, L^e, $\boldsymbol{G}^e$ and D^e, we complete the semi-discrete form of the scheme given in eq. (6). Assembling the semi-discrete system defined on each element $\mathcal{E}$ into global system, we arrive to following global problem

$$D\mathbf{U} = \mathbf{0}, \tag{12.1}$$

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{N}(\mathbf{U}) = -\mathbf{G}P + \frac{1}{Re}\mathbf{L}\mathbf{U} + \mathbf{S_U}, \tag{12.2}$$

$$\frac{\partial \Theta}{\partial t} + N(\Theta, \mathbf{U}) = \frac{1}{RePr}L\Theta + S_{\Theta}. \tag{12.3}$$

To simplify the notation, we use capital letters and drop the superscript e to denote the global assembled vectors of the degrees of freedom as in Karakus et al. (2019b). In the next section, we proceed to the fully discrete scheme by introducing the semi-explicit time integration method.

Temporal Discretization

A high-order temporal discretization is used for the flow and energy equations by adopting an 3^{rd} order backward differentiation method for the stiff diffusive terms and an 3^{rd} order extrapolation method for advective terms. Then, equation (12) can be advanced from time level t^n to $t^{n+1} = t^n + \Delta t$ by using this formulation as

$$D \cdot \boldsymbol{U}^{n+1} = 0, \tag{13.1}$$

$$\gamma \boldsymbol{U}^{n+1} = \sum_{i=0}^{2} \beta_i \boldsymbol{U}^{n-i} - \Delta t \sum_{i=0}^{2} \alpha_i \left(\boldsymbol{N}(\boldsymbol{U}^{n-i}) - \boldsymbol{S}_U^{n-i}\right) + Re^{-1}\Delta t L \boldsymbol{U}^{n+1} - \Delta t \boldsymbol{G} P^{n+1} \tag{13.2}$$

$$\gamma \Theta^{n+1} = \sum_{i=0}^{S} \beta_i \Theta^{n-i} - \Delta t \sum_{i=0}^{S} \alpha_i N\left(\Theta^{n-i}, \boldsymbol{U}^{n-i}\right) + \Delta t \sum_{i=0}^{S} \alpha_i S_{\Theta}^{n-i} + (RePr)^{-1}\Delta t L \Theta^{n+1} \tag{13.3}$$

where β, and γ correspond to the stiffly stable backwards differentiation coefficients and α is the extrapolation scheme coefficients. For the third order scheme, the coefficients are given as $\gamma = 11/6$, $\beta_0 = 3$, $\beta_1 = -3/2$,$\beta_2 = 1/3$ and $\alpha_0 = 3$, $\alpha_1 = -3$, $\alpha_2 = 1$.

Following Shahbazi et al. (2007), the fully discrete scheme eq. (13) is replaced with an algebraically split version in order to avoid solving a fully coupled system for velocity, temperature and pressure which reads six fractional steps as follows

$$\hat{\Theta} = \sum_{i=0}^{S} \beta_i \Theta^{n-i} - \Delta t \sum_{i=0}^{2} \alpha_i \left(N(\Theta^{n-i}, \mathbf{U}^{n-i}) - S_{\Theta}^{n-i}\right), \tag{14.1}$$

$$\left(-L + \frac{\gamma RePr}{\Delta t}\mathcal{I}\right)\Theta^{n+1} = \frac{RePr}{\Delta t}\hat{\Theta}, \tag{14.2}$$

$$\widehat{\mathbf{U}} = \sum_{i=0}^{S} \beta_i \mathbf{U}^{n-i} - \Delta t \sum_{i=0}^{S} \alpha_i \left(\mathbf{N}\left(\mathbf{U}^{n-i}\right) - \mathbf{S}_{\mathbf{U}}^{n-i}\right), \tag{14.3}$$

$$\left(-L + \frac{\gamma Re}{\Delta t}\mathcal{I}\right)\widehat{\widehat{\mathbf{U}}} = \frac{Re}{\Delta t}\widehat{\mathbf{U}}, \tag{14.4}$$

$$-LP^{n+1} = -\frac{\gamma}{\Delta t}\mathbf{D} \cdot \widehat{\widehat{\mathbf{U}}}, \tag{14.5}$$

$$\mathbf{U}^{n+1} = \widehat{\widehat{\mathbf{U}}} - \frac{\Delta t}{\gamma}\mathbf{G}P^{n+1}. \tag{14.6}$$

