

Copyright
by
Shinhoo Kang
2019

The Dissertation Committee for Shinhoo Kang
certifies that this is the approved version of the following dissertation:

**High-order (Hybridized) Discontinuous Galerkin method
for Geophysical flows**

Committee:

Tan Bui-Thanh, Supervisor

Karen E. Willcox

Leszek F. Demkowicz

Omar Ghattas

Todd Arbogast

Fabrizio Bisetti

**High-order (Hybridized) Discontinuous Galerkin method
for Geophysical flows**

by

Shinhoo Kang

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

August 2019

Dedicated to my beloved wife Mi hye Pyo,
my children Yeseong, Yean, Yeha,
and my parents Jongsup Kang and Okhui Kim.
For their endless love, support and encouragement

Acknowledgments

I would like to extend special thanks to the following people. This thesis would not be possible without their support. The first is my advisor, Prof. Tan Bui-Thanh. He gave me this PhD opportunity, guided me to tackle various topics, and supported me so far. He reminds me of a nuclear power plant, which is powerful, dynamic, incomparable to other energy sources, and requires high technology for operations. He has a strong passion for research. He works very hard and is very energetic. He also has high standards for research. I still think that it takes time to reach such a standard (compared to him), but I will keep it in my mind and try to follow what I learned from him in my career.

I would like to thanks my committee members Prof. Willcox, Prof. Demkowicz, Prof. Ghattas, Prof. Arbogast and Prof. Bisetti. I was lucky enough to have world-class professors in my committee. I appreciate their time and valuable opinions on my research. I thank Prof. Willcox and Prof. Ghattas for their time and valuable comments on my research. I thank Prof. Demkowicz for his teaching, especially math classes for engineering students. It has been incredibly helpful for my research. I thank Prof. Arbogast for his teaching and helpful comments on my research. He is very kind and has a keen eye for research. Based on his scaled system, I could develop the HDG method for the degenerate elliptic equation. I thank Prof. Bisetti for his CFD teaching and helpful suggestions for my research.

I would like to thank my collaborators Prof. Francis X. Giraldo and Prof. David A. Kopriva. I thank Prof. Francis X. Giraldo for the recommendation letter and sharing me his NUMA2D code, with which I was able to develop IMEX HDG-

DG scheme. I thank Prof. David A. Kopriva for teaching the concepts of watertight, metric identity, two-point flux and more. I also thank Prof. Jeonghun Lee and Prof. Sanghyun Lee for helpful suggestions on my research and PhD life. I would like to thank Sue for helping with all the travels and conferences.

I thank my colleagues and friends. I want to thank Sriram Muralikrishnan. He is my academic brother. He is really smart and helpful for discussing research and many other things. We have overcome many difficulties and shared happiness and sadness. I am very happy that we both graduated together. I also thank Stephen Shannon for helpful discussion and sharing his code on nonconforming meshes. This was very helpful for me to develop a sliding-mesh interface. I also would like to thank Korean friends Sehyuk, Seungryul, Seongpil, Yunho, Sehyun, Anji, and Geon. Sehyuk, Seungryul and Seongpil have helped me a lot in settling in Austin. Yunho is steady and reliable. He helped me prepare for the qualifying exam and many other things. Sehyun and Anji remind me of university life at Handong Global University in South Korea. They also gave helpful suggestions for my research. I thank my other friends in Austin/PhD time Woohyung, Juyoung, Wonsun, Sundeep, Chenglin, Daiju, and Youngwon.

I would like to thank my friends and professors in Korea. I thank Dr. Taejin Oh for helping and supporting me to pursue a PhD. I also thank his recommendation letter. I thank Dr. Mi-Lim Ou for giving me a chance to work in national institute of meteorological sciences. I thank my previous advisor at Handong Global University, Prof. Cheo K. Lee for teaching, guiding and encouraging me to pursue a PhD. He accepted me as his student in the Master's program under one condition that I will pursue a PhD after graduation. I am very happy that I could finally keep my promise with him. I thank my dear friends Prof. Doohee Chung and Dr. Eundo Kim in Handong Global University. We had shared and discussed our uncertain future and

dreams together. After long journey, we finally arrived at academia. I am thrilled how God will lead each one of us in future.

Finally, I would like to thank my father, mother, sister Jihu, brother-in-law Heojin, brother-in-law Junhyuk, my father-in-law, my mother-in-law and the rest of my family for their love and support. Of course, I thank my beloved wife Mi hye and my sons Yeseong, Yean, Yeha for their patience, love and support. They help me to make each day special. I truly thank you all and may God bless you.

This work was partially supported by DOE grants DE-SC0010518, DE-SC0011118 and NSF Grant NSF-DMS1620352. I am grateful for the supports.

High-order (Hybridized) Discontinuous Galerkin method for Geophysical flows

by

Shinhoo Kang, Ph.D.

The University of Texas at Austin, 2019

Supervisor: Tan Bui-Thanh

As computational research has grown, simulation has become a standard tool in many fields of academic and industrial areas. For example, computational fluid dynamics (CFD) tools in aerospace and research facilities are widely used to evaluate the aerodynamic performance of aircraft or wings. Weather forecasts are highly dependent on numerical weather prediction (NWP) model. However, it is still difficult to simulate the complex physical phenomena of a wide range of length and time scales with modern computational resources. In this study, we develop a robust, efficient and high-order accurate numerical methods and techniques to tackle the challenges.

First, we use high-order spatial discretization using (hybridized) discontinuous Galerkin (DG) methods. The DG method combines the advantages of finite volume and finite element methods. As such, it is well-suited to problems with large gradients including shocks and with complex geometries, and large-scale simulations. However, DG typically has many degrees-of-freedom. To mitigate the expense, we use hybridized DG (HDG) method that introduces new “trace unknowns” on the mesh

skeleton (mortar interfaces) to eliminate the local “volume unknowns” with static condensation procedure and reduces globally coupled system when implicit time-stepping is required. Also, since the information between the elements is exchanged through the mesh skeleton, the mortar interfaces can be used as a glue to couple multi-phase regions, e.g., solid and fluid regions, or non-matching grids, e.g., a rotating mesh and a stationary mesh. That is the HDG method provides an efficient and flexible coupling environment compared to standard DG methods.

Second, we develop an HDG-DG IMEX scheme for an efficient time integrating scheme. The idea is to divide the governing equations into stiff and nonstiff parts, implicitly treat the former with HDG methods, and explicitly treat the latter with DG methods. The HDG-DG IMEX scheme facilitates high-order temporal and spatial solutions, avoiding too small a time step. Numerical results show that the HDG-DG IMEX scheme is comparable to an explicit Runge-Kutta DG scheme in terms of accuracy while allowing for much larger timestep sizes. We also numerically observe that IMEX HDG-DG scheme can be used as a tool to suppress the high-frequency modes such as acoustic waves or fast gravity waves in atmospheric or ocean models. In short, IMEX HDG-DG methods are attractive for applications in which a fast and stable solution is important while permitting inaccurate processing of the fast modes.

Third, we also develop an EXPONENTIAL DG scheme for an efficient time integrators. Similar to the IMEX method, the governing equations are separated into linear and nonlinear parts, then the two parts are spatially discretized with DG methods. Next, we analytically integrate the linear term and approximate the nonlinear term with respect to time. This method accurately handles the fast wave modes in the linear operator. To efficiently evaluate a matrix exponential, we employ the cutting-edge adaptive Krylov subspace method.

Finally, we develop a sliding-mesh interface by combining nonconforming treatment and the arbitrary Lagrangian-Eulerian (ALE) scheme for simulating rotating flows, which are important to estimate the characteristics of a rotating wind turbine or understanding vortical structures shown in atmospheric or astronomical phenomena. To integrate the rotating motion of the domain, we use the ALE formulation to map the governing equation to the stationary reference domain and introduce mortar interfaces between the stationary mesh and the rotating mesh. The mortar structure on the sliding interface changes dynamically as the mesh rotates.

Table of Contents

Acknowledgments	v
Abstract	viii
List of Tables	xiv
List of Figures	xvi
Chapter 1. Introduction	1
1.1 Outline	5
Chapter 2. Degenerate Elliptic equations	7
2.1 Introduction	7
2.2 Handling the degeneracy	11
2.2.1 The scaled system	11
2.2.2 Upwind-based HDG flux	12
2.3 HDG formulation	14
2.3.1 Finite element definitions and notations	14
2.3.2 Weak form	15
2.3.3 Well-posedness	18
2.3.4 Error analysis	20
2.4 Numerical results	25
2.4.1 Non-degenerate case	25
2.4.2 Degenerate case with a smooth solution	27
2.4.3 Degenerate case with low solution regularity	29
2.4.4 Sensitivity of τ for the degenerate case with smooth solution . .	32
2.4.5 Enhance accuracy by post-processing	32
2.4.6 Non-degenerate case in three dimensions	36
2.5 Discussions	38
2.6 Auxiliary results	42

Chapter 3. IMEX HDG-DG	44
3.1 Implicit-Explicit (IMEX) Runge-Kutta methods	45
3.2 Shallow water equations	47
3.2.1 Planar shallow water equations	47
3.2.2 Shallow water equations on a sphere	48
3.3 Euler equations	51
3.4 Spatial Discretization	53
3.4.1 DG and HDG spatial discretization	53
3.4.2 Coupled HDG-DG spatial discretization	55
3.4.2.1 Shallow water equations	55
3.4.2.2 Euler equations	57
3.5 Temporal Discretization	58
3.6 Numerical Results	60
3.6.1 Moving vortex	61
3.6.2 Water height perturbation	66
3.6.3 Steady-state geostrophic flow	69
3.6.4 Steady-state geostrophic flow with compact support	73
3.6.5 Zonal flow over an isolated mountain	74
3.6.6 Rossby-Haurwitz wave	75
3.6.7 Barotropic instability test	76
3.6.8 Inertia-gravity waves	77
3.6.9 Rising thermal bubble	79
3.6.10 Density current	80
3.7 Discussions	81
3.8 Auxilliary results	83
3.8.1 Hybridized upwind HDG flux with a one-trace variable	83
3.8.2 Well-balancing property	84
 Chapter 4. EXPONENTIAL DG	 93
4.1 Exponential time integrations	94
4.1.1 Krylov subspace projection-based methods	97
4.2 Governing equations and spatial discretization	99
4.2.1 Constructing a linear and a nonlinear operator in DG	100
4.2.2 Euler equations	101
4.3 Numerical Results	103
4.3.1 A system of two ordinary differential equations (ODE)	103
4.3.2 Isentropic vortex translation	107
4.4 Discussions	115

Chapter 5. Sliding-mesh Interface	120
5.1 Nonconforming mesh treatment	121
5.1.1 Projection operator from a face to a mortar	123
5.1.2 Projection operator from a mortar to a face	124
5.1.3 Watertight mesh	125
5.2 Arbitrary Lagrangian-Eulerian	127
5.2.1 Weak formulation	129
5.2.2 Linear convection equation	131
5.2.3 Linear convection-diffusion equation	132
5.3 Sliding mesh interface	132
5.4 Numerical results	133
5.4.1 Steady-state advection with a discontinuous solution	134
5.4.2 Constant solution	135
5.4.3 Cosine bell	137
5.4.4 Rotating Gaussian	138
5.4.4.1 Nonconforming mesh	139
5.4.4.2 Moving mesh	140
5.4.4.3 Rotating mesh	143
5.4.5 Isentropic vortex	144
5.5 Discussions	146
Chapter 6. Conclusions and Future Work	148
Appendices	153
Appendix A. Flux Jacobian of Euler equations	154
A.1 Two-dimensional Euler equation	154
A.2 Three-dimensional Euler equation	156
Appendix B. Solid body rotation	159
Bibliography	161
Vita	192

List of Tables

2.1	Non-degenerate case: the results show that the HDG solutions for scaled pressure p and scaled velocity \mathbf{u} converge to the exact solution with optimal order of $k+1$ for both triangular and rectangular meshes. The upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ is used.	27
2.2	Degenerate case with a smooth solution: the errors $\ p^e - p\ _{\Omega_h}$ and the convergence rates for the scaled pressure. Four cases are presented: τ given (2.49), $\tau = \frac{1}{h}$, $\tau = 1$ and $\tau = 10$	33
2.3	Degenerate case with a smooth solution: the errors $\ \mathbf{u}^e - \mathbf{u}\ _{\Omega_h}$ and the convergence rates for the scaled velocity. Four cases are presented: τ given (2.49), $\tau = \frac{1}{h}$, $\tau = 1$ and $\tau = 10$	34
2.4	Non-degenerate case: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The post-processed solutions show asymptotically $(k+2)$ convergence rates for $k > 1$. . .	37
2.5	Degenerate case with a smooth solution: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [-0.5, 1]^2$. The post-processed solutions show asymptotically $(k + \frac{3}{2})$ convergence rates for $k > 1$	38
2.6	Degenerate case with low solution regularity for $\beta = -\frac{3}{4}$: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [-0.5, 1]^2$. . .	39
2.7	Degenerate case with low solution regularity for $\beta = -\frac{3}{4}$: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [0, 1]^2$. . .	40
2.8	Non-degenerate case in three dimensions: the results show that the HDG solutions for scaled pressure p and scaled velocity \mathbf{u} converge to the exact solutions with the rate in above $k + \frac{1}{2}$. The upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ is used.	41
3.1	Spatial convergence for the traveling vortex test using ARS3 HDG-DG. . .	63
3.2	Moving vortex for $Fr = 0.1$: the errors and the convergence rates for the free surface elevation η , and wallclock time at $T = 0.2$. The errors are computed over the domain $\tilde{\Omega} \in [-2, 2]^2$ with 32×32 elements. . .	66

3.3	Moving vortex for $Fr = 0.01$: the errors and the convergence rates for the free surface elevation η , and wall clock time at $T = 0.1$. The errors are computed over the domain $\tilde{\Omega} \in [-2, 2]^2$ with 32×32 elements.	67
3.4	Comparison of the ARS HDG-DG schemes with the three-trace variables $(\hat{\phi}_\eta, \hat{U}, \hat{V})$ in (3.28) and the one-trace variable $(\hat{\phi}_\eta)$ in (3.43) for the moving vortex for $Fr = 0.1$: the errors and the convergence rates for the free surface elevation η and the velocity \mathbf{u} , and wall clock time at $T = 0.2$. The errors are computed over the domain $\Omega \in [-2, 2]^2$ with 32×32 elements and $p = 6$ solution order.	92
4.1	System of two ODEs: time convergence study of ARS111 and ETD1. Using L_d gives better stability and accuracy than using L_c . With the small timestep size ($dt < 10^{-3}$), ETD1 solutions are more accurate than ARS111 solutions.	106
4.2	Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with $Cr = 27.9(dt = 0.25)$ and RK4 with $Cr = 0.56(dt = 0.005)$ are performed with $N_e = 6400$ and $k = 8$ solution order up to $t = 2$	113
4.3	Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with different degrees-of-freedom (DOF). For the number of DOF (NDOF) of 17,561,600, simulations are performed on $N_e = 51200$ and $k = 6$ solution order with $dt = 0.25(Cr = 32.98)$ up to $t = 1$, whereas for NDOF of 4,665,600, simulations are performed on $N_e = 6400$ and $k = 8$ solution order with $dt = 0.25(Cr = 27.93)$ up to $t = 2$	114
4.4	Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with $Cr = 16.49(dt = 0.125)$ and $Cr = 32.98(dt = 0.25)$ are performed with $N_e = 51200$ and $k = 6$ solution order up to $t = 1$	115
4.5	Isentropic vortex translation in three dimensions: weak scaling results for EPI2 with fixed Courant numbers. Each core has 18,225 degrees-of-freedom (=25 elements and 8th solution order).	117
4.6	Isentropic vortex translation in three dimensions: the total number of Krylov steps, N_{Krylov} , linearly increases by (i) the refining level, and (ii) Courant number.	119
5.1	The rotating Gaussian test: the results show that the DG solutions converge to the exact solution with the optimal order of $k + 1$ for linear conforming, linear nonconforming and curved nonconforming meshes.	141

List of Figures

2.1	Coarse grids for non-degenerate case with (a) rectangular and (b) triangular elements.	26
2.2	Degenerate case with a smooth solution: (a) contour plot of the pressure p field and (b) contour plot of the scaled pressure p with $N_e = 64^2$ and $k = 4$. The pressure field changes smoothly in the two-phase region Ω_2 , but suddenly becomes zero in the one-phase region Ω_1 . The abrupt change near the intersection Ω_{12} between the one- and two-phase regions is alleviated with the use of the scaled pressure p	28
2.3	Degenerate case with a smooth solution: convergence study for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field. The $(k + \frac{1}{2})$ convergence rates are obtained approximately for both the scaled pressure p and the scaled velocity \mathbf{u}	29
2.4	Degenerate case with low solution regularity: simulated with $N_e = 64^2$ and $k = 4$ are (a) pressure p for $\beta = -\frac{1}{4}$, (b) scaled pressure p for $\beta = -\frac{1}{4}$, (c) pressure p with $\beta = -\frac{3}{4}$, and (d) scaled pressure p for $\beta = -\frac{3}{4}$. The pressure field with $\beta = -\frac{3}{4}$ is less regular than that with $\beta = -\frac{1}{4}$. In both the cases, the pressure fields have low regularity near the intersection Ω_{12}	30
2.5	Degenerate case with low solution regularity: a convergence study with $\beta = -\frac{1}{4}$ for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field.	31
2.6	Degenerate case with low solution regularity: a convergence study with $\beta = -\frac{3}{4}$ for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field.	31
3.1	The moving vortex test case: time evolution of the free surface elevation η at times $t = 0$, $t = 1$ and $t = 2$ computed using ARS3 HDG-DG. The contour levels are from -0.6m to 0m with the step-size of 0.05m	62
3.2	Comparison between IMEX HDG-DG and RKDG for the moving vortex test case: (a) accuracy/stability and (b) wallclock time.	64
3.3	Water height perturbation test: free surface evolution for RK2 DG ($\Delta t=0.0002$) at $t = 0.016$, $t = 0.017$ and $t = 0.018$	69
3.4	Water height perturbation test: free surface evolution for RK2 DG ($\Delta t=0.0001$, left), ARS2 HDG-DG ($\Delta t=0.002$, center) and ARS2 HDG-DG ($\Delta t=0.02$, right) at times $t = 0$, $t = 0.06$, $t = 0.1$ and $t = 0.5$	70
3.5	Comparison between IMEX HDG-DG and RKDG for the smooth gravity wave: (a) the accuracy/stability and (b) the wall clock time. . . .	71