In the splitting scheme eq. (14.1) and (14.3) are pure advection evaluations, eq. (14.2) and (14.4) are implicit screened Poisson solves for diffusive terms and eq. (14.5) is a pressure Poisson equation to enforce incompressibility condition. Equation (14.6) corrects the intermediate velocity field incorporating the updated pressure. This splitting scheme reduces the cost of the temporal discretization to a combination of two explicit and three linear elliptic solve steps. Since the advection evaluations are explicitly treated in time, the maximum stable time step size is restricted by Courant-Fredrichs-Lewy (CFL) condition. Although stability of explicit schemes is restricted by a severe time step size, they are computationally inexpensive that conveys the computational load of each time step to elliptic solves. Because of that, effective linear solvers are crucial to obtain fast and scalable coupled flow and energy solvers.

Linear Solvers

Each time step of the temporal splitting discretization (14) requires solving two discrete Helmholtz equations, (14.2), (14.4) and a discrete Poisson problem (14.3). For high resolution solutions, i.e., large meshes and/or high degree N, assembling a full matrix and using a direct solver is not practically applicable. Thus, due to symmetric positive-definite structure of the IPDG discretization (10.1), we choose a preconditioned conjugate gradient (PCG) iterative method to solve the elliptic equations.

For the solution of Helmholtz equations (14.2) and (14.4), we choose the scaled inverse mass matrix on each element for the preconditioner. As the time step size is relatively small and coefficients $Re/\Delta t$ and $RePr/\Delta t$ are large, the Helmholtz operators are dominated by the mass matrix. Since the mass matrix is block diagonal and geometric factors are constant for triangular elements, inverting the mass matrix is computationally cheap and simple which makes the inverse matrix preconditioner very effective for the Helmholtz solves (Karakus et al., 2019b).

For the pressure Poisson problem, we consider a p-multigrid preconditioner (pMG) where we manually coarsen from degree N to degree 1 before setting up the algebraic multigrid (AMG) levels for the degree 1 coarse stiffness matrix. The coarse levels of this AMG method are constructed as unsmoothed aggregations of maximal independent node sets, see (Notay, 2006, 2010), while smoothing is chosen to be a degree 2 Chebyshev iteration (Adams et al., 2003). The multigrid preconditioning cycle itself consists of a K-cycle on the finest two levels, followed by a V-cycle for the remaining coarse levels (Gandham et al., 2014). With this manual coarsening approach, we are able to implement the finest levels of the multigrid cycle in a matrix-free way and avoid the storage of the full degree N stiffness matrix.

RESULTS

We have implemented the solver on the libParanumal (Chalmers et al., 2020) library using C++ together with the OCCA API and OKL kernel language (Medina et al., 2014) and MPI to enable using distributed multi-GPU/CPU platforms. OCCA is an abstracted programming model designed to encapsulate native languages for parallel devices such as CUDA, OpenCL, HIP, and OpenMP. For all the results presented in this section, we have compiled the source code using the GNU GCC 9.3.0 compiler and the Nvidia CUDA V11.0.221 NVCC compiler. The tests were run using Nvidia Tesla P100 GPUs on a machine equipped with an Intel Xeon E5-2680v4 processor.

We solved 2D thermal convection tests to verify the spatial accuracy of the presented algorithm on 2D unstructured triangular grids. To evaluate the accuracy of the numerical scheme, we use L_2 norm of error given as,

$$L_2 = \sum(\phi^e - \phi^{ex}, \phi^e - \phi^{ex})_{\mathcal{E}^e \in \Omega}, \quad (15)$$

where ϕ is the arbitrary field that error is computed and ϕ^{ex} denotes the exact solution or a very accurate approximation of the exact solution if it is not explicitly known.