3.6	Geostrophic flow test case: (a) total height field from ARS2 HDG-DG at day 12 with $Cr = 1.36$, (b) the exact solution, and (c) the relative error of the height field.	72
3.7	Geostrophic flow test case: (a) time evolution of height field error in L_1 , L_2 and L_∞ norms, and (b) mass and energy loss of the ARS2 HDG-DG solution with timestep size of 864 s (i.e. $Cr = 1.36$).	72
3.8	Convergence studies for the ARS2 HDG-DG scheme when applied to the steady-state geostrophic flow with compact support with $Cr = 0.7$: (a) h-convergence with $p = 3$ and (b) p -convergence with $N_e = 6144$	73
3.9	Flow over an isolated mountain (red circle) computed with the ARS2 HDG-DG scheme: shown are the total water height after (a) 5 days, (b) 10 days, and (c) 15 days. The numerical experiments are performed on a grid with $N_e = 384$ elements, solution order $p = 8$, and $Cr = 1.46$. Contour levels are from 5000 m to 6000 m with 21 levels.	74
3.10	The zonal flow over an isolated mountain: total height field computed from ARS2 HDG-DG (left) at day 15 with Cr of 1.46; the relative difference (right) with RK2 in the height field.	75
3.11	Rossby-Haurwitz wave: the total height field computed from ARS2 HDG-DG after a) 0 days, (b) 7 days, and (c) 14 days. The numerical experiment is performed on the grid with $N_e = 864$ elements, solution order $p = 5$, and $Cr = 1.2$. Contour levels are from 8000 m to 10 600 m with the step size of 173 m.	76
3.12	Rossby-Haurwitz wave: the total height field computed from ARS2 HDG-DG (left) after 14 days with Cr of 1.2; the relative difference (right) with RK2 in the height field.	76
3.13	Barotropic instability test: relative vorticity field of the ARS2 HDG-DG at (a) 4 days, (b) 5 days, and (c) 6 days. The numerical experiment was performed on the grid of $N_e = 5400$ and $p = 4$ with a time-step size of 173 s ($Cr=0.94$). The vorticity ranges from $-1.1 \times 10^{-4} \text{ s}^{-1}$ to $1.8 \times 10^{-4} \text{ s}^{-1}$	77
3.14	Barotropic instability test: relative vorticity fields of RK2 DG (top) and ARS2 HDG-DG (middle) at 6 days with Cr of 0.94 and 0.12, respectively; the difference (bottom) in the vorticity field between ARS2 HDG-DG and RK2 DG.	78
3.15	Inertia-gravity wave: the snapshots of the potential temperature perturbations at $T = [0, 1000, 2000, 3000] \text{ s}$. ARS2 with $dt = 10 \text{ s}$ solutions are plotted on the grid with $(N_{ex}, N_{ez}) = (30, 2)$ elements and solution order $k = 10$. Contour levels range from -0.0015 K to 0.003 K with 13 isosurfaces.	86
3.16	Inertia-gravity wave: the potential temperature perturbations at $T = 3000 \text{ s}$ are plotted for RK4 with $dt = 0.1 \text{ s}$ and ARS2 with $dt = 10 \text{ s}$	87
3.17	The inertia-gravity wave: the potential temperature perturbations are sampled at $z = 5 \text{ km}$ along $x \in [0, 300] \text{ km}$ at $T = 3000 \text{ s}$. ARS2 of $dt = 10 \text{ s}$ shows a good agreement with RK4 of $dt = 0.1 \text{ s}$	87

3.18	Rising thermal bubble: the snapshots of the potential temperature perturbations at $T = [0, 300, 600, 900]$ are plotted for ARS2 HDG-DG with $dt = 0.5(Cr = 52.62)$. The numerical experiment is performed on the grid with $(N_{ex}, N_{ez}) = (10, 10)$ elements and solution order $k = 10$. Contour levels range from 0 K to 0.5 K with 11 isosurfaces.	88
3.19	Rising thermal bubble: RK4 with $dt = 0.005\text{ s}(Cr = 0.5262)$ and ARS2 with $dt = 0.5\text{ s}(Cr = 52.62)$ solutions at $T = 600\text{ s}$	89
3.20	Density current: the potential temperature perturbations is evolved from $t = 0\text{ s}$ to $t = 900\text{ s}$. ARS2 with $dt = 0.2\text{ s}(Cr = 1.7)$ is used for the simulation on the grid with $(N_{ex}, N_{ez}) = (32, 8)$ elements and solution order $k = 8$. Contour levels range from -9.5 K to 0.5 K with 11 isosurfaces.	90
3.21	Density current: RK4 with $dt = 0.04\text{ s}(Cr = 0.378)$ and ARS2 with $dt = 0.2\text{ s}(Cr = 1.7)$ solutions at $t = 900\text{ s}$	91
3.22	Density current: the potential temperature perturbation is sampled at $z = 1.2\text{ km}$ along $x \in [0, 15]\text{ km}$ at $T = 900\text{ s}$	91
4.1	System of two ODEs: (a) magnitude and (b) phase of the exact solution of (4.40a).	104
4.2	System of two ODEs: time convergence study of ETD1-L, ETD1-J and EPI2. We observe the first-order convergence rate for ETD1-L and the second-order convergence rate for ETD1-J and EPI2. The use of Jacobian improves not only the accuracy, but also the stability of ETD1.	104
4.3	System of two ODEs: time convergence study of ARS111-L, ARS111-J and EPI2. ARS111-L shows the first-order convergence rate, whereas ARS111-J shows the first-order convergence rate near 10^{-4} timestep size, but the second-order convergence rate as loosen the timestep size.	105
4.4	Isentropic vortex example to the two-dimensional Euler equation is simulated from $t \in [0, 10]$: (a) RK4 with $Cr = 0.86$, (b) EPI2 with $Cr = 8.56$ and (c) EXPRB42 with $Cr = 85.56$ are chosen for $k = 12$ and $N_e = 256$ spatial discretization.	108
4.5	(a) Uniform mesh with $N_e = 256$ elements and (b) non-uniform meshes with $N_e = 250$ elements.	109
4.6	Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 16$	109
4.7	Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 8$	110
4.8	Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on non-uniform mesh. The computational domain is discretized with $N_e = 250$ and $k = 16$	110
4.9	Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a non-uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 8$	111

4.10	Isentropic vortex translation in three dimensions: snapshots for (a) EPI2 with $Cr = 27.9(dt = 0.25)$. Simulations are performed on the grid with $N_e = 6400$ and $k = 8$ solution order for $t \in [0, 20]$	111
4.11	Isentropic vortex translation in three dimensions: strong scaling study for EPI2 with $Cr = 8.1(dt = 0.03125)$. The computational domain is discretized with $N_e = 3, 276, 800$ and $k = 4$ solution order for $np = \{16, 32, 64, 128, 256, 512, 868\} \times 48$	116
4.12	Isentropic vortex translation in 3D: weak scaling study with fixed Courant numbers. The computational domain is discretized with $N_e = \{1, 8, 64, 512, 4096\} \times 100$ and $k = 8$ solution order for $np = \{4, 32, 256, 2048, 16384\}$, respectively, so that each processor has 25 elements.	118
4.13	Isentropic vortex translation in three dimensions: weak scaling study with fixed timestep size. The computational domain is discretized with $N_e = \{100, 800, 6400, 51200\}$ and $k = 8$ solution order for $np = \{4, 32, 256, 2048\}$, respectively, so that each processor has 25 elements.	119
5.1	Examples of (a) a linear nonconforming mesh and (b) a mortar decomposition. The mortar elements conform to the volume elements K_1 and K_2	121
5.2	Examples of (a) a curved nonconforming mesh and (b) a mortar decomposition. The mortar element m_i is shared by two volume elements K_1 and K_{i+1} for $i = 1, 2$ and 3.	123
5.3	Examples of (a) a conforming element and (b) a nonconforming element.	125
5.4	Examples of (a) a curved conforming mesh and (b) a local refinement. The parent element K_2 in (a) is subdivided into four child elements in (b).	126
5.5	Mapping between physical to reference domain.	128
5.6	A domain is decomposed into the stationary and the rotating regions. Each region is discretized with triangle elements. Isoparametric curved elements are used. Sliding mesh interfaces are constructed between stationary and rotating meshes.	134
5.7	Two-dimensional steady-state transport equation with a discontinuous solution: (a) a curved nonconforming mesh and (b) the solution field, u	136
5.8	Constant solution: error histories are shown for (a) $k = \{3, 4, 5, 6\}$ polynomial order with fixed dt and for (b) $Cr = \{0.01, 0.05, 0.1\}$ with fixed $k = 6$. The freestream is preserved up to rounding error over a curved nonconforming mesh. L^2 errors are measured during $t \in [0, 0.5]$, where the 4th order RK method is used.	137
5.9	Cosine bell test: (a) energy $\frac{u^2}{2}$ history shows the scheme is stable. (b) energy loss is observed at the level of $\mathcal{O}(10^{-10})$. SSPRK3 is used for time integration from $t \in [0, 10]$	138
5.10	Snapshots for the rotating Gaussian bell at $t = \{0, 0.02, 0.04, 0.06\}$. The simulation was conducted with the 6th order polynomial degrees and 4th order RK method of $dt = 10^{-3}$	139

5.11	Coarse grids with (a) linear conforming, (b) linear nonconforming and (c) curved nonconforming meshes.	140
5.12	Convergence results with (a) linear conforming, (b) linear nonconforming and (c) curved nonconforming meshes. L^2 errors are measured after 100 time integrations, where the 4th order RK method is used with $dt = 10^{-6}$	140
5.13	Snapshots of the moving mesh at $t = \{0, 0.1, 0.2, 0.3\}$. The mesh moves with a mapping function of (5.34).	142
5.14	Snapshots of the rotating Gaussian bell at $t = \{0, 0.1, 0.2, 0.3\}$ with $\kappa = 0.01$	142
5.15	Same to Figure 5.14 except $\kappa = 0.1$	142
5.16	Rotating Gaussian with moving mesh: h convergence study for (a) u and (b) q . Backward Euler method is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$ for $\kappa = 0.01$. (Here, p stands for the solution order.)	143
5.17	Snapshots of the rotating mesh at $t = \{0, 0.1, 0.2, 0.3\}$	144
5.18	Rotating Gaussian with a sliding mesh: h -convergence study for (a) u and (b) q . Backward Euler method is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$ for $\kappa = 0.01$. (Here, p stands for the solution order.)	144
5.19	Snapshots of the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$. The domain is decomposed into the rotating mesh (red) and the stationary mesh (black). The blue element is highlighted to emphasize the rotation.	145
5.20	Snapshots of ρ for the isentropic vortex on the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$. The simulation was conducted with the 6th order polynomial degrees and the 4th order RK method of $dt = 10^{-3}$	145
5.21	Snapshots of ρv for the isentropic vortex on the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$	145
5.22	Isentropic vortex with a sliding mesh: h -convergence study for (a) ρ , (b) ρu , (c) ρv and (d) ρE . RK4 is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$. (Here, p stands for the solution order.)	146

Chapter 1

Introduction

The computing power has exponentially grown ever since ENIAC, the first computer was developed. The initial effort to improve its performance has been focused on a single CPU, e.g., increasing clock speed, but nowadays the extreme scale computing power comes from a massively parallel computing system. Due to the trend of modern computer design, numerical methods and algorithms are required to support parallel computing to some extent.

Meanwhile, as computational research has grown, simulation has become a standard tool in many fields of academic and industrial areas. For example, computational fluid dynamics (CFD) tools in aerospace and research facilities are widely used to evaluate the aerodynamic performance of aircraft or wings. Weather forecasts are highly dependent on numerical weather prediction (NWP) model. However, it is still difficult to simulate the complex physical phenomena of a wide range of length and time scales with modern computational resources.

The high-order discontinuous Galerkin (DG) method—originally developed [171, 125, 103] for the neutron transport equation—has been studied extensively for virtually all types of partial differential equations (PDEs) [63, 222, 9, 49, 10, 132, 14, 73, 163, 200, 81, 226]. This is due to the fact that DG combines advantages of finite volume and finite element methods. As such, it is well-suited to problems with large gradients including shocks and with complex geometries, and large-scale simulations demanding parallel implementations. In particular, for numerical modeling of

magma dynamics, the DG methods have been used to study the interaction between the fluid melt and the solid matrix [201, 200], and to include a porosity-dependent bulk viscosity and a solid upwelling effect [186]. Another example is the shallow water equations that describe the motion of a thin layer of incompressible and inviscid fluid. Because they capture essential dynamical characteristics such as nonlinear advection and gravity waves in geophysical flows, they are widely used in oceanography and atmospheric sciences. For the modeling of geophysical flows, spatial discretization using high order discontinuous Galerkin (DG) finite element methods have been of considerable interest [152, 33, 30, 89, 87] due to their flexibility in dealing with complex geometries, high order accuracy, compact stencil, upwind stabilization, etc [52]. In spite of these advantages, DG methods for steady-state and/or time-dependent problems that require implicit time-integrators are more expensive in comparison to other existing numerical methods, since DG typically has many more (coupled) unknowns.

As an effort to mitigate the computational expense associated with DG methods, the hybridized (also known as hybridizable) discontinuous Galerkin (HDG) methods are introduced for various types of PDEs including Poisson-type equation [47, 48, 110, 155, 44, 66], Stokes equation [46, 157], Euler and Navier-Stokes equations, wave equations [161, 147, 159, 129, 158, 92, 55, 175], to name a few. In [30, 32, 34], an upwind HDG framework was introduced that provides a unified and systematic construction of HDG methods for a large class of PDEs.

To fully discretize a time-dependent partial differential equation (PDE), temporal discretization is also necessary. Explicit time integrators such as Runge-Kutta methods are popular due to their simplicity and ease in computer implementation. However, fast waves, such as acoustic/gravity waves, limit the time-step size severely for high-order DG methods (see, e.g., [89]). For long-time integration, which is not uncommon in geophysical fluid dynamics, this can lead to an excessive number of

time steps, and hence substantially taxing computing and storage resources. On the other hand, fully-implicit methods could be expensive, especially for nonlinear PDEs for which Newton-like methods are required. Semi-implicit time-integrators have been designed to balance the time-step size restriction due to fast waves and the computational expense required by nonlinearities [12, 42, 165]. In the context of low-speed fluid flows, including Euler, Navier-Stokes, and shallow water equations, implicit-explicit (IMEX) DG methods have been proposed and proven to be much more advantageous than either explicit or fully-implicit DG methods [70, 174]. The common feature of these methods is that they employ implicit time-stepping schemes for the linear part of the PDE under consideration that contains the fastest waves, and explicit time-integrators for the (resulting) nonlinear part for which the fastest waves are removed. *Unlike standard operator splitting methods, this class of IMEX schemes facilitate high-order solutions both in time and space.* In particular, they provide the flexibility in employing separate high-order discretization methods for the fast linear and for the slow nonlinear operators. There have been studies recently concerning the stability and convergence analysis in the context of discontinuous Galerkin coupled with IMEX methods. In particular, the local discontinuous Galerkin (LDG) based IMEX schemes have been discussed for a linear convection-diffusion system [212] and for a nonlinear convection-diffusion system [61, 213].

Alternatively, exponential time integrators have been received attention due to the positive characteristics such as stability and accuracy. The methods have been applied to various types of PDEs including linear advection-diffusion equations [36], Schrödinger equation [37], Maxwell equations [27, 25], magnetohydrodynamics (MHD) equations [203], Euler equations [131], incompressible Navier-Stokes equations [113], compressible Navier-Stokes equations [187, 130], shallow water equations [43], to name a few.

Exponential time integrators is similar to IMEX methods. The exponential integrators also split the governing equation into linear and nonlinear parts. However, the way to treat the linear part is different. Exponential time integrators exactly integrate the linear part by multiplying an integrating factor. As a result, one should cope with a matrix exponential. Compared to other nonlinear solvers, exponential time integrators are attractive because it does not require any linear solve. A Newton-Krylov method, for example, needs a series of linear solve. Thus, for good scalability and efficiency, a good preconditioner is necessary, but developing an efficient preconditioner is not a trivial task. Especially, showing excellent scalability is challenging. On the other hand, Exponential time integrators are completely free from a linear solve. From the perspective of parallel computing, this method has the potential to be scalable in large-scale simulations. However, there is still a price to pay. Exponential integrators need to compute a matrix exponential, which is computationally demanding for large-scale simulation.

Many researchers have done various studies to deal with the challenge, one way is to use Krylov subspace, where a large matrix is projected onto a small Krylov subspace so that computing the matrix exponential becomes less expensive. To improve the Krylov subspace projection-based algorithm, rational Krylov method [145, 170], restart Krylov method [198, 2], block Krylov method [26, 112], adaptive Krylov method [162] are proposed. Lately, [82] and [133] enhance the computational efficiency of the adaptive Krylov method by replacing the Arnoldi procedure [11] with the incomplete orthogonalization procedure [119, 210]. The work in [202] shows that the exponential propagation iterative (EPI) schemes can be efficient instead of a Newton-Krylov method without a good preconditioner for a large-scale stiff problem. The work in [43] observes the second-order EPI2 gave comparable results to the explicit fourth-order Runge-Kutta (RK4). For elastodynamic problems, exponential

time integrators outperform the backward Euler integrators [144, 143].

Another important topic is mesh refinement methods [140, 114, 60] for accelerating large-scaling simulations. The idea is to initially compute a solution on a coarse grid and then to increase spatial resolution in the region of interests. In finite-element context, we can either increase the solution order (known as p -refinement) or add more elements by subdividing elements on a coarse grid (known as h -refinement). The refinements require communications from a coarse grid to a fine grid (refining), and from a fine grid to a coarse grid (coarsening), which are realized through a numerical flux in the (H) DG spatial discretization.

The mesh refinement technique can be utilized to simulate rotating flows by combining with the Arbitrary Lagrangian-Eulerian (ALE) method [62]. Rotating flows is important across a wide range of applications such as atmospheric and oceanic circulations, hurricane flows, jet engines, wind turbines, vacuum cleaners and so on. For example, for wind turbine applications, it is essential to predict the wake structure accurately and efficiently in a large-scale wind energy project because the wake effects can cause power loss. Many researchers have been simulated the rotating flows using various methods: DG-Fourier solver with sling meshes [71], the sliding-mesh spectral difference [233], the DG-ALE combined with L_2 projection [106], the space-time DG with L_2 projection [218], finite volume (FV) with sliding mesh [195], the DG with an overset grid [109] to name a few.

1.1 Outline

Our main objective in this thesis is to develop a robust, efficient and high-order accurate numerical methods to tackle the challenges.

We first start with the HDG methods for linear degenerate elliptic equations

in Chapter 2, which is arising from the two-phase flows such as mantle convection or glacier dynamics. Since the HDG method provides mortar interfaces on the mesh skeleton, the mortar structure can be used as adhesives to couple the multi-phase regions. We show that the proposed HDG method is well-posed by using an energy approach. We derive *a priori* error estimates for the method on simplicial meshes in both two- and three-dimensions. Several numerical results are presented to confirm the theoretical estimates. We enhance the HDG solutions by post-processing. In Chapter 3, we propose an IMEX HDG-DG framework for shallow water and Euler systems. We start by decomposing the original flux into a linear part (obtained from linearizing the flux at the lake at rest condition) containing the fast wave and a nonlinear part for which the fastest wave is removed. We spatially discretize the former using an HDG method, and the latter using a DG approach. We show various numerical examples to demonstrate the performance of our methods. In Chapter 4 we develop EXPONENTIAL DG scheme, where exponential time integrators are applied to the DG spatial discretization. By adding and subtracting a linear flux based on a flux Jacobian at continuous level, we split a governing equation into stiff and nonstiff terms. Then the system is spatially discretized with the DG method. For accurate and efficient evaluation of matrix exponential, we have used the adaptive Krylov subspace method [162] and its variant [82] and [133]. Numerical results demonstrate that EXPONENTIAL DG methods provide stable solutions with larger timestep sizes as well as promising scalable results. In Chapter 5, we develop a (H) DG sliding-mesh interface for simulating rotating flows by combining a curved nonconforming mesh treatment and ALE formulations. We numerically demonstrate that our approach of sliding-mesh interfaces gives stable and high-order accurate solutions for convection, convection-diffusion and Euler equations. Finally, in Chapter 6 we summarize our work in this thesis and also provide possible directions for future research.