Poiseuille Flow

In the first test case, we consider two-dimensional channel flow with a fully developed Poiseuille profile to show the spectral convergence rate of the solver. The channel having the dimension of $[0,2] \times [-1,1]$ is uniformly heated from the lower wall with $\theta_L = 1$ while isothermal upper plate is kept with $\theta_U = 0$. No-slip boundary conditions are imposed for upper and lower walls. The fully developed velocity field with linear temperature profile is prescribed as the initial condition which reads

$$u = 1 - y^2, \quad v = 0,$$

$$p = \frac{Ra}{2PrRe^2}\left(y - \frac{y^2}{2}\right) - \frac{2x}{Re}, \quad \theta = \frac{1-y}{2}.$$

Computations are performed on the successively refined meshes by uniformly dividing the initial coarse level grid. The initial mesh, defined with the characteristic scale h_0, has the element number of $K = 68$ which is constructed by triangles with the edge length of $h = 0.4$ on the boundary of domain $\partial\Omega$. Then the sequence of meshes constructed with the element number of $K = 272$, $K = 1088$ and $K = 4352$ for $h = 0.5h_0$, $h = 0.25h_0$ and $h = 0.125h_0$, respectively. Approximating polynomial order is changed from $N = 1$ to $N = 5$ for all mesh configurations.

Figure 1 shows the computed L_2 norm of the numerical error in the x component of velocity, u and the temperature, θ fields for the flow conditions $Ra = 1000$, $Re = 100$ and $Pr = 0.71$ at the final time $T = 10$. The velocity and the temperature remain unchanged indicating no flow bifurcation in the flow field. The figure clearly demonstrates the expected spectral converge rates in the numerical error i.e. h^{N+1} accuracy for velocity and temperature.

Differentially Heated Square Cavity

We focus on the natural convection problem on a closed enclosure. The cavity width and height are denoted with $W = 1$ and $H = 1$ respectively leading $W:H$ aspect ratio, $A = 1$. The enclosure boundary conditions are simple and consist of no-slip walls, i.e. $u = v = 0$ on all four walls. The thermal boundary conditions on the left and right walls are prescribed as

$$\theta_L = 1, \quad \theta_R = 0$$

and the upper and the lower walls are thermally insulated

$$\frac{\partial\theta}{\partial y} = 0, \quad for \quad y = 0, y = H.$$

We solved the problem for polynomial order, $N = 4$ on a relatively coarse grid composed of uniformly spaced triangular grids having characteristic length of $h = 0.1$ which leads element number of $K = 242$. The flow conditions are $Pr = 0.71$, $Ra = 10^3, 10^4, 10^5$. Reference velocity is selected as $U_r = \sqrt{h\beta(\theta_L - \theta_R)W}$ so that $Re = \sqrt{Gr}$.

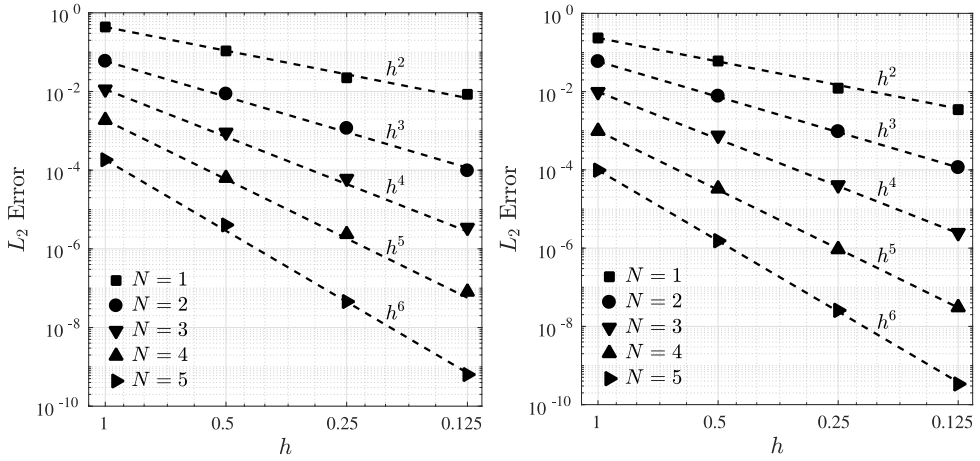

Figure 1. Spatial accuracy test for the Poiseuille test problem using L_2 relative errors on successively refined triangular elements. The error in the x-velocity is shown on the left and the error in the temperature is shown right

Table 1. Maximum and minimum velocities along the center lines of square cavity for $Pr = 0.71$ and $Ra = 10^3, 10^4, 10^5$