Chapter 2

Degenerate Elliptic equations

In this chapter ¹, we develop a high-order hybridized discontinuous Galerkin (HDG) method for a linear degenerate elliptic equation arising from a two-phase mixture of mantle convection or glacier dynamics. We show that the proposed HDG method is well-posed by using an energy approach. We derive *a priori* error estimates for the method on simplicial meshes in both two- and three-dimensions. The error analysis shows that the convergence rates are optimal for both the scaled pressure and the scaled velocity for non-degenerate problems and are sub-optimal by half order for degenerate ones. Several numerical results are presented to confirm the theoretical estimates. We also enhance the HDG solutions by post-processing. The superconvergence rates of $(k+2)$ and $(k+\frac{3}{2})$ are observed for both a non-degenerate case and a degenerate case away from the degeneracy. Degenerate problems with low regularity solutions are also studied, and numerical results show that high-order methods are beneficial in terms of accuracy.

2.1 Introduction

The Earth's core is hotter than the Earth's surface, which leads to thermal convection in which the cold mantle is dense and sinks while the hot mantle is light and rises to the surface. The induced current, i.e., mantle convection, moves slowly and

¹ The contents of this chapter are largely based on the published manuscript [104]. The contributions of the author in the article ranged from numerical implementation of the algorithm, participation in the theoretical analysis and writing the manuscript.

cools gradually. The evolution and circulation of the mantle induce plate tectonics, volcanic activity, and variation in crustal chemical composition. Therefore, the study of mantle dynamics is critical to understanding how the planet functions [95]. Glacier dynamics, on the other hand, describes the movement of glaciers and ice sheets. Glaciers and ice sheets interact with the atmosphere, the oceans, and the landscape [41], which could lead to a large impact on weather and climate change [107]. Though mantle convection and glacier dynamics are different in nature, their dynamics can be mathematically modeled by the Stokes equations combined with a Darcy equation accounting for melt.

In this study, we are interested in developing numerical methods for both glacial dynamics and mantle convection described by a similar two-phase mathematical model, as we now briefly discuss. In glacial dynamics, the mixture of ice and water is observed near the temperature at the pressure-melting point, which is in a phase-transition process [76, 13]. In mantle dynamics, a partially molten rock is generated by supplying heat or reducing the pressure. In both cases, the relative motion between the melt and the solid matrix is modeled by two-phase flow [141].

We adopt the mathematical model in [142, 141, 188, 189, 18, 17, 38]. In particular, the mixture parameter of fluid melt and solid matrix is described by the porosity ϕ —the relative volume of fluid melt with respect to the bulk volume—which separates the solid single-phase ($\phi = 0$) and fluid-solid two-phase ($\phi > 0$) regions [6]. The partially molten regions are governed by Darcy flow through a deformable solid matrix which is modeled as a highly viscous Stokes fluid [97, 38]. We use the subscript f and s to distinguish between the fluid melt and solid matrix, and boldface lowercase letters for vector-valued functions. We denote by \mathbf{v}_f and \mathbf{v}_s the velocities of fluid and solid, \tilde{p}_f and \tilde{p}_s the pressures, ρ_f and ρ_s the densities, μ_f and μ_s the

viscosities, and σ_f and σ_s the stresses. Darcy's law [56, 141] states

$$\phi(\mathbf{v}_f - \mathbf{v}_s) = -\frac{\kappa(\phi)}{\mu_f}(\nabla \tilde{p}_f - \rho_f \mathbf{g}), \quad (2.1)$$

where $\kappa(\phi)$ is the permeability with $\kappa(0) = 0$ and \mathbf{g} is the gravity. We assume that the solid matrix is more viscous than the fluid melt ($\mu_f \ll \mu_s$) so that the fluid and the solid stresses can be modeled as

$$\sigma_f := -\tilde{p}_f \mathcal{J}, \quad (2.2)$$

$$\sigma_s := -\tilde{p}_s \mathcal{J} + \mu_s (\nabla \mathbf{v}_s + \nabla \mathbf{v}_s^T) - \frac{2}{3} \mu_s \nabla \cdot \mathbf{v}_s \mathcal{J}, \quad (2.3)$$

where \mathcal{J} is the second order identity tensor. The mixture of the melt and the solid matrix obeys the Stokes equation [18, 38]

$$\nabla \cdot (\phi \sigma_f + (1 - \phi) \sigma_s) + (\phi \rho_f + (1 - \phi) \rho_s) \mathbf{g} = 0. \quad (2.4)$$

The mass conservations of the fluid melt and the solid matrix are given as [141, 18]

$$\frac{\partial(\rho_f \phi)}{\partial t} + \nabla \cdot (\rho_f \phi \mathbf{v}_f) = 0, \quad (2.5)$$

$$\frac{\partial(\rho_s(1 - \phi))}{\partial t} + \nabla \cdot (\rho_s(1 - \phi) \mathbf{v}_s) = 0. \quad (2.6)$$

Applying a Boussinesq approximation (constant and equal densities for non-buoyancy terms) to (2.5)–(2.6), the total mass conservation of the mixture can be written as

$$\nabla \cdot (\phi \mathbf{v}_f + (1 - \phi) \mathbf{v}_s) = 0. \quad (2.7)$$

The pressure jump between the melt and the matrix phases (the compaction relation) is given by [193, 17]

$$(\tilde{p}_s - \tilde{p}_f) = -\frac{\mu_s}{\phi} \nabla \cdot \mathbf{v}_s. \quad (2.8)$$

The coupled Darcy-Stokes system (2.1), (2.4), (2.7) and (2.8) describes the motion of the mantle flow (and glacier dynamics). The challenge is when $\phi = 0$. Since solid matrix always exists, the Stokes part is well-posed, but the Darcy part is degenerate when $\phi = 0$.

In this chapter, we shall focus on addressing the challenge of solving the linear degenerate elliptic equation arising from the Darcy part of the system. With a change of variables, (2.1) and a combination of (2.7)–(2.8) become

$$\tilde{\mathbf{v}} + d(\phi)^2(\nabla \tilde{p} - \tilde{\mathbf{g}}) = 0, \quad \text{in } \Omega, \quad (2.9a)$$

$$\nabla \cdot \tilde{\mathbf{v}} + \phi \tilde{p} = \phi^{\frac{1}{2}} f, \quad \text{in } \Omega, \quad (2.9b)$$

$$\phi \tilde{p} = \phi^{\frac{1}{2}} g_D, \quad \text{on } \Gamma_D, \quad (2.9c)$$

where $\Omega \subset \mathbb{R}^{dim}$, $dim = 2$ or 3 , is an open and bounded domain, $\Gamma_D = \partial\Omega$ the Dirichlet boundary, g_D the Dirichlet data, \mathbf{n} the outward unit normal vector, $\tilde{\mathbf{v}} = \phi(\mathbf{v}_f - \mathbf{v}_s)$ the Darcy velocity, $\tilde{p} = \tilde{p}_f$ the fluid pressure, $\tilde{\mathbf{g}} = \rho_f \mathbf{g}$, $f = \phi^{\frac{1}{2}} \tilde{p}_s$, $\mu_s = 1$ and $d(\phi) = \sqrt{\frac{\kappa(\phi)}{\mu_f}}$. Though d is a function of ϕ , we shall write d instead of $d(\phi)$ for the simplicity of the exposition.

The boundary value problem (2.9) has been studied in [7], where the scaled velocity and pressure were proposed in order to obtain well-posedness. For numerical implementation, a cell-centered finite difference method [8] and a mixed finite element method [7] have been studied. The results showed that the numerical schemes are stable and have an optimal convergence rate for smooth solutions. However, these schemes are low order accurate approaches.

The main goal of this chapter is to develop a high-order HDG scheme for the linear degenerate elliptic equation (2.9). In section 2.2, we briefly discuss the scaled system for (2.9). In section 2.3, we derive the HDG formulation for the scaled system based on the upwind HDG framework. The key feature is that we have modified the upwind HDG flux to accommodate the degenerate regions. When the porosity vanishes, the resulting HDG system becomes ill-posed because the upwind

parameter associated with the HDG flux disappears. To overcome the difficulty, we introduce a generalized stabilization parameter that is an extension of the upwind based stabilization parameter. It has positive values on the degenerate interfaces. Next, we show the well-posedness and error analysis of the HDG system under the assumption that the grid well matches with the intersection between the fluid melt and the solid matrix. In section 2.4, various numerical results for the scaled system will be presented to confirm the accuracy and robustness of the proposed HDG scheme. Finally, we conclude the chapter and discuss future research directions in section 2.5.

2.2 Handling the degeneracy

Let $(\cdot, \cdot)_\Omega$ be the L^2 inner-product on Ω , and $\langle \cdot, \cdot \rangle_{\partial\Omega}$ be the L^2 inner-product on $\partial\Omega$. We denote the L^2 norm by $\|\cdot\|_\Omega = (\cdot, \cdot)_\Omega^{\frac{1}{2}}$ on Ω and by $\|\cdot\|_{\partial\Omega} = \langle \cdot, \cdot \rangle_{\partial\Omega}^{\frac{1}{2}}$ on $\partial\Omega$. We also define the weighted L^2 norm on $\partial\Omega$ by $\|\cdot\|_{\partial\Omega, \tau} = \langle |\tau| \cdot, \cdot \rangle_{\partial\Omega}^{\frac{1}{2}} = \left(\int_{\partial\Omega} |\tau| (\cdot)^2 dx \right)^{\frac{1}{2}}$. For any $s \neq 0$, we denote the $H^s(D)$ -norm as $\|\cdot\|_{s,D}$, for example, $\|\cdot\|_{\frac{1}{2}, \partial\Omega}$ is the norm of $H^{\frac{1}{2}}(\partial\Omega)$.

2.2.1 The scaled system

When the porosity becomes zero, the system (2.9) degenerates. However, we can still investigate how the solutions behave as the porosity vanishes. According to [7], a priori energy estimates for the system (2.9) read as

$$\|d^{-1}\tilde{\mathbf{v}}\|_\Omega + \left\| \phi^{\frac{1}{2}}\tilde{p} \right\|_\Omega + \left\| \phi^{-\frac{1}{2}}\nabla \cdot \tilde{\mathbf{v}} \right\|_\Omega \leq c \left(\|g_D\|_{H^{\frac{1}{2}}(\partial\Omega)} + \|d\tilde{\mathbf{g}}\|_\Omega + \|f\|_\Omega \right), \quad (2.10)$$

For some constant $c > 0$. Note that we may lose control of the pressure \tilde{p} as the porosity approaches zero. This is because the fluid pressure \tilde{p} is not defined in the solid regions.

To have the control of the pressure, following [7], we define the scaled velocity

and the scaled pressure as $\mathbf{u} = d^{-1}\tilde{\mathbf{v}}$ and $p = \phi^{\frac{1}{2}}\tilde{p}$, respectively.

The system (2.9) becomes

$$\mathbf{u} + d\nabla\left(\phi^{-\frac{1}{2}}p\right) = d\tilde{\mathbf{g}}, \quad \text{in } \Omega, \quad (2.11a)$$

$$\phi^{-\frac{1}{2}}\nabla \cdot (d\mathbf{u}) + p = f, \quad \text{in } \Omega, \quad (2.11b)$$

$$p = g_D, \quad \text{on } \Gamma_D. \quad (2.11c)$$

Here, we interpret the differential operators in (2.11) as

$$d\nabla(\phi^{-\frac{1}{2}}p) = -\frac{1}{2}\phi^{-\frac{3}{2}}d\nabla\phi p + \phi^{-\frac{1}{2}}d\nabla p, \quad (2.12)$$

$$\phi^{-\frac{1}{2}}\nabla \cdot (d\mathbf{u}) = \phi^{-\frac{1}{2}}\nabla d \cdot \mathbf{u} + \phi^{-\frac{1}{2}}d\nabla \cdot \mathbf{u}, \quad (2.13)$$

where we assume that

$$\phi^{-\frac{1}{2}}d \in L^\infty(\Omega), \quad (2.14a)$$

$$\phi^{-\frac{1}{2}}\nabla d \in (L^\infty(\Omega))^{dim}, \quad (2.14b)$$

$$\phi^{-\frac{3}{2}}d\nabla\phi \in (L^\infty(\Omega))^{dim}. \quad (2.14c)$$

With the assumption (2.14), the scaled system does not degenerate. If the porosity vanishes, then $d(\phi) = 0$, which leads to $\mathbf{u} = 0$ and $p = f$.

The energy estimates for the scaled system (2.11) read as

$$\|u\|_\Omega + \|p\|_\Omega + \left\| \phi^{-\frac{1}{2}}\nabla \cdot (d\mathbf{u}) \right\|_\Omega \leq c \left(\|g_D\|_{H^{\frac{1}{2}}(\partial\Omega)} + \|d\tilde{\mathbf{g}}\|_\Omega + \|f\|_\Omega \right), \quad (2.15)$$

for some constant $c > 0$ [7]. We clearly see that we have control of the scaled pressure p even when the porosity becomes zero.

2.2.2 Upwind-based HDG flux

With some simple manipulation, the scaled system (2.11) can be rewritten as

$$\mathbf{u} - \phi^{-\frac{1}{2}}\nabla dp + \nabla \cdot \left(\phi^{-\frac{1}{2}}dp\mathcal{J} \right) = d\tilde{\mathbf{g}}, \quad \text{in } \Omega, \quad (2.16a)$$

$$\frac{1}{2}\phi^{-\frac{3}{2}}d\nabla\phi \cdot \mathbf{u} + p + \nabla \cdot \left(\phi^{-\frac{1}{2}}d\mathbf{u} \right) = f, \quad \text{in } \Omega. \quad (2.16b)$$

We cast the scaled system (2.16) into the conservative form

$$\nabla \cdot \mathcal{F}(\mathbf{r}) + \mathcal{G}\mathbf{r} = \mathbf{f}, \quad \text{in } \Omega, \quad (2.17)$$

where we have defined the scaled velocity $\mathbf{u} := (u_1, u_2, u_3)$, the solution vector $\mathbf{r} := (u_1, u_2, u_3, p)$, the source vector $\mathbf{f} := (d\tilde{g}_1, d\tilde{g}_2, d\tilde{g}_3, f)$, the flux tensor

$$\mathcal{F} := (F_1, F_2, F_3) := \mathcal{F}(\mathbf{r}) := \phi^{-\frac{1}{2}}d \begin{pmatrix} p & 0 & 0 \\ 0 & p & 0 \\ 0 & 0 & p \\ u_1 & u_2 & u_3 \end{pmatrix} \quad (2.18)$$

and

$$\mathcal{G} := \begin{pmatrix} 1 & 0 & 0 & -\phi^{-\frac{1}{2}}\frac{\partial d}{\partial x} \\ 0 & 1 & 0 & -\phi^{-\frac{1}{2}}\frac{\partial d}{\partial y} \\ 0 & 0 & 1 & -\phi^{-\frac{1}{2}}\frac{\partial d}{\partial z} \\ \frac{1}{2}\phi^{-\frac{3}{2}}d\frac{\partial \phi}{\partial x} & \frac{1}{2}\phi^{-\frac{3}{2}}d\frac{\partial \phi}{\partial y} & \frac{1}{2}\phi^{-\frac{3}{2}}d\frac{\partial \phi}{\partial z} & 1 \end{pmatrix}. \quad (2.19)$$

We define the normal vector $\mathbf{n} := (n_1, n_2, n_3)$ and the flux Jacobian

$$\mathcal{A} = \sum_{k=1}^3 n_k \frac{\partial F_k}{\partial \mathbf{r}} = \phi^{-\frac{1}{2}}d \begin{pmatrix} 0 & 0 & 0 & n_1 \\ 0 & 0 & 0 & n_2 \\ 0 & 0 & 0 & n_3 \\ n_1 & n_2 & n_3 & 0 \end{pmatrix}, \quad (2.20)$$

which has four eigenvalues $(-\phi^{-\frac{1}{2}}d, 0, 0, \phi^{-\frac{1}{2}}d)$ and distinct eigenvectors

$$W_1 = \begin{pmatrix} -n_1 \\ -n_2 \\ -n_3 \\ 1 \end{pmatrix}, W_2 = \begin{pmatrix} -n_2 \\ n_1 \\ 0 \\ 0 \end{pmatrix}, W_3 = \begin{pmatrix} -n_3 \\ 0 \\ n_1 \\ 0 \end{pmatrix}, \text{ and } W_4 = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ 1 \end{pmatrix}. \quad (2.21)$$

The system (2.17) can be considered as a steady state hyperbolic system [205].