	$Ra = 10^3$		$Ra = 10^4$		$Ra = 10^5$	
	u_{max}	v_{max}	u_{max}	v_{max}	u_{max}	v_{max}
Present	0.137	0.139	0.192	0.233	0.129	0.257
Stokos et al. (2015)	0.137	0.139	0.192	0.233	0.130	0.256
De Vahl Davis (1983)	0.136	0.138	0.192	0.234	0.153	0.261

In Table 1, we presented the computed maximum and minimum velocities along the horizontal $y = 0.5$ and the vertical $x = 0.5$ lines. u_{max} stands for the maximum value of $x-$velocity, u along the vertical line while v_{max} denotes maximum $y-$velocity, v along the horizontal line. For all simulations, our results compare well with the reported values in the relevant literature (De Vahl Davis, 1983; Stokos et al., 2015). Figure 2 and Figure 3 illustrate temperature contours and velocity profiles on center lines for different Ra numbers. The temperature contours and velocity fields depicted by center lines profiles provide qualitative agreement with those reported in the literature (Stokos et al., 2015; Hossain et al., 2021).

Mixed Convection

The square cylinder test is considered with various channel confinement degrees and aiding ($Ri > 0$) and opposing ($Ri < 0$) buoyancy strengths or without the effect of thermal buoyancy $Ri = 0$. The computational domain is composed of a square cylinder of length, B located at the center of the domain and a vertically aligned channel of width L. The channel blockage ratio $BR = B/L$ defines the degree of channel confinement when imposing no-slip conditions on the left and right sides of the channel. We consider uniform inflow velocity on the lower side and zero Neumann conditions are used on the upper boundary for velocity.

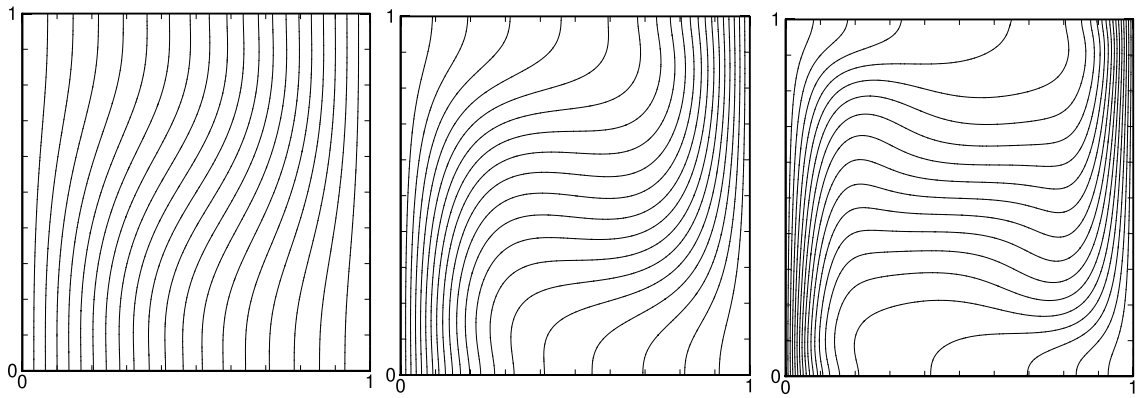

Figure 2. Temperature contours for the square cavity test. The temperature contours drawn between 1 and 0 with the increment of 0.05 for $Ra = 10^3, 10^4, 10^5$ from left to right

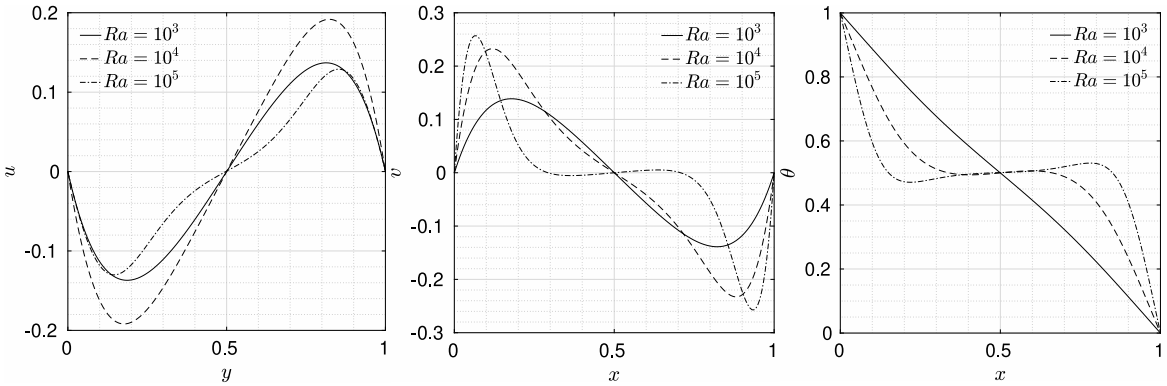

Figure 3. Velocity and temperature profiles along the $y = 0.5$ and $x = 0.5$ lines for the square cavity test