Finally, following the upwind HDG framework in [30] we can construct the upwind HDG flux with scalar \hat{p} and vector $\hat{\mathbf{u}}$ trace unknowns as

$$\hat{\mathcal{F}}(\hat{\mathbf{r}}) \cdot \mathbf{n} = \phi^{-\frac{1}{2}}d \begin{pmatrix} n_1 \hat{p} \\ n_2 \hat{p} \\ n_3 \hat{p} \\ \hat{\mathbf{u}} \cdot \mathbf{n} \end{pmatrix} := \mathcal{F}(\mathbf{r}) \cdot \mathbf{n} + |\mathcal{A}|(\mathbf{r} - \hat{\mathbf{r}}) = \phi^{-\frac{1}{2}}d \begin{pmatrix} n_1 (p + (\mathbf{u} - \hat{\mathbf{u}}) \cdot \mathbf{n}) \\ n_2 (p + (\mathbf{u} - \hat{\mathbf{u}}) \cdot \mathbf{n}) \\ n_3 (p + (\mathbf{u} - \hat{\mathbf{u}}) \cdot \mathbf{n}) \\ \mathbf{u} \cdot \mathbf{n} + (p - \hat{p}) \end{pmatrix}, \quad (2.22)$$

where $\hat{\mathbf{r}} = (\hat{u}_1, \hat{u}_2, \hat{u}_3, \hat{p})$, $|\mathcal{A}| := W|D|W^{-1}$, D is the diagonal matrix of eigenvalues of W_1 , W_2 , W_3 and W_4 , and W is the matrix of corresponding eigenvectors. Following [30], we can compute $\hat{\mathbf{u}} \cdot \mathbf{n}$ as a function of \hat{p} , and hence $\hat{\mathbf{u}} \cdot \mathbf{n}$ can be eliminated. The upwind HDG flux can be then written in terms of \hat{p} as

$$\hat{\mathcal{F}}(\hat{\mathbf{r}}) \cdot \mathbf{n} = \phi^{-\frac{1}{2}} d \begin{pmatrix} n_1 \hat{p} \\ n_2 \hat{p} \\ n_3 \hat{p} \\ \mathbf{u} \cdot \mathbf{n} + (p - \hat{p}) \end{pmatrix}. \quad (2.23)$$

2.3 HDG formulation

2.3.1 Finite element definitions and notations

We denote by $\Omega_h := \cup_{i=1}^{N_e} K_i$ the mesh containing a finite collection of non-overlapping elements, K_i , that partition Ω . Here, h is defined as $h := \max_{j \in \{1, \dots, N_e\}} \text{diam}(K_j)$. Let $\partial\Omega_h := \{\partial K : K \in \Omega_h\}$ be the collection of the boundaries of all elements. Let us define $\mathcal{E}_h := \mathcal{E}_h^o \cup \mathcal{E}_h^\partial$ as the skeleton of the mesh which consists of the set of all uniquely defined faces/interfaces, where \mathcal{E}_h^∂ is the set of all boundary faces on $\partial\Omega$, and $\mathcal{E}_h^o = \mathcal{E}_h \setminus \mathcal{E}_h^\partial$ is the set of all interior interfaces. For two neighboring elements K^+ and K^- that share an interior interface $e = K^+ \cap K^-$, we denote by q^\pm the trace of the solutions on e from K^\pm . We define \mathbf{n}^- as the unit outward normal vector on the boundary ∂K^- of element K^- , and $\mathbf{n}^+ = -\mathbf{n}^-$ the unit outward normal of a neighboring element K^+ . On the interior interfaces $e \in \mathcal{E}_h^o$, we define the mean/average operator $\{\{\mathbf{v}\}\}$, where \mathbf{v} is either a scalar or a vector quantify, as $\{\{\mathbf{v}\}\} := (\mathbf{v}^- + \mathbf{v}^+)/2$, and the jump operator $[\mathbf{v} \cdot \mathbf{n}] := \mathbf{v}^+ \cdot \mathbf{n}^+ + \mathbf{v}^- \cdot \mathbf{n}^-$. On the boundary faces $e \in \mathcal{E}_h^\partial$, we define the mean and jump operators as $\{\{\mathbf{v}\}\} := \mathbf{v}$, $[\mathbf{v}] := \mathbf{v}$.

Let $\mathcal{P}^k(D)$ denote the space of polynomials of degree at most k on a domain D . Next, we introduce discontinuous piecewise polynomial spaces for scalars and

vectors as

$$\begin{aligned}
V_h(\Omega_h) &:= \{v \in L^2(\Omega_h) : v|_K \in \mathcal{P}^k(K), \forall K \in \Omega_h\}, \\
\Lambda_h(\mathcal{E}_h) &:= \{\lambda \in L^2(\mathcal{E}_h) : \lambda|_e \in \mathcal{P}^k(e), \forall e \in \mathcal{E}_h\}, \\
\mathbf{V}_h(\Omega_h) &:= \{\mathbf{v} \in [L^2(\Omega_h)]^m : \mathbf{v}|_K \in [\mathcal{P}^k(K)]^m, \forall K \in \Omega_h\}, \\
\mathbf{\Lambda}_h(\mathcal{E}_h) &:= \{\mathbf{\lambda} \in [L^2(\mathcal{E}_h)]^m : \mathbf{\lambda}|_e \in [\mathcal{P}^k(e)]^m, \forall e \in \mathcal{E}_h\}.
\end{aligned}$$

and similar spaces $V_h(K)$, $\Lambda_h(e)$, $\mathbf{V}_h(K)$, and $\mathbf{\Lambda}_h(e)$ by replacing Ω_h with K and \mathcal{E}_h with e . Here, m is the number of components of the vector under consideration.

We define the broken inner products as $(\cdot, \cdot)_{\Omega_h} := \sum_{K \in \Omega_h} (\cdot, \cdot)_K$ and $\langle \cdot, \cdot \rangle_{\partial\Omega_h} := \sum_{K \in \partial\Omega_h} \langle \cdot, \cdot \rangle_{\partial K}$, and on the mesh skeleton as $\langle \cdot, \cdot \rangle_{\mathcal{E}_h} := \sum_{e \in \mathcal{E}_h} \langle \cdot, \cdot \rangle_e$. We also define the associated norms as $\|\cdot\|_{\Omega_h} := (\sum_{K \in \Omega_h} \|\cdot\|_K^2)^{\frac{1}{2}}$, $\|\cdot\|_{\partial\Omega_h} := (\sum_{K \in \Omega_h} \|\cdot\|_{\partial K}^2)^{\frac{1}{2}}$, and the weighted norm $\|\cdot\|_{\partial\Omega_h, \tau} := (\sum_{K \in \Omega_h} \|\cdot\|_{\partial K, \tau}^2)^{\frac{1}{2}}$ (recall $\|\cdot\|_{\partial K, \tau} = \left\| |\tau|^{\frac{1}{2}}(\cdot) \right\|_{\partial K}$).

2.3.2 Weak form

From now on, we conventionally use \mathbf{u}^e , p^e and \hat{p}^e for the exact solution while \mathbf{u} , p and \hat{p} are used to denote the HDG solution.

Unlike the DG approach, in which \hat{p} on an interface is computed using information from neighboring elements that share that interface, i.e.,

$$\hat{p} = \frac{1}{2} [\mathbf{u} \cdot \mathbf{n}] + \{\{p\}\}, \quad (2.24)$$

the idea behind HDG is to treat \hat{p} as a new unknown. Testing (2.16) or (2.17) with (\mathbf{v}, q) and integrating by parts we obtain the local solver for each element by replacing the flux $\langle \mathcal{F} \cdot \mathbf{n}, (\mathbf{v}, q) \rangle_{\partial K}$ with the HDG numerical flux $\left\langle \hat{\mathcal{F}} \cdot \mathbf{n}, (\mathbf{v}, q) \right\rangle_{\partial K}$. The local

solver reads: find $(\mathbf{u}, p, \hat{p}) \in \mathbf{V}_h(K) \times V_h(K) \times \Lambda_h(\partial K)$ such that

$$\begin{aligned} & (\mathbf{u}, \mathbf{v})_K - \left(\phi^{-\frac{1}{2}} \nabla dp, \mathbf{v} \right)_K \\ & - \left(\phi^{-\frac{1}{2}} dp, \nabla \cdot \mathbf{v} \right)_K + \left\langle \phi^{-\frac{1}{2}} d\hat{p}, \mathbf{v} \cdot \mathbf{n} \right\rangle_{\partial K} = (d\tilde{\mathbf{g}}, \mathbf{v})_K, \end{aligned} \quad (2.25a)$$

$$\begin{aligned} & (p, q)_K + \left(\frac{1}{2} \phi^{-\frac{3}{2}} d \nabla \phi \cdot \mathbf{u}, q \right)_K - \left(\phi^{-\frac{1}{2}} d\mathbf{u}, \nabla q \right)_K \\ & + \left\langle \phi^{-\frac{1}{2}} d(\mathbf{u} \cdot \mathbf{n} + (p - \hat{p})), q \right\rangle_{\partial K} = (f, q)_K, \end{aligned} \quad (2.25b)$$

for all $(\mathbf{v}, q) \in \mathbf{V}_h(K) \times V_h(K)$.

Clearly we need an additional equation to close the system since we have introduced an additional trace unknown \hat{p} . The natural condition is the conservation, that is, the continuity of the HDG flux. For the HDG method to be conservative, it is sufficient to weakly enforce the continuity of the last component of the HDG flux (2.22) on each face e of the mesh skeleton, i.e.,

$$\left\langle \left[\left[\phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \phi^{-\frac{1}{2}} d(p - \hat{p}) \right] \right], \hat{q} \right\rangle_e = 0, \quad \forall e \in \mathcal{E}_h^o. \quad (2.26)$$

On degenerate faces, where $\phi = 0$, the conservation condition (2.26) is trivially satisfied. These faces would need to be sorted out and removed from the system. However, this creates implementation issues. To avoid this, we introduce a more general HDG flux

$$\hat{\mathcal{F}} \cdot \mathbf{n} := \begin{pmatrix} n_1 \phi^{-\frac{1}{2}} d\hat{p} \\ n_2 \phi^{-\frac{1}{2}} d\hat{p} \\ n_3 \phi^{-\frac{1}{2}} d\hat{p} \\ \phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - \hat{p}) \end{pmatrix}, \quad (2.27)$$

where τ is a positive function on the edge. For example, we can take $\tau = \phi^{-\frac{1}{2}} d$ for non-degenerate faces (i.e., faces with $\phi > 0$), and for degenerate ones (i.e. faces with $\phi = 0$) we take $\tau = \gamma > 0$. Alternatively, we can take a single value $\tau = \mathcal{O}(1/h)$ over the entire mesh skeleton. We shall compare these choices in Section 2.4. With this

HDG flux, the conservation condition (2.26) becomes

$$\left\langle \left[\left[\phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - \hat{p}) \right] \right], \hat{q} \right\rangle_e = 0, \quad \forall e \in \mathcal{E}_h^o, \quad \forall \hat{q} \in \Lambda_h(e). \quad (2.28)$$

On the Dirichlet boundary Γ_D , we impose the boundary data g_D to \hat{p} through the weak form of

$$\langle \tau \hat{p}, \hat{q} \rangle_{\Gamma_D} = \langle \tau g_D, \hat{q} \rangle_{\Gamma_D}, \quad \forall \hat{q} \in \Lambda_h(\Gamma_D). \quad (2.29)$$

With the general HDG flux (2.27) and Dirichlet boundary condition (2.11c), the local equation (2.25) now becomes

$$\begin{aligned} & (\mathbf{u}, \mathbf{v})_K - \left(\phi^{-\frac{1}{2}} \nabla dp, \mathbf{v} \right)_K - \left(\phi^{-\frac{1}{2}} dp, \nabla \cdot \mathbf{v} \right)_K \\ & + \left\langle \phi^{-\frac{1}{2}} d\hat{p}, \mathbf{v} \cdot \mathbf{n} \right\rangle_{\partial K \setminus \Gamma_D} + \left\langle \phi^{-\frac{1}{2}} dg_D, \mathbf{v} \cdot \mathbf{n} \right\rangle_{\partial K \cap \Gamma_D} = (d\tilde{\mathbf{g}}, \mathbf{v})_K, \end{aligned} \quad (2.30a)$$

$$\begin{aligned} & (p, q)_K + \left(\frac{1}{2} \phi^{-\frac{3}{2}} d\nabla \phi \cdot \mathbf{u}, q \right)_K - \left(\phi^{-\frac{1}{2}} d\mathbf{u}, \nabla q \right)_K \\ & + \left\langle \phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - \hat{p}), q \right\rangle_{\partial K \setminus \Gamma_D} \\ & + \left\langle \phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - g_D), q \right\rangle_{\partial K \cap \Gamma_D} = (f, q)_K. \end{aligned} \quad (2.30b)$$

The HDG comprises the local solver (2.30), the global equation (2.28) and the boundary condition (2.29). By summing (2.30) over all elements and (2.28) over the mesh skeleton, we obtain the complete HDG system with the weakly imposed Dirichlet boundary condition (2.29): find $(\mathbf{u}, p, \hat{p}) \in \mathbf{V}_h(\Omega_h) \times V_h(\Omega_h) \times \Lambda_h(\mathcal{E}_h)$ such

that

$$\begin{aligned} (\mathbf{u}, \mathbf{v})_{\Omega_h} - \left(\phi^{-\frac{1}{2}} \nabla dp, \mathbf{v} \right)_{\Omega_h} - \left(\phi^{-\frac{1}{2}} dp, \nabla \cdot \mathbf{v} \right)_{\Omega_h} + \left\langle \phi^{-\frac{1}{2}} d\hat{p}, \mathbf{v} \cdot \mathbf{n} \right\rangle_{\partial\Omega_h \setminus \Gamma_D} \\ = (d\tilde{\mathbf{g}}, \mathbf{v})_{\Omega_h} - \left\langle g_D, \phi^{-\frac{1}{2}} d\mathbf{v} \cdot \mathbf{n} \right\rangle_{\Gamma_D}, \end{aligned} \quad (2.31a)$$

$$\begin{aligned} (p, q)_{\Omega_h} + \left(\frac{1}{2} \phi^{-\frac{3}{2}} d\nabla \phi \cdot \mathbf{u}, q \right)_{\Omega_h} - \left(\phi^{-\frac{1}{2}} d\mathbf{u}, \nabla q \right)_{\Omega_h} \\ + \left\langle \phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - \hat{p}), q \right\rangle_{\partial\Omega_h \setminus \Gamma_D} \\ + \left\langle \phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau p, q \right\rangle_{\Gamma_D} = (f, q)_{\Omega_h} + \langle \tau g_D, q \rangle_{\Gamma_D}, \end{aligned} \quad (2.31b)$$

$$- \left\langle \left[\left[\phi^{-\frac{1}{2}} d\mathbf{u} \cdot \mathbf{n} + \tau(p - \hat{p}) \right] \right], \hat{q} \right\rangle_{\mathcal{E}_h \setminus \Gamma_D} + \langle \tau \hat{p}, \hat{q} \rangle_{\Gamma_D} = \langle \tau g_D, \hat{q} \rangle_{\Gamma_D}, \quad (2.31c)$$

for all $(\mathbf{v}, q, \hat{q}) \in \mathbf{V}_h(\Omega_h) \times V_h(\Omega_h) \times \Lambda_h(\mathcal{E}_h)$. Note that this form resembles the weak Galerkin framework [216, 217, 232, 148]. Indeed, HDG and the weak Galerkin method are equivalent in this case.

The HDG computation consists of three steps: first, solve the local solver for (\mathbf{u}, p) as a function of \hat{p} element-by-element, completely independent of each other; second, substitute (\mathbf{u}, p) into the global equation (2.28) to solve for \hat{p} on the mesh skeleton; and finally recover the local volume unknown (\mathbf{u}, p) in parallel.

2.3.3 Well-posedness

Let us denote the bilinear form on the left hand side of (2.31) as $a((\mathbf{u}, p, \hat{p}); (\mathbf{v}, q, \mu))$ and the linear form on right hand side as $\ell((\mathbf{v}, q, \mu))$. We begin with an energy estimate for the HDG solution.

Proposition 1 (Discrete energy estimate). Suppose $g_D \in L^2(\Gamma_D)$, $f \in L^2(\Omega_h)$, and $d(\phi)\tilde{\mathbf{g}} \in L^2(\Omega_h)$. If $\tau = \mathcal{O}(1/h)$, then it holds that

$$a((\mathbf{u}, p, \hat{p}); (\mathbf{u}, p, \hat{p})) = \|\mathbf{u}\|_{\Omega_h}^2 + \|p\|_{\Omega_h}^2 + \|\hat{p}\|_{\Gamma_D, \tau}^2 + \|p\|_{\Gamma_D, \tau}^2 + \|p - \hat{p}\|_{\partial\Omega_h \setminus \Gamma_D, \tau}^2 \quad (2.32)$$

$$\leq c \left(\|g_D\|_{\Gamma_D, \tau}^2 + \|\tilde{\mathbf{g}}\|_{\Omega_h}^2 + \|f\|_{\Omega_h}^2 \right), \quad (2.33)$$

for some positive constant $c = c(\phi, d, \tau, h, k)$. In particular, there is a unique solution (\mathbf{u}, p, \hat{p}) to the HDG system (2.31).

Proof. We start with the following identities

$$-\left(\phi^{-\frac{1}{2}}\nabla dp, \mathbf{v}\right)_K - \left(\phi^{-\frac{1}{2}}dp, \nabla \cdot \mathbf{v}\right)_K = -\left(p, \phi^{-\frac{1}{2}}\nabla \cdot (d\mathbf{v})\right)_K, \quad (2.34a)$$

$$\begin{aligned} & \left(\frac{1}{2}\phi^{-\frac{3}{2}}d\nabla\phi \cdot \mathbf{u}, q\right)_K - \left(\phi^{-\frac{1}{2}}d\mathbf{u}, \nabla q\right)_K \\ &= \left(\phi^{-\frac{1}{2}}\nabla \cdot (d\mathbf{u}), q\right)_K - \left\langle \phi^{-\frac{1}{2}}d\mathbf{u} \cdot \mathbf{n}, q \right\rangle_{\partial K}. \end{aligned} \quad (2.34b)$$

Now taking $\mathbf{v} = \mathbf{u}$, $q = p$, and $\hat{q} = \hat{p}$ in (2.31) and (2.34), and then adding all equations in (2.31) gives

$$\begin{aligned} a((\mathbf{u}, p, \hat{p}); (\mathbf{u}, p, \hat{p})) &= \|\mathbf{u}\|_{\Omega_h}^2 + \|p\|_{\Omega_h}^2 + \|\hat{p}\|_{\Gamma_D, \tau}^2 + \|p\|_{\Gamma_D, \tau}^2 + \|p - \hat{p}\|_{\partial\Omega_h \setminus \Gamma_D, \tau}^2 = \\ &= -\left\langle g_D, \phi^{-\frac{1}{2}}d\mathbf{u} \cdot \mathbf{n} \right\rangle_{\Gamma_D} - \langle \tau g_D, p \rangle_{\Gamma_D} + \langle \tau g_D, \hat{p} \rangle_{\Gamma_D} + (d\tilde{\mathbf{g}}, \mathbf{u})_{\Omega_h} + (f, p)_{\Omega_h}, \end{aligned}$$

which, after invoking the Cauchy-Schwarz and Young inequalities, becomes

$$\begin{aligned} a((\mathbf{u}, p, \hat{p}); (\mathbf{u}, p, \hat{p})) &\leq \frac{\|\phi^{-\frac{1}{2}}d\|_{\infty}}{2\varepsilon_1} \|g_D\|_{\Gamma_D, \tau}^2 + \frac{\varepsilon_1}{2} \|\mathbf{u}\|_{\Gamma_D, \tau^{-1}}^2 \\ &+ \frac{1}{2\varepsilon_2} \|g_D\|_{\Gamma_D, \tau}^2 + \frac{\varepsilon_2}{2} \|p - \hat{p}\|_{\Gamma_D, \tau}^2 + \frac{1}{2\varepsilon_3} \|d\tilde{\mathbf{g}}\|_{\Omega_h}^2 + \frac{\varepsilon_3}{2} \|\mathbf{u}\|_{\Omega_h}^2 + \frac{1}{2\varepsilon_4} \|f\|_{\Omega_h}^2 + \frac{\varepsilon_4}{2} \|p\|_{\Omega_h}^2, \end{aligned}$$

which yields the desired energy estimate after applying an inverse trace inequality (c.f. Lemma (2.53)) for the second term on right hand side and choosing sufficiently small values for $\varepsilon_1, \varepsilon_2, \varepsilon_3$ and ε_4 . \square

Since the HDG system (2.31) is linear and square in terms of the HDG variables (\mathbf{u}, p, \hat{p}) , the uniqueness result in Proposition 1 implies existence and stability, and hence the well-posedness of the HDG system.