The square cylinder is kept at a fixed temperature of $\theta = 1$ and homogeneous temperature is applied on the channel side walls and the inlet. In this test, we consider following geometric and flow parameters: blockage ratios, $BR = 0\%, 10\%, 30\%, 50\%$, and Richardson number, $Ri = -1.0, -0.5, 0.0$ for the fixed Reynolds number $Re = 100$ and Prandtl number $Pr = 0.71$.

Table 2. Comparison of Strouhal number for flow past a square cylinder test at $Re = 100$ for $BR = 2.3\%$ and $Ri = 0$

	N	St
Present	3	0.144
	5	0.145
Ferrer and Willden (2011)	5	0.144
Darekar and Sherwin (2001)	6	0.145
Shahbazi et al. (2007)	4	0.145

As a preliminary test, we solved the problem without buoyancy forcing, $Ri = 0$ to validate the solver in a complex flow. The flow condition $R = 100$ leads an unsteady Von Karman vortex street with a characteristic vortex shedding frequency which can be represented as a non-dimensional Strouhal number,$St = fB/U_r$, where f, B, and U_r are the non-dimensional vortex shedding frequency, characteristic length as cylinder dimension and the uniform free stream velocity, respectively. We use the same domain and boundary conditions with Darekar and Sherwin (2001); Ferrer and Willden (2011); Shahbazi et al. (2007), with blockage ratio, $BR = 2.3\%$. In order to make direct comparisons between the reference studies, we created identical mesh configuration and used the same non-dimensional time step size which lead to 896 triangular elements and $\Delta t = 0.002$. In Table 2, we report computed St numbers for the approximating polynomial orders, $N = 3$ and $N = 5$. In both cases, we observe that present solver provides good agreement with the reference, high-order numerical results.

Figure 4 shows instantaneous vorticity contours around the square cylinder for $BR = 30\%$ and $Ri = -1.0, -0.5, 0.5$. The free and forced convection effects are comparable under these flow settings which makes the vortex shedding phenomena more complicated. In the figure, the vorticity contours show a wavering motion which increases with increased cooling (decreasing $Ri < 0$) of the cylinder and decreases with increased heating (increasing $Ri > 0$) before it ceases at critical \$Ri\$ for the given specific blockage ratio. In this work, we only focus on the solver performance in reflecting the flow physics and skip to determine critical $Ri > 0$ numbers that Von Karman vortex street breakdowns.

Table 3. Strouhal numbers for different Ri numbers and blockage ratios at $Re = 100$

	Ri		
BR	-1.0	-0.5	0
10%	0.127	0.137	0.176
30%	0.303	0.312	0.322
50%	0.498	0.508	0.527

In Table 3, Strouhal numbers are listed for different opposing Ri numbers and channel blockage ratios for $N = 5$. A fast Fourier transform of the lift coefficient's time history is used to determine the Strouhal number accurately. For the opposing buoyancy cases, the table shows that the Strouhal number increases monotonically with increasing Ri at a constant blockage ratio before Ri number reaches the critical value where the breakdown of the Karman vortex street is obseved. It is also shown that the Strouhal number increases monotonically with increasing blockage ratio due to the acceleration of the cylinder-wall boundary layer.