Lemma 1 (Consistency). Suppose (\mathbf{u}^e, p^e) is a weak solution of (2.11), which is sufficiently regular. Then $(\mathbf{u}^e, p^e, p^e|_{\mathcal{E}_h})$ satisfies the HDG formulation (2.31). In particular, the Galerkin orthogonality holds, i.e.,

$$a((\mathbf{u}^e - \mathbf{u}, p^e - p, p^e|_{\mathcal{E}_h} - \hat{p}); (\mathbf{v}, q, \hat{q})) = 0, \quad \forall (\mathbf{v}, q, \hat{q}) \in \mathbf{V}_h(\Omega_h) \times V_h(\Omega_h) \times \Lambda_h(\mathcal{E}_h). \quad (2.35)$$

The proof is a simple application of integration by parts and hence omitted.

2.3.4 Error analysis

We restrict the analysis for simplicial meshes and adopt the projection-based error analysis in [48]. To begin, we define \hat{p}^e as the trace of p^e . For any element K , $e \in \mathcal{E}_h$, $e \subset \partial K$, we denote by $\mathbf{P}(\mathbf{u}^e, p^e, \hat{p}^e) := (\mathbb{P}\mathbf{u}^e, \mathbb{P}p^e, \Pi\hat{p}^e)$, where Π is the standard L^2 -projection, a collective projection of the exact solution. Let us define

$$\boldsymbol{\varepsilon}_{\mathbf{u}}^I := \mathbf{u}^e - \mathbb{P}\mathbf{u}^e, \quad \boldsymbol{\varepsilon}_{\mathbf{u}}^h := \mathbf{u} - \mathbb{P}\mathbf{u}^e, \quad (2.36)$$

$$\varepsilon_p^I := p^e - \mathbb{P}p^e, \quad \varepsilon_p^h := p - \mathbb{P}p^e, \quad (2.37)$$

$$\varepsilon_{\hat{p}}^I := \hat{p}^e - \Pi\hat{p}^e, \quad \varepsilon_{\hat{p}}^h := \hat{p} - \Pi\hat{p}^e, \quad (2.38)$$

and then the projections $\mathbb{P}\mathbf{u}^e$ and $\mathbb{P}p^e$ are defined by

$$(\boldsymbol{\varepsilon}_{\mathbf{u}}^I, \mathbf{v})_K = 0, \quad \mathbf{v} \in [\mathcal{P}_{k-1}(K)]^{dim}, \quad (2.39a)$$

$$(\varepsilon_p^I, q)_K = 0, \quad q \in \mathcal{P}_{k-1}(K), \quad (2.39b)$$

$$\langle \alpha \boldsymbol{\varepsilon}_{\mathbf{u}}^I \cdot \mathbf{n} + \tau \varepsilon_p^I, \hat{q} \rangle_e = 0, \quad \hat{q} \in \mathcal{P}_k(e), \quad (2.39c)$$

for each $K \in \Omega_h$, $e \in \mathcal{E}_h$ and $e \subset \partial K$. Here α , to be defined below, is a positive constant on each face e of element K .

Lemma 2. Let $\tau_K := \tau/\alpha$. The projections $\mathbb{P}\mathbf{u}^e$ and $\mathbb{P}p^e$ are well-defined, and

$$\begin{aligned}\|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_K + h \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{1,K} &\leq ch^{k+1} \|\mathbf{u}^e\|_{k+1,K} + ch^{k+1} \tau_K^* \|p^e\|_{k+1,K}, \\ \|\varepsilon_p^I\|_K + h \|\varepsilon_p^I\|_{1,K} &\leq c \frac{h^{k+1}}{\tau_K^{\max}} \|\nabla \cdot \mathbf{u}^e\|_{k,K} + ch^{k+1} \|p^e\|_{k+1,K},\end{aligned}$$

where $\tau_K^{\max} := \max \tau_K|_{\partial K}$ and $\tau_K^* := \tau_K|_{\partial K \setminus e^*}$, where e^* is the edge on which τ_K is maximum.

The proof can be obtained from [48].

Since the interpolation errors $\boldsymbol{\varepsilon}_{\mathbf{u}}^I, \varepsilon_p^I$ and $\varepsilon_{\hat{p}}^I$ have optimal convergence order, by the triangle inequality, the convergent rates of the total errors $\boldsymbol{\varepsilon}_{\mathbf{u}} = \boldsymbol{\varepsilon}_{\mathbf{u}}^I + \boldsymbol{\varepsilon}_{\mathbf{u}}^h, \varepsilon_p = \varepsilon_p^I + \varepsilon_p^h$, and $\varepsilon_{\hat{p}} = \varepsilon_{\hat{p}}^I + \varepsilon_{\hat{p}}^h$ are determined by those of the discretization errors $\boldsymbol{\varepsilon}_{\mathbf{u}}^h, \varepsilon_p^h$ and $\varepsilon_{\hat{p}}^h$. We use an energy approach to estimate the discretization errors. To begin, let us define

$$\mathcal{E}_h^2 := \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}^2 + \|\varepsilon_p^h\|_{\Omega_h}^2 + \|\varepsilon_{\hat{p}}^h\|_{\Gamma_D, \tau}^2 + \|\varepsilon_p^h\|_{\Gamma_D, \tau}^2 + \|\varepsilon_p^h - \varepsilon_{\hat{p}}^h\|_{\partial\Omega_h \setminus \Gamma_D, \tau}^2.$$

Lemma 3 (Error equation). It holds that

$$\begin{aligned}\mathcal{E}_h^2 = & \underbrace{-\left(\phi^{-\frac{1}{2}}\varepsilon_p^I, \nabla \cdot (d\boldsymbol{\varepsilon}_{\mathbf{u}}^h)\right)_{\Omega_h}}_A + \underbrace{\left\langle \phi^{-\frac{1}{2}}d\varepsilon_{\hat{p}}^I, \boldsymbol{\varepsilon}_{\mathbf{u}}^h \cdot \mathbf{n} \right\rangle_{\partial\Omega_h \setminus \Gamma_D}}_B - \underbrace{\left(d\boldsymbol{\varepsilon}_{\mathbf{u}}^I, \nabla \left(\phi^{-\frac{1}{2}}\varepsilon_p^h\right)\right)_{\Omega_h}}_C \\ & + \underbrace{\left\langle \left(\phi^{-\frac{1}{2}}d - \alpha\right) \boldsymbol{\varepsilon}_{\mathbf{u}}^I \cdot \mathbf{n}, \varepsilon_p^h - \varepsilon_{\hat{p}}^h \right\rangle_{\partial\Omega_h \setminus \Gamma_D} + \left\langle \left(\phi^{-\frac{1}{2}}d - \alpha\right) \boldsymbol{\varepsilon}_{\mathbf{u}}^I \cdot \mathbf{n}, \varepsilon_p^h \right\rangle_{\Gamma_D}}_D\end{aligned}\quad (2.40)$$

Proof. The proof is straightforward by first adding and subtracting appropriate projections in the Galerkin orthogonality equation (2.35), second using the definition of the projections (2.39), and finally taking $\mathbf{v} = \boldsymbol{\varepsilon}_{\mathbf{u}}^h$, $q = \varepsilon_p^h$, and $\hat{q} = \varepsilon_{\hat{p}}^h$. \square

The next step is to estimate A, B, C and D . To that end, we define α on faces of an element K as

$$\alpha := \begin{cases} \overline{\phi^{-\frac{1}{2}}d} & \text{if } \overline{\phi^{-\frac{1}{2}}d} \neq 0 \\ 1 & \text{otherwise} \end{cases}, \quad (2.41)$$

where $\overline{\phi^{-\frac{1}{2}}d}$ is the average of $\phi^{-\frac{1}{2}}d$ on the element K .

Lemma 4 (Estimation for A). There exists a positive constant $c = c(\phi, d)$ such that

$$|A| \leq c \|\varepsilon_p^I\|_{\Omega_h} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}.$$

Proof. We have

$$|A| \leq \left| \left(\varepsilon_p^I, \phi^{-\frac{1}{2}} \nabla d \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h \right)_{\Omega_h} \right| + \left| \left(\varepsilon_p^I, \phi^{-\frac{1}{2}} d \nabla \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h \right)_{\Omega_h} \right|$$

Bounding the first term is straightforward:

$$\left| \left(\varepsilon_p^I, \phi^{-\frac{1}{2}} \nabla d \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h \right)_{\Omega_h} \right| \leq c \left\| \phi^{-\frac{1}{2}} \nabla d \right\|_{\infty} \|\varepsilon_p^I\|_{\Omega_h} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}.$$

For the second term, we have

$$\begin{aligned} \left| \left(\varepsilon_p^I, \phi^{-\frac{1}{2}} d \nabla \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h \right)_{\Omega_h} \right| &= \left| \left(\varepsilon_p^I, \left(\phi^{-\frac{1}{2}} d - \overline{\phi^{-\frac{1}{2}}d} \right) \nabla \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h \right)_{\Omega_h} \right| \\ &\leq ch \|\varepsilon_p^I\|_{\Omega_h} \left\| \phi^{-\frac{1}{2}} d \right\|_{W^{1,\infty}(\Omega_h)} \|\nabla \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h} \\ &\leq c \|\varepsilon_p^I\|_{\Omega_h} \left\| \phi^{-\frac{1}{2}} d \right\|_{W^{1,\infty}(\Omega_h)} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}, \end{aligned}$$

where we have used (2.39b) in the first equality, the Cauchy-Schwarz inequality and the Bramble–Hilbert lemma (see, e.g., [29]) in the first inequality, and Lemma 8 (in the appendix) in the last inequality. Here, $W^{1,\infty}$ is a standard Sobolev space. \square

Lemma 5 (Estimation for B). There exists a positive constant $c = c(\phi, d)$ such that

$$|B| \leq ch^{\frac{1}{2}} \|\varepsilon_p^I\|_{\partial\Omega_h} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}.$$

Proof. We have

$$\begin{aligned} |B| &= \left| \left\langle \varepsilon_p^I, \left(\phi^{-\frac{1}{2}} d - \overline{\phi^{-\frac{1}{2}}d} \right) \boldsymbol{\varepsilon}_{\mathbf{u}}^h \cdot \mathbf{n} \right\rangle_{\partial\Omega_h \setminus \Gamma_D} \right| \leq \|\varepsilon_p^I\|_{\partial\Omega_h} \left\| \phi^{-\frac{1}{2}} d - \overline{\phi^{-\frac{1}{2}}d} \right\|_{L^\infty(\partial\Omega_h)} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\partial\Omega_h} \\ &\leq ch \left\| \phi^{-\frac{1}{2}} d \right\|_{W^{1,\infty}(\Omega_h)} \|\varepsilon_p^I\|_{\partial\Omega_h} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\partial\Omega_h}, \end{aligned}$$

where we have used the property of L^2 -projection $\Pi \widehat{p^e}$ in the first equality, the Cauchy-Schwarz inequality in the first inequality, and the Bramble–Hilbert lemma in the last inequality. Now the best approximation of $\Pi \widehat{p^e}$ implies $\|\varepsilon_{\widehat{p}}^I\|_{\partial\Omega_h} \leq \|\varepsilon_p^I\|_{\partial\Omega_h}$ and (2.56) gives the result. \square

Lemma 6 (Estimation for C). There exists a positive constant $c = c(\phi, d)$ such that

$$|C| \leq c \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\Omega_h} \|\varepsilon_p^h\|_{\Omega_h}.$$

Proof. We have

$$|C| \leq \left| \left(\frac{1}{2} \phi^{-\frac{3}{2}} d \nabla \phi \cdot \boldsymbol{\varepsilon}_{\mathbf{u}}^I, \varepsilon_p^h \right)_{\Omega_h} \right| + \left| \left(\boldsymbol{\varepsilon}_{\mathbf{u}}^I, \phi^{-\frac{1}{2}} d \nabla \varepsilon_p^h \right)_{\Omega_h} \right|.$$

The rest of the proof is similar to that of Lemma 4 by using (2.39a). \square

Lemma 7 (Estimation for D). There exists a positive constant $c = c(\phi, d)$ such that

$$|D| \leq c\beta \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\partial\Omega_h, \tau^{-1}} \left(\|\varepsilon_p^h - \varepsilon_{\widehat{p}}^h\|_{\partial\Omega_h \setminus \Gamma_{D, \tau}} + \|\varepsilon_p^h\|_{\Gamma_{D, \tau}} \right),$$

where

$$\beta := \begin{cases} h & \text{if } \overline{\phi^{-\frac{1}{2}} d} \neq 0 \quad \forall K \in \Omega_h \\ 1 & \text{otherwise} \end{cases}.$$

Proof. Employing similar techniques as in estimating B , we have

$$|D| \leq \left\| \phi^{-\frac{1}{2}} d - \alpha \right\|_{L^\infty(\partial\Omega_h)} \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\partial\Omega_h, \tau^{-1}} \left(\|\varepsilon_p^h - \varepsilon_{\widehat{p}}^h\|_{\partial\Omega_h \setminus \Gamma_{D, \tau}} + \|\varepsilon_p^h\|_{\Gamma_{D, \tau}} \right).$$

Now using the definition of α in (2.41) and the Bramble–Hilbert lemma,

$$\left\| \phi^{-\frac{1}{2}} d - \alpha \right\|_{L^\infty(\partial\Omega_h)} \leq \left\| \phi^{-\frac{1}{2}} d - \alpha \right\|_{L^\infty(\Omega_h)} \leq c\beta,$$

and this ends the proof. \square

Now comes the main result of this section.

Theorem 1. Suppose $\mathbf{u}^e \in [H^{k+1}(\Omega_h)]^{dim}$ and $p^e \in H^{k+1}(\Omega_h)$. Then

$$\begin{aligned} & \|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h} + \|\varepsilon_p^h\|_{\Omega_h} + \|\varepsilon_{\hat{p}}^h\|_{\Gamma_{D,\tau}} + \|\varepsilon_p^h\|_{\Gamma_{D,\tau}} + \|\varepsilon_p^h - \varepsilon_{\hat{p}}^h\|_{\partial\Omega_h \setminus \Gamma_{D,\tau}} \\ & \leq c \left(\|\mathbf{u}^e\|_{k+1,\Omega_h} + \|p^e\|_{k+1,\Omega_h} \right) \times \begin{cases} h^{k+1} & \text{if } \phi^{-\frac{1}{2}}d \neq 0 \\ h^{k+\frac{1}{2}} & \text{otherwise} \end{cases} \quad \forall K \in \Omega_h, \end{aligned}$$

where $c = c(\phi, d, \tau)$ is a positive constant independent of h .

Proof. Using the results in Lemmas 3—7 and the Cauchy-Schwarz inequality, we have

$$\begin{aligned} \mathcal{E}_h^2 & \leq c \left(\|\varepsilon_p^I\|_{\Omega_h}^2 + \beta h \|\varepsilon_p^I\|_{\partial\Omega_h}^2 + \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\Omega_h}^2 + \beta \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\partial\Omega_h, \tau^{-1}}^2 \right)^{\frac{1}{2}} \times \\ & \quad \left(\|\boldsymbol{\varepsilon}_{\mathbf{u}}^h\|_{\Omega_h}^2 + \|\varepsilon_p^h\|_{\Omega_h}^2 + \|\varepsilon_p^h - \varepsilon_{\hat{p}}^h\|_{\partial\Omega_h \setminus \Gamma_{D,\tau}}^2 + \|\varepsilon_p^h\|_{\Gamma_{D,\tau}}^2 \right)^{\frac{1}{2}} \\ & \leq c \left(\|\varepsilon_p^I\|_{\Omega_h}^2 + \beta h \|\varepsilon_p^I\|_{\partial\Omega_h}^2 + \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\Omega_h}^2 + \beta \|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\partial\Omega_h, \tau^{-1}}^2 \right)^{\frac{1}{2}} \times \mathcal{E}_h. \quad (2.42) \end{aligned}$$

The estimate for $\|\varepsilon_p^I\|_{\Omega_h}^2$ and $\|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\Omega_h}^2$ can be obtained directly from Lemma 2. Now using Lemma 9 in the appendix and approximation properties of $\mathbb{P}\mathbf{u}^e, \mathbb{P}p^e$ in Lemma 2 gives

$$\begin{aligned} \|\varepsilon_p^I\|_{\partial\Omega_h}^2 & \leq c \sum_K \left(\|\nabla \varepsilon_p^I\|_{0,K} + h^{-1} \|\varepsilon_p^I\|_{0,K} \right) \|\varepsilon_p^I\|_{0,K} \\ & \leq ch^{2k+1} \left(\max_K \frac{1}{\tau_K^{\max}} \|\mathbf{u}^e\|_{k+1,\Omega_h} + \|p^e\|_{k+1,\Omega_h} \right)^2. \quad (2.43) \end{aligned}$$

Similarly we can obtain

$$\|\boldsymbol{\varepsilon}_{\mathbf{u}}^I\|_{\partial\Omega_h, \tau^{-1}}^2 \leq ch^{2k+1} \max_K \frac{1}{\tau} \left(\|\mathbf{u}^e\|_{k+1,\Omega_h} + \max_K \tau_K^* \|p^e\|_{k+1,\Omega_h} \right)^2. \quad (2.44)$$

The assertion is now ready by combining the inequalities (2.42)—(2.44), the definition of β , and the Cauchy-Schwarz inequality. \square

Remark 1. When the system is degenerate, but the exact solution is piecewise smooth, the convergence rate is sub-optimal by half order. The above proof, especially inequality (2.44), shows that this suboptimality may not be improved by using $\tau = \mathcal{O}(h^{-1})$.

The reason is that the gain by half order from $\max_K \frac{1}{\tau}$ is taken away by the loss of half order from $\max_K \tau_K^*$. This will be confirmed in our numerical studies of the sensitivity of τ on the convergence rate in Section 2.4.4.

2.4 Numerical results

In this section, we present numerical examples to support the HDG approach and its convergence analysis. For a non-degenerate case, we consider a sine solution test, while for degenerate cases, we choose smooth and non-smooth solution tests [7]. We take the upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ for a non-degenerate case, and the generalized parameter

$$\tau = \begin{cases} \phi^{-\frac{1}{2}}d & \text{for } \phi > 0, \\ 1/h & \text{for } \phi = 0 \end{cases}$$

for degenerate cases. We also conduct several numerical computations to understand if the stabilization parameter τ can affect the accuracy of the HDG solution and its convergence rate. We assume that porosity ϕ is known and $d = \phi$ in all the numerical examples. The domain Ω is chosen as $\Omega = (0, 1)^{dim}$ or $\Omega = (-1, 1)^{dim}$, which is either uniformly discretized with n_e rectangular tensor product elements in each dimension (so that the total number of elements is $N_e = n_e^{dim}$), or N_e triangular elements. Though we have rigorous optimal convergence theory for only simplicial meshes (see Theorem 1), a similar result is expected for quadrilateral/hexahedral meshes (see the numerical results in the following sections). Since rectangular meshes are convenient for all problems in this paper with simple interfaces between the fluid melt and the solid matrix, we use rectangular meshes hereafter, except for the test in Section 2.4.1.

2.4.1 Non-degenerate case

We consider a non-degenerate case on $\Omega = (0, 1)^2$ with the porosity given by $\phi = \exp(2(x + y))$.

We choose the pressure to be $p^e = \exp(-(x + y)) \sin(m_x \pi x) \sin(m_y \pi y)$.

The corresponding manufactured scaled solutions are given as

$$p^e = \sin(m_x \pi x) \sin(m_y \pi y), \quad (2.45a)$$

$$u_x^e = \exp(x + y) \sin(m_y \pi y) (\sin(m_x \pi x) - m_x \pi \cos(m_x \pi x)), \quad (2.45b)$$

$$u_y^e = \exp(x + y) \sin(m_x \pi x) (\sin(m_y \pi y) - m_y \pi \cos(m_y \pi y)). \quad (2.45c)$$

Here, we take $m_x = 2$ and $m_y = 3$.

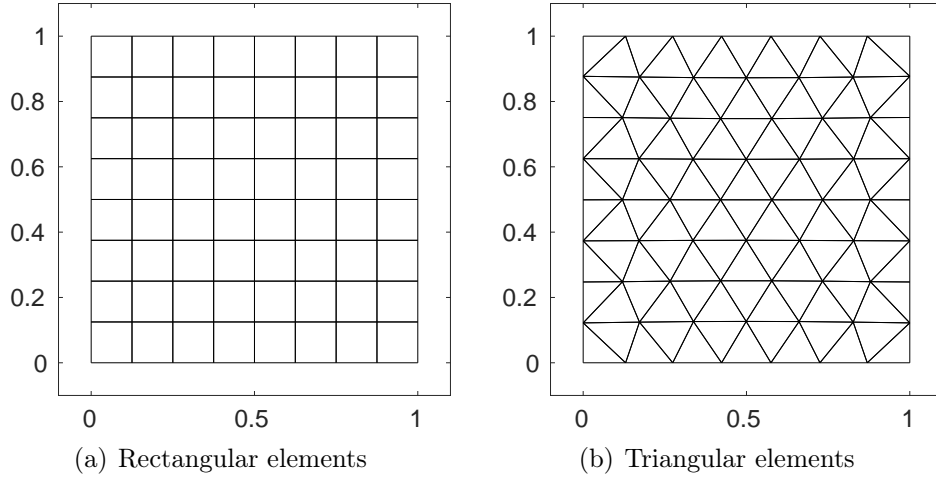


Figure 2.1: Coarse grids for non-degenerate case with (a) rectangular and (b) triangular elements.

Table 2.1 shows h -convergence results in the $L^2(\Omega_h)$ -norm using a sequence of nested meshes with $N_e = \{8^2, 32^2, 128^2\}$ for rectangular. The corresponding coarse meshes are shown in Figure 2.1. We observe approximately the optimal convergence rates of $(k + 1)$ for both scaled pressure p and scaled velocity \mathbf{u} for both mesh types.

Table 2.1: Non-degenerate case: the results show that the HDG solutions for scaled pressure p and scaled velocity \mathbf{u} converge to the exact solution with optimal order of $k + 1$ for both triangular and rectangular meshes. The upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ is used.

k	h	Rectangular elements				Triangular elements				
		$\ p^e - p\ _{\Omega_h}$ error	order	$\ \mathbf{u}^e - \mathbf{u}\ _{\Omega_h}$ error	order	h	$\ p^e - p\ _{\Omega_h}$ error	order	$\ \mathbf{u}^e - \mathbf{u}\ _{\Omega_h}$ error	order
1	0.031	3.6E-02	—	8.5E-01	1.4	0.140	6.5E-01	—	5.0E+00	—
	0.016	1.2E-02	1.6	2.9E-01	1.6	0.070	1.8E-01	1.8	1.4E+00	1.8
	0.008	3.4E-03	1.8	8.8E-02	1.7	0.035	4.7E-02	1.9	3.8E-01	1.9
2	0.031	1.1E-03	—	2.7E-02	—	0.140	9.4E-02	—	8.3E-01	—
	0.016	1.6E-04	2.7	4.3E-03	2.7	0.070	1.2E-02	2.9	1.1E-01	3.0
	0.008	2.2E-05	2.8	6.2E-04	2.8	0.035	1.6E-03	3.0	1.4E-02	3.0
3	0.031	2.0E-05	—	4.7E-04	—	0.140	8.9E-03	—	6.7E-02	—
	0.016	1.4E-06	3.8	3.5E-05	3.8	0.070	5.9E-04	3.9	4.6E-03	3.9
	0.008	9.5E-08	3.9	2.4E-06	3.8	0.035	3.7E-05	4.0	2.9E-04	4.0
4	0.031	3.3E-07	—	8.8E-06	—	0.140	8.0E-04	—	7.3E-03	—
	0.016	1.2E-08	4.8	3.2E-07	4.8	0.070	2.6E-05	5.0	2.3E-04	5.0
	0.008	3.9E-10	4.9	1.1E-08	4.8	0.035	8.1E-07	5.0	7.2E-06	5.0

2.4.2 Degenerate case with a smooth solution

Following [7] we consider the smooth pressure of the form $p^e = \cos(6xy^2)$ on $\Omega = (-1, 1)^2$ and the following degenerate porosity

$$\phi = \begin{cases} 0, & x \leq -\frac{3}{4} \text{ or } y \leq -\frac{3}{4}, \\ (x + \frac{3}{4})^\alpha (y + \frac{3}{4})^{2\alpha}, & \text{otherwise.} \end{cases} \quad (2.46)$$

We note that $\phi^{-\frac{1}{2}}\nabla\phi \in [L^\infty(\Omega)]^2$ for $\alpha \geq 2$, and we take $\alpha = 2$. The one-phase region is denoted as $\Omega_1 := \{(x, y) : x \leq -\frac{3}{4} \text{ or } y \leq -\frac{3}{4}\}$ with $\phi = 0$, and the two-phase region is given by $\Omega_2 := \{(x, y) : -\frac{3}{4} < x < 1 \text{ and } -\frac{3}{4} < y < 1\}$ with $\phi > 0$. We define the intersection of these two regions by $\partial\Omega_{12} := \overline{\Omega}_1 \cap \overline{\Omega}_2$. In $\overline{\Omega}_1$, the exact

scaled pressure and scaled velocity vanish. In $\bar{\Omega}_2$, the exact solutions are given by

$$p^e = \left(x + \frac{3}{4}\right)^{\frac{\alpha}{2}} \left(y + \frac{3}{4}\right)^{\alpha} \cos(6xy^2), \quad (2.47a)$$

$$u_x^e = 6y^2 \left(x + \frac{3}{4}\right)^{\alpha} \left(y + \frac{3}{4}\right)^{2\alpha} \sin(6xy^2), \quad (2.47b)$$

$$u_y^e = 12xy \left(x + \frac{3}{4}\right)^{\alpha} \left(y + \frac{3}{4}\right)^{2\alpha} \sin(6xy^2). \quad (2.47c)$$

In Figure 2.2 are the contours of the pressure p and the scaled pressure p computed from our HDG method using $N_e = 64^2$ rectangular elements and solution order $k = 4$. We observe that the pressure p changes smoothly in the two-phase region Ω_2 , but abruptly becomes zero in the one-phase region Ω_1 . The sudden pressure jump on the intersection Ω_{12} is alleviated with the use of the scaled pressure p .

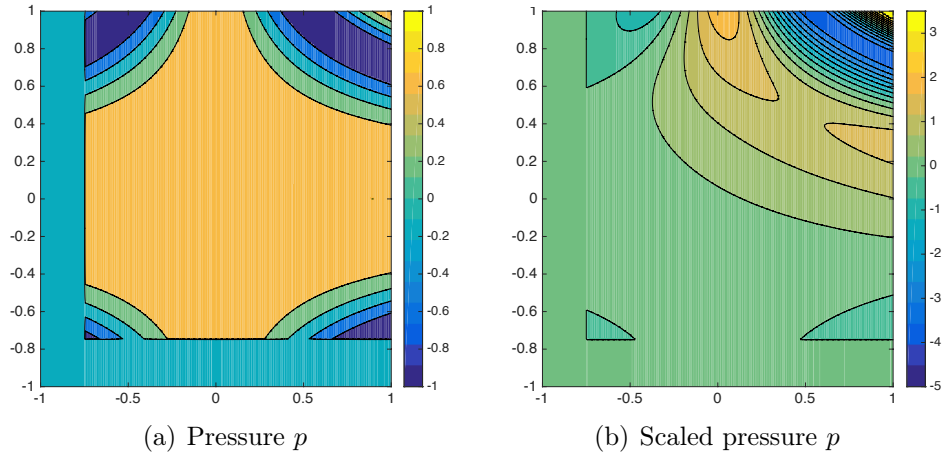


Figure 2.2: Degenerate case with a smooth solution: (a) contour plot of the pressure p field and (b) contour plot of the scaled pressure p with $N_e = 64^2$ and $k = 4$. The pressure field changes smoothly in the two-phase region Ω_2 , but suddenly becomes zero in the one-phase region Ω_1 . The abrupt change near the intersection Ω_{12} between the one- and two-phase regions is alleviated with the use of the scaled pressure p .

For a convergence study, we use a sequence of meshes with $n_e = \{16, 32, 64, 128\}$ and with $k = \{1, 2, 3, 4\}$. Here we choose an even number of elements so that the mesh skeleton aligns with the intersection $\partial\Omega_{12}$. As can be seen in Figure 2.3, the

convergence rate of $(k + \frac{1}{2})$ is observed more or less for both the scaled pressure p and the scaled velocity \mathbf{u} , and this agrees with Theorem 1 for the degenerate case with a piecewise smooth solution.

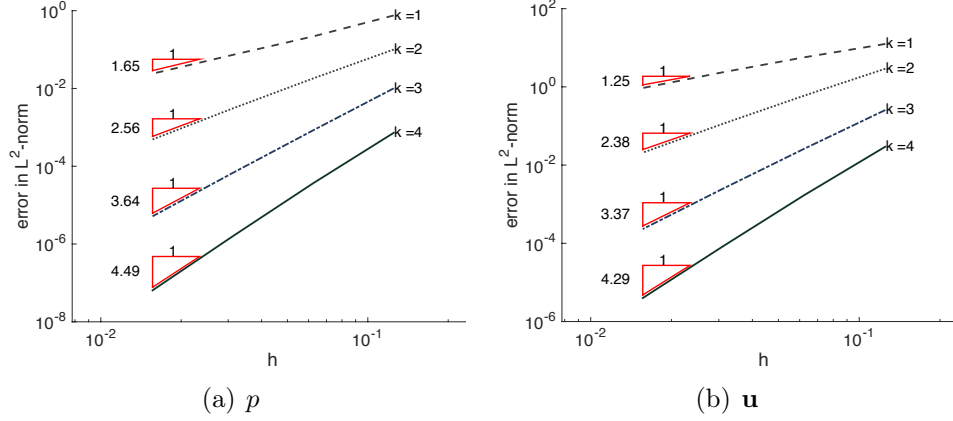


Figure 2.3: Degenerate case with a smooth solution: convergence study for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field. The $(k + \frac{1}{2})$ convergence rates are obtained approximately for both the scaled pressure p and the scaled velocity \mathbf{u} .

2.4.3 Degenerate case with low solution regularity

Similar to [7], we choose the exact pressure to be $p^e = y(y - 3x)(x + \frac{3}{4})^\beta$ with $\beta = -\frac{1}{4}$ or $-\frac{3}{4}$, and the porosity ϕ is defined in (2.46). Similar to Section 2.4.2, we take $\alpha = 2$. The exact solutions then read

$$p^e = y(y - 3x) \left(x + \frac{3}{4}\right)^{\frac{\alpha}{2} + \beta} \left(y + \frac{3}{4}\right)^\alpha, \quad (2.48a)$$

$$u_x^e = y \left(\beta(3x - y) + 3 \left(x + \frac{3}{4}\right) \right) \left(x + \frac{3}{4}\right)^{\alpha + \beta - 1} \left(y + \frac{3}{4}\right)^{2\alpha}, \quad (2.48b)$$

$$u_y^e = (3x - 2y) \left(x + \frac{3}{4}\right)^{\alpha + \beta} \left(y + \frac{3}{4}\right)^{2\alpha}. \quad (2.48c)$$

The pressure and the scaled pressure fields are simulated with $N_e = 64^2$ and $k = 4$ for the two different cases: $\beta = -\frac{1}{4}$ and $\beta = -\frac{3}{4}$ in Figure 2.4. As can be seen from (2.48)

and Figure 2.4 that smaller β implies lower solution regularity. The pressure field with $\beta = -\frac{3}{4}$ is less regular than that with $\beta = -\frac{1}{4}$. For both cases, we also observe that the pressure p fields become stiffer (stiff “boundary layer”) near the intersection at $x = -\frac{3}{4}$, while the scaled pressure p fields are much less stiff.

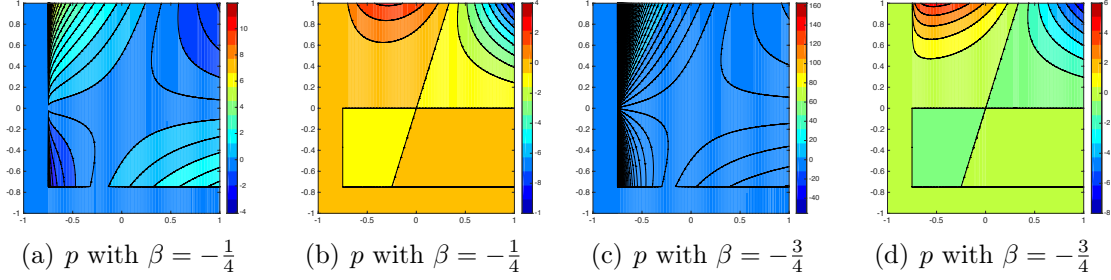


Figure 2.4: Degenerate case with low solution regularity: simulated with $N_e = 64^2$ and $k = 4$ are (a) pressure p for $\beta = -\frac{1}{4}$, (b) scaled pressure p for $\beta = -\frac{1}{4}$, (c) pressure p with $\beta = -\frac{3}{4}$, and (d) scaled pressure p for $\beta = -\frac{3}{4}$. The pressure field with $\beta = -\frac{3}{4}$ is less regular than that with $\beta = -\frac{1}{4}$. In both the cases, the pressure fields have low regularity near the intersection Ω_{12} .

When $\beta = -\frac{1}{4}$, the scaled pressure p and the scaled velocity \mathbf{u} reside in $H^{1.25-\varepsilon}$ for $\varepsilon > 0$ [7]. In order to see how the HDG solution behaves for this case, we perform a convergence study with $n_e = \{16, 32, 64, 128\}$ and $k = \{1, 2, 4, 8\}$. As shown in Figure 2.5, the scaled pressure p and the scaled velocity \mathbf{u} converge to the exact counterparts with the rate of about 1.25. Note that though our error analysis in Section 2.3.4 considers exact solutions residing in standard Sobolev spaces with integer powers, it can be straightforwardly extended to solutions in fractional Sobolev spaces. For this example, the convergence rate is bounded above by $1.25 - \varepsilon$ regardless of the solution order. However, the high order HDG solutions are still beneficial in terms of accuracy, for example, the HDG solution with $k = 8$ is 4.5 times more accurate than that with $k = 4$.

When $\beta = -\frac{3}{4}$, the scaled pressure p and the scaled velocity \mathbf{u} lie in $H^{0.75-\varepsilon}$ for $\varepsilon > 0$ [7]. We conduct a convergence study with $n_e = \{16, 32, 64, 128\}$ and

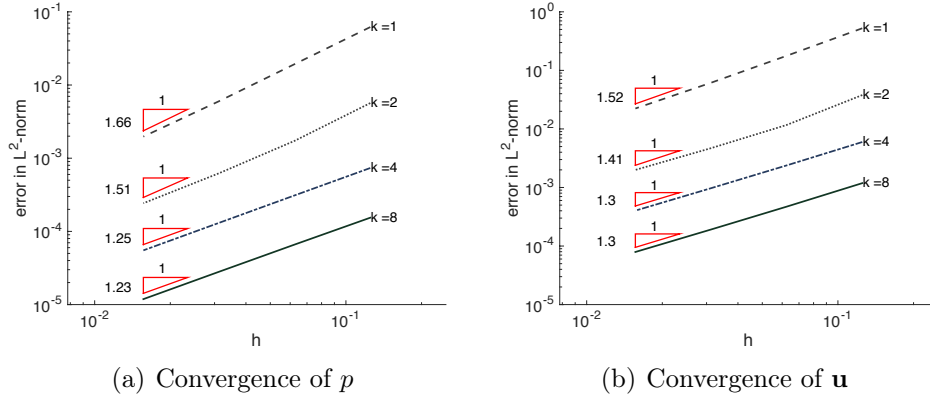


Figure 2.5: Degenerate case with low solution regularity: a convergence study with $\beta = -\frac{1}{4}$ for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field.

$k = \{1, 2, 4, 8\}$. As shown in Figure 2.6, the convergence rate of about 0.75 is observed for both the scaled pressure p and the scaled velocity \mathbf{u} .

Similar to the case of $\beta = -\frac{1}{4}$, high order HDG solutions, in spite of more computational demand, are beneficial from an accuracy standpoint. For instance, the HDG solution with $k = 8$ is 2.5 times more accurate than that with $k = 4$.

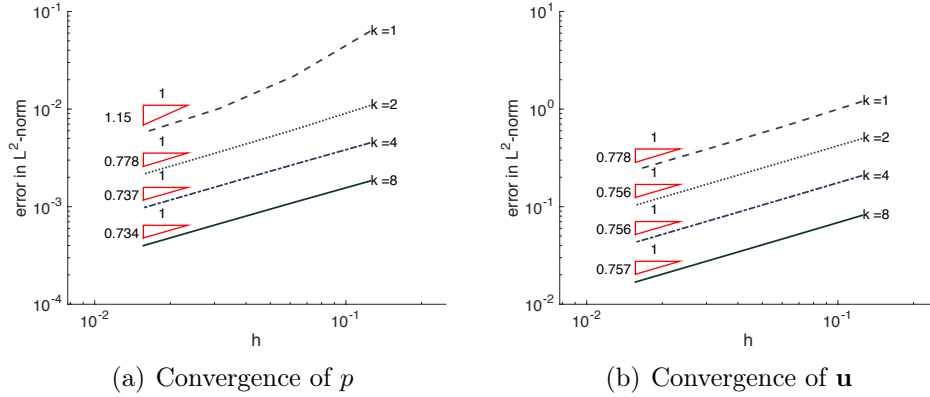


Figure 2.6: Degenerate case with low solution regularity: a convergence study with $\beta = -\frac{3}{4}$ for (a) the scaled pressure p field and (b) the scaled velocity \mathbf{u} field.