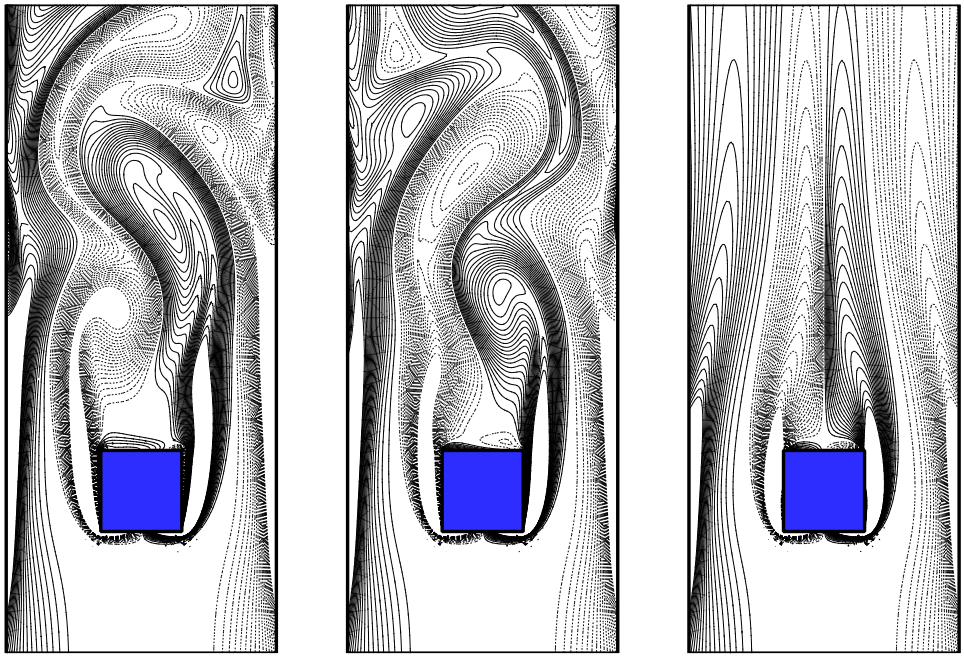

Figure 4. Instantaneous vorticity contours at $t = 100$ for different Ri and the influence of aiding and opposing buoyancy at a blockage ratio of 30% and $Re = 100$. The solid and dashed lines represent positive and negative vorticity contours, respectively. (Only a part of the domain is shown)

CONCLUSION

We have presented a GPU-accelerated, high-order, nodal discontinuous Galerkin approach for the solution of incompressible thermal convection problems on unstructured meshes. The incompressible Navier–Stokes and the energy equation written in terms of temperature are coupled through a Boussinesq type thermal body-forcing term. The equations system is discretized in time using an algebraic splitting scheme where advection terms are evaluated explicitly while diffusive terms together with Poisson equation to impose incompressibility are treated implicitly. This further decouples velocity field from pressure. We showed that the scheme achieves designed order of accuracy and preserves stability for thermal convection problems.

The presented solver is implemented on the open source project libParanumal (LIBrary of PARAllel NUMerical ALgorithms) (Chalmers et al., 2020). libParanumal consists of a collection of mini-apps with high-performance portable implementations of high-order finite element discretizations for a range of different fluid flow models (Karakus et al., 2019a,b).

Extending the scheme to 3D using tetrahedral elements and assessing its performance on multi-GPU systems for triangular/tetrahedral elements remain to be investigated.

ACKNOWLEDGEMENTS

The author thanks to Dr. Noel Chalmers and Prof. Tim Warburton for their helpful discussions and valuable comments.

REFERENCES

Adams, M., Brezina, M., Hu, J., and Tuminaro, R., 2003, Parallel multigrid smoothing: polynomial versus Gauss–Seidel, *Journal of Computational Physics*, 188(2), 593–610.

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

Chalmers, N., Karakus, A., Austin, A. P., Swirydowicz, K., and Warburton, T., 2020, libParanumal: a performance portable high-order finite element library. *Release 0.4.0*.

Chan, J., Wang, Z., Modave, A., Remacle, J. F., and Warburton, T., 2016, GPU-accelerated discontinuous Galerkin methods on hybrid meshes, *Journal of Computational Physics*, 318, 142–168.

Darekar, R. M. and Sherwin, S. J., 2001, Flow past a square-section cylinder with a wavy stagnation face, *Journal of Fluid Mechanics*, 426, 263–295.

De Vahl Davis, G., 1983, Natural convection of air in a square cavity: A bench mark numerical solution, *International Journal for Numerical Methods in Fluids*, 3(3), 249–264.

Ferrer, E. and Willden, R. H. J., 2011, A high order discontinuous Galerkin finite element solver for the

incompressible Navier–Stokes equations, *Computers & Fluids*, 46(1), 224–230.