2.4.4 Sensitivity of τ for the degenerate case with smooth solution

In this section we assess numerically whether the sub-optimality in Theorem 1 is sharp. To that end, we consider the degenerate example with smooth solution in Section 2.4.2 again here. Recall that the generalized parameter τ is given by

$$\tau := \begin{cases} \phi^{-\frac{1}{2}}d & \text{for } \phi > 0, \\ \gamma & \text{for } \phi = 0. \end{cases} \quad (2.49)$$

We first compare the convergence rates for three different values of γ , namely $\gamma \in \{1/h, 1, 10\}$, and the numerical results (not shown here) show that the convergence rates are the same and are sub-optimal by half order. For that reason we show only the case when $\gamma = 1/h$ in the fourth column of Tables 2.2 and 2.3, in which we report the convergence rates of p and \mathbf{u} , respectively.

We now present convergence rates for the cases where we use a single value for τ over the entire mesh skeleton \mathcal{E}_h . We consider three cases: $\tau = (1/h, 1, 10)$. The convergence rates of p and \mathbf{u} for these parameters are shown in the sixth, eighth, and tenth columns of Tables 2.2 and 2.3. The results for $\tau = 1$ and $\tau = 10$ show the convergence rate of about $(k + \frac{1}{2})$. The cases with $\tau = \frac{1}{h}$ initially have the convergence rate of $(k + 1)$ for both the scaled pressure p and the scaled velocity \mathbf{u} , then approach the predicted asymptotic rate of $(k + \frac{1}{2})$ as the grid is refined. If we look at the value of the errors at any grid level, the cases with $\tau = 1/h$ have the smallest errors compared to the other cases (including the cases with τ given in (2.49)). It could be due to the initial higher-order convergence and/or smaller error constants. We thus recommend that $\tau = 1/h$ should be used.

2.4.5 Enhance accuracy by post-processing

In this section, we explore the superconvergence property for the degenerate elliptic equations. It is well-known that the HDG methods have a superconvergence

Table 2.2: Degenerate case with a smooth solution: the errors $\|p^e - p\|_{\Omega_h}$ and the convergence rates for the scaled pressure. Four cases are presented: τ given (2.49), $\tau = \frac{1}{h}$, $\tau = 1$ and $\tau = 10$.

k	h	$\tau = \begin{cases} \phi^{-\frac{1}{2}d} & \text{for } \phi > 0 \\ 1/h & \text{for } \phi = 0 \end{cases}$		$\tau = \frac{1}{h}$		$\tau = 1$		$\tau = 10$	
		error	order	error	order	error	order	error	order
1	0.125	7.53E-01	—	1.75E-01	—	2.61E+00	—	3.83E-01	—
	0.063	2.19E-01	1.78	4.11E-02	2.09	4.81E-01	2.44	1.23E-01	1.64
	0.031	7.32E-02	1.58	9.14E-03	2.17	1.42E-01	1.76	4.10E-02	1.58
	0.016	2.40E-02	1.61	2.31E-03	1.99	5.00E-02	1.51	1.31E-02	1.64
2	0.125	1.00E-01	—	3.31E-02	—	1.64E-01	—	6.44E-02	—
	0.063	1.82E-02	2.47	3.32E-03	3.32	3.28E-02	2.32	1.12E-02	2.53
	0.031	3.08E-03	2.56	3.89E-04	3.09	6.32E-03	2.38	1.79E-03	2.64
	0.016	4.91E-04	2.65	6.16E-05	2.66	1.15E-03	2.45	2.71E-04	2.72
3	0.125	1.02E-02	—	2.82E-03	—	1.82E-02	—	5.78E-03	—
	0.063	8.53E-04	3.57	1.65E-04	4.09	1.71E-03	3.42	4.68E-04	3.63
	0.031	6.86E-05	3.64	1.10E-05	3.91	1.54E-04	3.47	3.63E-05	3.69
	0.016	5.24E-06	3.71	8.46E-07	3.70	1.32E-05	3.55	2.69E-06	3.75
4	0.125	7.24E-04	—	2.45E-04	—	1.18E-03	—	4.61E-04	—
	0.063	3.70E-05	4.29	7.20E-06	5.09	6.74E-05	4.13	2.30E-05	4.33
	0.031	1.62E-06	4.52	2.17E-07	5.05	3.29E-06	4.36	9.60E-07	4.58
	0.016	6.56E-08	4.62	8.53E-09	4.67	1.51E-07	4.45	3.73E-08	4.69

property [51], i.e., the post-processed solution p^* converges faster than p . For the standard elliptic system, a post-processed solution p^* has $(k + 2)$ convergence rate. For degenerate equations, the convergence rate, as shown in the numerical results, is dictated by the regularity of the degenerate solutions. For that reason, it is not meaningful to post-process the solution over the entire domain Ω . However, over subdomains with positive porosity, the solution has higher regularity and thus the post-processed counterpart is expected to exhibit super-convergence. To this end, we seek a new approximation $p^* \in \mathcal{P}^{k+1}(K)$ by minimizing $\left\| \mathbf{u} - \frac{1}{2}\phi^{-\frac{3}{2}}d\nabla\phi p + \phi^{-\frac{1}{2}}d\nabla p^* \right\|^2$

Table 2.3: Degenerate case with a smooth solution: the errors $\|\mathbf{u}^e - \mathbf{u}\|_{\Omega_h}$ and the convergence rates for the scaled velocity. Four cases are presented: τ given (2.49), $\tau = \frac{1}{h}$, $\tau = 1$ and $\tau = 10$.

k	h	$\tau = \begin{cases} \phi^{-\frac{1}{2}d} & \text{for } \phi > 0 \\ 1/h & \text{for } \phi = 0 \end{cases}$		$\tau = \frac{1}{h}$		$\tau = 1$		$\tau = 10$	
		error	order	error	order	error	order	error	order
1	0.125	1.25E+01	—	7.42E+00	—	1.52E+01	—	1.03E+01	—
	0.063	5.71E+00	1.13	2.23E+00	1.73	7.36E+00	1.04	4.48E+00	1.20
	0.031	2.39E+00	1.26	5.66E-01	1.98	3.33E+00	1.14	1.78E+00	1.33
	0.016	9.37E-01	1.35	1.51E-01	1.91	1.45E+00	1.20	6.66E-01	1.42
2	0.125	2.91E+00	—	1.66E+00	—	3.65E+00	—	2.36E+00	—
	0.063	6.00E-01	2.28	2.37E-01	2.80	8.36E-01	2.13	4.57E-01	2.37
	0.031	1.15E-01	2.38	3.58E-02	2.73	1.82E-01	2.20	8.25E-02	2.47
	0.016	2.08E-02	2.47	5.95E-03	2.59	3.77E-02	2.27	1.41E-02	2.55
3	0.125	2.55E-01	—	1.24E-01	—	3.60E-01	—	1.88E-01	—
	0.063	2.64E-02	3.28	9.49E-03	3.71	4.00E-02	3.17	1.89E-02	3.32
	0.031	2.54E-03	3.37	6.95E-04	3.77	4.23E-03	3.24	1.76E-03	3.42
	0.016	2.32E-04	3.46	5.44E-05	3.68	4.24E-04	3.32	1.57E-04	3.49
4	0.125	2.95E-02	—	1.71E-02	—	3.615E-02	—	2.44E-02	—
	0.063	1.72E-03	4.10	6.53E-04	4.71	2.348E-03	3.944	1.34E-03	4.18
	0.031	8.59E-05	4.32	2.27E-05	4.84	1.321E-04	4.152	6.36E-05	4.40
	0.016	3.99E-06	4.43	8.74E-07	4.70	6.999E-06	4.238	2.81E-06	4.50

over an element K , which leads to the following local equations:

$$(\nabla p^*, \nabla \omega)_K = - \left(\phi^{\frac{1}{2}} d^{-1} \mathbf{u}, \nabla \omega \right)_K + \frac{1}{2} (\phi^{-1} \nabla \phi p, \nabla \omega)_K, \quad (2.50a)$$

$$(p^*, 1)_K = (p, 1)_K, \quad (2.50b)$$

for all $\omega \in \mathcal{P}^{k+1}(K)$. Since the resulting linear system has $k + 2$ equations and any one of (2.50a) is a linear combination of the others, we remove one of the rows of the linear system of (2.50a) in order to obtain a unique solution. Similarly, we utilize the post-processing technique for the unscaled fluid pressure p . We seek a new

approximation $\tilde{p}^* \in \mathcal{P}^{k+1}(K)$ by minimizing $\|\mathbf{u} - d\nabla\tilde{p}^*\|^2$ over an element K , which leads to the following local equations:

$$(\nabla\tilde{p}^*, \nabla\omega)_K = - (d^{-1}\mathbf{u}, \nabla\omega)_K, \quad (2.51a)$$

$$(\tilde{p}^*, 1)_K = (p, 1)_K, \quad (2.51b)$$

for all $\omega \in \mathcal{P}^{k+1}(K)$.

Table 2.4 shows h -convergence results for the non-degenerate case in Section 2.4.1 with $(m_x, m_y) = (1, 2)$ using a series of nested meshes, $N_e = \{8^2, 16^2, 32^2\}$, for rectangular elements. We observe that both the post-processed scaled pressure p^* and the post-processed fluid pressure \tilde{p}^* converge faster than the scaled pressure p and the fluid pressure p . The convergence rates of both the p^* and \tilde{p}^* are approximately $(k + 2)$, except when $k = 1$.

Next, we examine if the post-processing technique can be utilized in a degenerate case. To address the question, we first define a subdomain $\tilde{\Omega} \subset \Omega$, which is a two-phase region and is “far” enough from the degenerate regions so that the solutions are less affected by the degeneracy. We perform several convergence studies for the degenerate cases in Section 2.4.2 and Section 2.4.3 using a sequence of nested meshes, $N_e = \{16^2, 32^2, 64^2\}$, for rectangular elements.

Table 2.5 shows h -convergence results for the degenerate case with the smooth solution in Section 2.4.2 over $\tilde{\Omega} \in [-0.5, 1]^2$. In general, both the post-processed solutions p^* and \tilde{p}^* converge faster than the solutions p and p . When $k > 1$, the convergence rates for the post-processed solutions p^* and \tilde{p}^* are $(k + \frac{3}{2})$, one order faster than p and p .

For the degenerate case with low solution regularity in Section 2.4.3, the HDG solution already attains the maximal convergence order. It is thus not meaningful

to post-process the solution over the whole domain as there will be no gain in the convergence rate. Instead, we conduct post-processing studies over $\tilde{\Omega} \in [-0.5, 1]^2$ and $\tilde{\Omega} \in [0, 1]^2$ where the solution is expected to be more regular, and hence allowing the post-processing procedure to achieve a better convergence rate.

In Table 2.6 with $\beta = -\frac{3}{4}$, the convergence rate for the scaled pressure p is approximately $(k + 1)$ when $k \leq 2$, but it starts to degrade to 2.3 as the order k increases. This numerically implies that the exact solution p^e over $\tilde{\Omega}$ resides in $H^{2.5-\varepsilon}(\tilde{\Omega})$ (i.e., it has a higher regularity than over the whole domain). When the convergence rate of p reaches the maximum possible, there is no improvement for the post-processed counterpart.

Another observation is that the post-processed solution over the sub-domain still provides a benefit in terms of accuracy to a certain extent. In Table 2.6 and 2.7, the post-processed scaled pressure p^* has a smaller error than the scaled pressure p . However, the difference between p^* and p becomes negligible as the solution order increases. A similar behavior is also seen for the post-processed fluid pressure \tilde{p}^* and the fluid pressure p .

2.4.6 Non-degenerate case in three dimensions

We consider finally a non-degenerate case on $\Omega = (0, 1)^3$ with the positive porosity $\phi = \exp(2(x + y + z))$. Let the pressure

$$p^e = \sin(m_x \pi x) \sin(m_y \pi y) \sin(m_z \pi z) \exp(-(x + y + z)).$$

Table 2.4: Non-degenerate case: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The post-processed solutions show asymptotically $(k+2)$ convergence rates for $k > 1$.

k	h	$\ p^e - p\ _2$		$\ p^e - p^*\ _2$		$\ \tilde{p}^e - \tilde{p}\ _2$		$\ \tilde{p}^e - \tilde{p}^*\ _2$	
		error	order	error	order	error	order	error	order
1	0.1250	1.508E-01	—	5.852E-02	—	6.476E-02	—	2.822E-02	—
	0.0625	5.014E-02	1.589	1.245E-02	2.233	2.005E-02	1.692	5.928E-03	2.251
	0.0312	1.497E-02	1.743	2.647E-03	2.234	5.706E-03	1.813	1.253E-03	2.242
2	0.1250	1.337E-02	—	5.001E-04	—	4.612E-03	—	1.970E-04	—
	0.0625	2.053E-03	2.703	3.386E-05	3.884	6.805E-04	2.761	1.174E-05	4.068
	0.0312	2.912E-04	2.818	2.275E-06	3.896	9.361E-05	2.862	7.182E-07	4.031
3	0.1250	6.595E-04	—	1.263E-05	—	2.608E-04	—	3.761E-06	—
	0.0625	4.815E-05	3.776	4.484E-07	4.816	1.819E-05	3.842	1.232E-07	4.932
	0.0312	3.289E-06	3.872	1.523E-08	4.880	1.209E-06	3.911	3.987E-09	4.950
4	0.1250	3.109E-05	—	4.289E-07	—	1.083E-05	—	1.194E-07	—
	0.0625	1.113E-06	4.804	7.568E-09	5.825	3.731E-07	4.859	1.940E-09	5.944
	0.0312	3.762E-08	4.886	1.276E-10	5.890	1.231E-08	4.922	3.110E-11	5.963

The corresponding manufactured scaled solutions are given as

$$q^e = \sin(m_x \pi x) \sin(m_y \pi y) \sin(m_z \pi z), \quad (2.52a)$$

$$u_x^e = \exp(x + y + z) \sin(m_y \pi y) \sin(m_z \pi z) (\sin(m_x \pi x) - m_x \pi \cos(m_x \pi x)), \quad (2.52b)$$

$$u_y^e = \exp(x + y + z) \sin(m_x \pi x) \sin(m_z \pi z) (\sin(m_y \pi y) - m_y \pi \cos(m_y \pi y)), \quad (2.52c)$$

$$u_z^e = \exp(x + y + z) \sin(m_x \pi x) \sin(m_y \pi y) (\sin(m_z \pi z) - m_z \pi \cos(m_z \pi z)). \quad (2.52d)$$

Table 2.8 shows h -convergence results in the $L^2(\Omega_h)$ -norm using a sequence of nested meshes with $n_e = \{8, 12, 16, 20\}$. Here we take $m_x = m_y = m_z = 1$ and use the upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$. We observe the convergence rates between $(k + \frac{1}{2})$ and $(k + 1)$ for both scaled pressure p and scaled velocity \mathbf{u} . Recall that the

Table 2.5: Degenerate case with a smooth solution: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [-0.5, 1]^2$. The post-processed solutions show asymptotically $(k + \frac{3}{2})$ convergence rates for $k > 1$.

k	h	$\ p^e - p\ _2$		$\ p^e - p^*\ _2$		$\ \tilde{p}^e - \tilde{p}\ _2$		$\ \tilde{p}^e - \tilde{p}^*\ _2$	
		error	order	error	order	error	order	error	order
1	0.0625	7.659E-01	—	2.301E-01	—	2.264E-01	—	9.357E-02	—
	0.0312	2.211E-01	1.792	5.174E-02	2.153	6.588E-02	1.781	2.052E-02	2.189
	0.0156	7.361E-02	1.587	1.135E-02	2.188	2.085E-02	1.660	4.512E-03	2.185
2	0.0625	1.005E-01	—	6.934E-03	—	2.742E-02	—	2.829E-03	—
	0.0312	1.818E-02	2.467	5.790E-04	3.582	4.617E-03	2.570	2.049E-04	3.788
	0.0156	3.077E-03	2.563	4.939E-05	3.551	7.517E-04	2.619	1.522E-05	3.750
3	0.0625	1.007E-02	—	2.697E-04	—	2.689E-03	—	9.138E-05	—
	0.0312	8.422E-04	3.579	1.620E-05	4.512	2.174E-04	3.629	3.697E-06	4.627
	0.0156	6.758E-05	3.639	7.168E-07	4.498	1.703E-05	3.674	1.597E-07	4.533
4	0.0625	7.239E-04	—	2.446E-05	—	1.921E-04	—	5.478E-06	—
	0.0312	3.672E-05	4.301	6.056E-07	5.336	8.891E-06	4.434	1.279E-07	5.421
	0.0156	1.592E-06	4.528	1.400E-08	5.435	3.704E-07	4.585	2.928E-09	5.449

optimal convergence rate of $k + 1$ is proved for only simplices, though similar results for quadrilaterals and hexahedra are expected. Indeed, Table 2.8 shows that as the solution order increases, the convergence rate is above $k + \frac{1}{2}$.