Gandham, R., Esler, K., and Zhang, Y., 2014, A GPU accelerated aggregation algebraic multigrid method, *Computers & Mathematics with Applications*, 68(10), 1151–1160.

Gandham, R., Medina, D., and Warburton, T., 2015, GPU accelerated discontinuous Galerkin methods for shallow water equations, *Communications in Computational Physics*, 18(1), 37–64.

Hesthaven, J. S. and Warburton, T., 2008, Nodal discontinuous Galerkin methods: algorithms, analysis, and applications, Springer.

Hossain, M. Z., Cantwell, C. D., and Sherwin, S. J., 2021, A spectral/hp element method for thermal convection, *International Journal for Numerical Methods in Fluids*, 93(7), 2380–2395.

Karakus, A., Chalmers, N., Hesthaven, J. S., and Warburton, T., 2019a, Discontinuous Galerkin discretizations of the Boltzmann–BGK equations for nearly incompressible flows: Semi-analytic time stepping and absorbing boundary layers, *Journal of Computational Physics*, 390, 175–202.

Karakus, A., Chalmers, N., Swirydowicz, K., and Warburton, T., 2019b, A GPU accelerated discontinuous Galerkin incompressible flow solver, *Journal of Computational Physics*, 390, 380–404.

Karakus, A., Warburton, T., Aksel, M. H., and Sert, C., 2016a, A GPU-accelerated adaptive discontinuous Galerkin method for level set equation, *International Journal of Computational Fluid Dynamics*, 30(1), 56–68.

Karakus, A., Warburton, T., Aksel, M. H., and Sert, C., 2016b, A GPU accelerated level set reinitialization for an adaptive discontinuous Galerkin method, *Computers & Mathematics with Applications*, 72(3), 755–767.

Kumar, A. and Pothérat, A., 2020, Mixed baroclinic convection in a cavity, *Journal of Fluid Mechanics*, 88. Publisher: Cambridge University Press.

Medina, D. S., St-Cyr, A., and Warburton, T., 2014, OCCA: A unified approach to multi-threading languages. *arXiv:1403.0968*.

Modave, A., St-Cyr, A., and Warburton, T., 2016, GPU performance analysis of a nodal discontinuous Galerkin method for acoustic and elastic models, *Computers & Geosciences*, 91, 64–76.

Notay, Y., 2006, Aggregation-based algebraic multilevel preconditioning, *SIAM journal on matrix analysis and applications*, 27(4), 998–1018.

Notay, Y., 2010, An aggregation-based algebraic multigrid method, *Electronic transactions on numerical analysis*, 37(6), 123–146.

Roca, X., Nguyen, N. C., and Peraire, J., 2011, GPU-accelerated sparse matrix-vector product for a hybridizable discontinuous Galerkin method. *In Aerospace Sciences Meetings. American Institute of Aeronautics and Astronautics*, AIAA 2011–687.

Saha, S., Klewicki, J. C., Ooi, A. S. H., and Blackburn, H. M., 2015, Comparison of thermal scaling properties between turbulent pipe and channel flows via DNS, *International Journal of Thermal Sciences*, 89, 43–57.

Shahbazi, K., 2005, An explicit expression for the penalty parameter of the interior penalty method, *Journal of Computational Physics*, 205(2), 401–407.

Shahbazi, K., Fischer, P. F., and Ethier, C. R., 2007, A high-order discontinuous Galerkin method for the unsteady incompressible Navier-Stokes equations, *Journal of Computational Physics*, 222(1), 391– 407.

Stokos, K., Vrahliotis, S., Pappou, T., and Tsangaris, S., 2015, Development and validation of an incompressible Navier-Stokes solver including convective heat transfer, *International Journal of Numerical Methods for Heat & Fluid Flow*, 25(4), 861–886.

Warburton, T., 2006, An explicit construction of interpolation nodes on the simplex, *Journal of Engineering Mathematics*, 56(3), 247–262.

Wheeler, M. F., 1978, An elliptic collocation-finite element method with interior penalties, *SIAM Journal on Numerical Analysis*, 15(1), 152–161.

Swirydowicz, K., Chalmers, N., Karakus, A., and Warburton, T., 2019, Acceleration of tensor-product ´ operations for high-order finite element methods, *The International Journal of High Performance Computing Applications*, 33(4).