2.5 Discussions

In this chapter, we developed numerical methods for both glacier dynamics and mantle convection. Both phenomena can be described by a two-phase mixture model, in which the mixture of the fluid and the solid is described by the porosity ϕ (i.e., $\phi > 0$ implies the fluid-solid two-phase and $\phi = 0$ means the solid single-phase

Table 2.6: Degenerate case with low solution regularity for $\beta = -\frac{3}{4}$: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure p , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [-0.5, 1]^2$.

k	h	$\ p^e - p\ _2$		$\ p^e - p^*\ _2$		$\ \tilde{p}^e - \tilde{p}\ _2$		$\ \tilde{p}^e - \tilde{p}^*\ _2$	
		error	order	error	order	error	order	error	order
1	0.0625	5.982E-02	—	1.588E-02	—	6.842E-02	—	4.649E-02	—
	0.0312	1.711E-02	1.806	3.798E-03	2.064	1.808E-02	1.920	1.168E-02	1.992
	0.0156	4.627E-03	1.887	9.559E-04	1.990	4.674E-03	1.952	2.938E-03	1.992
2	0.0625	2.679E-03	—	1.641E-04	—	5.039E-03	—	1.644E-03	—
	0.0312	3.654E-04	2.874	1.971E-05	3.058	5.641E-04	3.159	1.068E-04	3.944
	0.0156	4.821E-05	2.922	3.479E-06	2.502	6.806E-05	3.051	8.619E-06	3.632
3	0.0625	7.697E-05	—	1.337E-05	—	1.761E-04	—	4.230E-05	—
	0.0312	5.928E-06	3.699	3.020E-06	2.146	1.207E-05	3.867	5.301E-06	2.996
	0.0156	6.829E-07	3.118	5.954E-07	2.343	1.253E-06	3.267	1.027E-06	2.367
4	0.0625	4.568E-06	—	4.249E-06	—	9.124E-06	—	7.593E-06	—
	0.0312	8.400E-07	2.443	8.381E-07	2.342	1.470E-06	2.634	1.458E-06	2.381
	0.0156	1.689E-07	2.314	1.689E-07	2.311	2.920E-07	2.331	2.920E-07	2.320

region). The challenge is when the porosity vanishes because the system degenerates, which make the problem difficult to solve numerically. To address the issue, following [7], we start by scaling variables to obtain the well-posedness. Then we spatially discretize the system using the upwind HDG framework. The key feature is that we have modified the upwind HDG flux to accommodate the degenerate (one-phase) region. When the porosity vanishes, the unmodified HDG system becomes ill-posed because the stabilization parameter associated with the HDG flux disappears. For this reason, we introduce the generalized stabilization parameter that is composed of the upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ in the two-phase region and a positive parameter $\tau = \frac{1}{h} > 0$ in the one-phase region. This enabled us to develop a high-

Table 2.7: Degenerate case with low solution regularity for $\beta = -\frac{3}{4}$: the errors and the convergence rates for the scaled pressure p , the post-processed scaled pressure p^* , the fluid pressure \tilde{p} , and the post-processed fluid pressure \tilde{p}^* . The errors are computed over the subdomain $\tilde{\Omega} \in [0, 1]^2$

k	h	$\ p^e - p\ _2$		$\ p^e - p^*\ _2$		$\ \tilde{p}^e - \tilde{p}\ _2$		$\ \tilde{p}^e - \tilde{p}^*\ _2$	
		error	order	error	order	error	order	error	order
1	0.0625	4.630E-02	—	1.061E-02	—	2.127E-02	—	7.002E-03	—
	0.0312	1.352E-02	1.776	2.237E-03	2.246	6.038E-03	1.817	1.413E-03	2.309
	0.0156	3.710E-03	1.865	5.230E-04	2.097	1.640E-03	1.881	3.148E-04	2.166
2	0.0625	2.210E-03	—	9.474E-05	—	1.034E-03	—	4.804E-05	—
	0.0312	3.055E-04	2.854	7.706E-06	3.620	1.388E-04	2.898	3.983E-06	3.592
	0.0156	4.061E-05	2.912	6.422E-07	3.585	1.817E-05	2.933	3.595E-07	3.470
3	0.0625	6.260E-05	—	1.584E-06	—	3.369E-05	—	8.669E-07	—
	0.0312	4.235E-06	3.886	1.202E-07	3.721	2.176E-06	3.952	1.028E-07	3.076
	0.0156	2.791E-07	3.924	2.114E-08	2.507	1.409E-07	3.949	1.987E-08	2.371
4	0.0625	1.172E-06	—	1.523E-07	—	8.112E-07	—	1.426E-07	—
	0.0312	4.882E-08	4.586	2.980E-08	2.354	3.792E-08	4.419	2.820E-08	2.338
	0.0156	6.110E-09	2.998	5.978E-09	2.317	5.703E-09	2.733	5.646E-09	2.321

order HDG method for a linear degenerate elliptic equation arising from a two-phase mixture of both glacier dynamics and mantle convection.

We have shown the well-posedness and the convergence analysis of our HDG scheme. The rigorous theoretical results tell us that our HDG method has the convergence rates of $(k + 1)$ for a non-degenerate case and $(k + \frac{1}{2})$ for a degenerate case with a piecewise smooth solution.

Several numerical results confirm that our proposed HDG method works well for linear degenerate elliptic equations. For the non-degenerate case, we obtain the $(k + 1)$ convergence rates of both the scaled pressure p and the scaled velocity \mathbf{u}

Table 2.8: Non-degenerate case in three dimensions: the results show that the HDG solutions for scaled pressure p and scaled velocity \mathbf{u} converge to the exact solutions with the rate in above $k + \frac{1}{2}$. The upwind based parameter $\tau = \phi^{-\frac{1}{2}}d$ is used.

k	h	$\ p^e - p\ _2$		$\ \mathbf{u}^e - \mathbf{u}\ _2$	
		error	order	error	order
1	0.1250	2.553E-02	—	5.895E-01	—
	0.0833	1.473E-02	1.356	3.527E-01	1.267
	0.0625	9.754E-03	1.433	2.399E-01	1.340
	0.0500	6.994E-03	1.491	1.757E-01	1.396
2	0.1250	1.405E-03	—	4.408E-02	—
	0.0833	5.274E-04	2.417	1.712E-02	2.333
	0.0625	2.576E-04	2.491	8.579E-03	2.402
	0.0500	1.460E-04	2.545	4.961E-03	2.455
3	0.1250	4.872E-05	—	1.477E-03	—
	0.0833	1.183E-05	3.491	3.684E-04	3.425
	0.0625	4.234E-06	3.572	1.348E-04	3.495
	0.0500	1.885E-06	3.626	6.105E-05	3.550
4	0.1250	1.100E-06	—	2.741E-05	—
	0.0833	1.705E-07	4.598	4.428E-06	4.496
	0.0625	4.447E-08	4.672	1.188E-06	4.573
	0.0500	1.551E-08	4.720	4.235E-07	4.622

in two dimensions, whereas in three dimensions we observe the convergence rates above $(k + \frac{1}{2})$. For the degenerate case with a smooth solution, the convergence rate of $(k + \frac{1}{2})$ is observed for both the scaled pressure p and the scaled velocity \mathbf{u} . For the degenerate case with low solution regularity, the convergence rates of the numerical solutions are bounded by the solution regularity, but the high-order method still shows a benefit in terms of accuracy. For smooth solutions for which the superconvergence property of HDG methods holds, we can enhance the HDG solutions by post-processing. We have shown that the post-processed HDG solution converges

to the exact solution faster than the HDG solution. The convergence rates for the post-processed solutions are approximately $(k + 2)$ for the non-degenerate case, and $(k + \frac{3}{2})$ for the degenerate case with a smooth solution. Through a parameter study, we found that using a positive parameter on the one-phase region does not affect the accuracy of a numerical solution. We also found that $\tau = 1/h$ showed slightly better performance in terms of error levels and convergence rates for the degenerate case with smooth solution.

In order for our proposed method to work in two-phase flows, the interfaces between matrix solid and fluid melt need to be identified and grids should be aligned with the interfaces. In other words, the degeneracies are always required to lie on a set of measure zero.

2.6 Auxiliary results

In this appendix we collect some technical results that are useful for our analysis.

Lemma 8 (Inverse Inequality [168, Lemma 1.44]). For $v \in \mathcal{P}_k(K)$ with $K \in \Omega_h$, there exists $c > 0$ independent of h such that

$$\|\nabla v\|_{0,K} \leq ch_K^{-1} \|v\|_{0,K}. \quad (2.53)$$

Lemma 9 (Trace inequality [168, Lemma 1.49]). For $v \in H^1(\Omega_h)$ and for $K \in \Omega_h$ with $e \subset \partial K$, there exists $c > 0$ independent of h such that

$$\|v\|_{0,e}^2 \leq c \left(\|\nabla v\|_{0,K} + h_K^{-1} \|v\|_{0,K} \right) \|v\|_{0,K}. \quad (2.54)$$

Applying the arithmetic-geometric mean inequality to the right side, we can derive

$$\|v\|_{0,e} \leq c \left(h_k^{\frac{1}{2}} \|\nabla v\|_{0,K} + h_K^{-\frac{1}{2}} \|v\|_{0,K} \right). \quad (2.55)$$

If $v \in H^1(\Omega_h)$ is in a piecewise polynomial space, we can derive the following inequality from Lemma 9 and the inverse inequality (Lemma 8):

$$\|v\|_{0,e} \leq ch_K^{-\frac{1}{2}} \|v\|_{0,K}. \quad (2.56)$$

Chapter 3

IMEX HDG-DG

In this chapter ¹, we propose IMEX HDG-DG schemes for shallow water and Euler systems. Of interest is subcritical flow in shallow water systems and low Mach number flow in Euler systems, where the speed of the gravity wave in shallow water systems and that of the acoustic wave in Euler systems are faster than that of nonlinear advection. In order to simulate these flows efficiently, we split the governing system into a stiff part describing the fast waves and a non-stiff part associated with nonlinear advection. The former is discretized implicitly with the HDG method while an explicit Runge-Kutta DG discretization is employed for the latter. The proposed IMEX HDG-DG framework: 1) facilitates high-order solutions both in time and space; 2) avoids overly small time-step sizes; 3) requires only one linear system solve per time stage; 4) relative to DG generates smaller and sparser linear systems while promoting further parallelism; and 5) suppresses the fast modes in the system with a large time-step size. Numerical results for various test cases demonstrate that our methods are beneficial for applications where slow modes are accurately treated while fast modes are inaccurately handled, i.e., a fast and stable solution is more important than an accurate solution.

We start by briefly discussing a class of implicit-explicit Runge-Kutta (IMEX-

¹ The contents of this chapter are largely based on the manuscript [105], a reduced version of which is under revision in the journal of computational physics. The contributions of the author ranged from the key ideas of the algorithm, numerical implementation, participation in the theoretical analysis and writing the manuscript.

RK) time integrators in Section 3.1. In Section 3.2, we present shallow water systems for planar and spherical surfaces. Of importance is the introduction of a linear-nonlinear splitting of the flux tensor to separate the fast wave. This is done by a linearization of the flux tensor around the “lake at rest” condition (to be defined). Next we present, in detail, a coupled HDG-DG spatial discretization for the split system in Section 3.4. The well-posedness of the semi-discrete HDG system and its rigorous convergence analysis are simultaneously shown for both planar and spherical geometries. In Section 3.5, we present an IMEX Runge-Kutta method for the semi-discrete HDG-DG system as well as the procedure for solving the implicit HDG part. Various numerical results for the shallow water systems for both planar and spherical flows will be presented in Section 3.6 to confirm the accuracy and efficiency of the proposed IMEX HDG-DG scheme. Finally we summarize and discuss this chapter in Section 3.7.

3.1 Implicit-Explicit (IMEX) Runge-Kutta methods

In this section, we briefly describe the key ideas behind a class of IMEX Runge-Kutta (IMEX-RK) methods. The readers are referred to [12, 165, 42, 220] for more details. We employ standard letters for scalars, boldface letters for vectors and calligraphic letters for tensors. Let us start with a notation of stiffness. An initial value problem is stiff if explicit time methods become numerically unstable unless extremely small time-step size is taken (This is due to the limited size of the stability regions of the explicit time methods). For a linear problem, the stiffness can be characterized by condition number, i.e., the ratio of the smallest eigenvalue to the largest eigenvalue in the system. When the magnitudes of the eigenvalues are significantly different, the system is said to be stiff. The stiffness can be geometrically induced or incurred by multi-scale features in the system. In this study, we are interested in deal-

ing with scale-separation stiffness. Especially, in a shallow water system, the scale-separation stiffness can be described by the Froude number(=advection speed/gravity wave speed) and thus the subcritical flow is considered as a stiff problem.

Let us begin by considering the following system of ordinary differential equations

$$\frac{d\mathbf{q}}{dt} = \mathbf{f}(\mathbf{q}) + \mathbf{g}(\mathbf{q}), \quad t \in (0, T), \quad (3.1)$$

with the initial condition $\mathbf{q}(0) = \mathbf{q}_0$. The functions \mathbf{f} and \mathbf{g} correspond to the non-stiff (slow time-varying) and the stiff (fast time-varying) parts, respectively. Note that they could be the result of applying two different spatial discretizations (e.g. DG and HDG methods as in this paper) for two differential operators associated with slow and fast waves. Here, we employ explicit Runge-Kutta methods for the temporal evolution corresponding to $\mathbf{f}(\mathbf{q})$ and diagonally implicit Runge-Kutta (DIRK) methods with s stages for the temporal evolution corresponding to $\mathbf{g}(\mathbf{q})$. Combining these temporal discretizations into one formula gives the IMEX-RK scheme at the i th stage [12, 165, 42]:

$$\mathbf{Q}^{(i)} = \mathbf{q}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{f}_j + \Delta t \sum_{j=1}^i \tilde{a}_{ij} \mathbf{g}_j, \quad i = 1, \dots, s, \quad (3.2a)$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \sum_{i=1}^s b_i \mathbf{f}_i + \Delta t \sum_{i=1}^s \tilde{b}_i \mathbf{g}_i, \quad (3.2b)$$

where $\mathbf{f}_i = \mathbf{f}(t^n + c_i \Delta t, \mathbf{Q}^{(i)})$, $\mathbf{g}_i = \mathbf{g}(t^n + \tilde{c}_i \Delta t, \mathbf{Q}^{(i)})$, $\mathbf{q}^n = \mathbf{q}(t^n)$ and $\mathbf{Q}^{(i)}$ is the i th intermediate state; here Δt is the time-step size. The scalar coefficients a_{ij} , \tilde{a}_{ij} , b_i , \tilde{b}_i , c_i and \tilde{c}_i determine all the properties of a given IMEX-RK scheme.

The actual forms of the non-stiff term $\mathbf{f}(\mathbf{q})$ and stiff term $\mathbf{g}(\mathbf{q})$ for our proposed coupled HDG-DG discretization for shallow water systems will be described in Section 3.5.

3.2 Shallow water equations

The homogeneous shallow water system in conservative form can be written as follows

$$\frac{\partial H}{\partial t} + \nabla \cdot (H\mathbf{u}) = 0, \quad (3.3a)$$

$$\frac{\partial (H\mathbf{u})}{\partial t} + \nabla \cdot \left(H\mathbf{u} \otimes \mathbf{u} + \frac{gH^2}{2} \mathcal{J}_d \right) = \mathbf{0}, \quad (3.3b)$$

where H is the total water depth, \mathbf{u} the horizontal velocity, d the dimension, \mathcal{J}_d the $d \times d$ identity matrix, and g the gravitational acceleration.

We can rewrite (3.3) as

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{U} = 0, \quad (3.4a)$$

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \left(\frac{\mathbf{U} \otimes \mathbf{U}}{\phi} + \frac{\phi^2}{2} \mathcal{J}_d \right) = \mathbf{0}, \quad (3.4b)$$

where $\phi = gH$ is the geopotential height and $\mathbf{U} = \phi\mathbf{u}$.

The nonlinear shallow water system (3.4) has two characteristic time scales: nonlinear advection and gravity waves with corresponding speeds $|\mathbf{u}|$ and $\sqrt{\phi}$, respectively. *In this paper, we consider subcritical flow ($|\mathbf{u}| < \sqrt{\phi}$), i.e., the differential operator associated with gravity waves is stiff.*

3.2.1 Planar shallow water equations

We first consider the two-dimensional shallow water equations on a plane. We split the total water column H into η and B such that $H = \eta + B$, where η is the free surface elevation over a reference plane (positive upward), and B is the water depth under the reference plane (positive downward), which is assumed to be constant in

time. Following [87], the governing equation (3.4) can be rewritten as

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathcal{F} = \mathbf{s} \quad \text{in } \Omega, \quad (3.5a)$$

$$\mathbf{q} = \mathbf{g}_D \quad \text{on } \Gamma_D, \quad (3.5b)$$

$$\mathcal{F} \cdot \mathbf{n} = \mathbf{g}_N \quad \text{on } \Gamma_N, \quad (3.5c)$$

where $\Omega \subset \mathbb{R}^2$ is a *planar domain*, $\partial\Omega = \bar{\Gamma}_D \cup \bar{\Gamma}_N$ is the boundary, $\mathbf{n} = (n_x, n_y)$ is outward unit normal vector on $\partial\Omega$, and $\mathbf{q} = (\phi_\eta, \mathbf{U})^T := (g\eta, (U, V))^T$ are the conservative variables. Here, $\mathcal{F} = (\mathbf{F}_x, \mathbf{F}_y)$ defined by

$$\mathbf{F}_x = \begin{pmatrix} \frac{UU}{\phi} + \frac{1}{2}\phi_\eta^2 + \phi_\eta\phi_B \\ \frac{UV}{\phi} \end{pmatrix}, \quad \mathbf{F}_y = \begin{pmatrix} \frac{VU}{\phi} \\ \frac{VV}{\phi} + \frac{1}{2}\phi_\eta^2 + \phi_\eta\phi_B \end{pmatrix}, \quad (3.6)$$

is the flux tensor, $\mathbf{s} = \left(0, \phi_\eta \frac{\partial \phi_B}{\partial x}, \phi_\eta \frac{\partial \phi_B}{\partial y}\right)^T$ the source vector, and $\phi_B = gB$ the reference geopotential height.

We can extract the fast gravity wave term (stiff operator), by linearizing the flux tensor (3.6) around the “lake at rest” condition, i.e., $\eta = 0$ and $\mathbf{u} = \mathbf{0}$, to obtain the linearized flux \mathcal{F}_L corresponding to the fast gravity wave [174, 87] as

$$\mathcal{F}_L = \begin{pmatrix} U & V \\ \phi_\eta\phi_B & 0 \\ 0 & \phi_\eta\phi_B \end{pmatrix}. \quad (3.7)$$

3.2.2 Shallow water equations on a sphere

In this paper, we are also interested in the shallow water equations on the Earth surface, and for that reason, we consider the two-dimensional shallow water equations on the sphere with the Earth radius $a = 6.371 \times 10^6$ m. We adopt the Lagrange multiplier approach [53, 91, 84, 122], i.e., we embed the two-dimensional flow on the spherical manifold into the three-dimensional space \mathbb{R}^3 . The shallow water

equations (3.4) on the spherical manifold can be cast into the following PDE in \mathbb{R}^3

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathcal{F} = \mathbf{s} \quad \text{in } \Omega, \quad (3.8)$$

where Ω is still the original surface of the sphere but now is considered a subset of \mathbb{R}^3 , $\mathbf{q} := (\phi, \mathbf{U})^T := (\phi, U, V, W)^T$ are the conservative variables, and

$$\mathcal{F} = \begin{pmatrix} U & V & W \\ \frac{UU}{\phi} + \frac{1}{2}\phi^2 & \frac{VU}{\phi} & \frac{WU}{\phi} \\ \frac{UV}{\phi} & \frac{VV}{\phi} + \frac{1}{2}\phi^2 & \frac{WV}{\phi} \\ \frac{UW}{\phi} & \frac{VW}{\phi} & \frac{WW}{\phi} + \frac{1}{2}\phi^2 \end{pmatrix} \quad (3.9)$$

is the flux tensor. Here $\mathbf{s} = (0, \mathbf{s}_U(\mathbf{q}))^T$, where $\mathbf{s}_U(\mathbf{q}) = -\phi \nabla \phi_s - f \hat{\mathbf{r}} \times \mathbf{U} + \mu \mathbf{r}$, is the source vector, $f = 2\Omega \sin \theta$ is the Coriolis parameter, Ω is the Earth's angular velocity, θ is the latitude coordinate, $\mathbf{r} = (x, y, z)$ is the position vector on the sphere, $\hat{\mathbf{r}} = \mathbf{r} a^{-1}$ is the unit normal vector on the sphere, ϕ_s is the surface topography, and μ is the Lagrange multiplier. In this approach, the tangential velocity on the sphere is denoted by $\mathbf{u} = (u, v, w)$ in the Cartesian coordinate system. Clearly, the additional degree of freedom allows fluid particles to depart from the spherical surface. One way to avoid this undesirable effect is to introduce a fictitious force via a Lagrange multiplier, which is chosen such that the velocity has no radial component on the sphere, i.e. $\mathbf{u} \cdot \mathbf{r} = 0$ [84]. By taking a dot product of \mathbf{r} and the momentum equation in (3.8), we have

$$\mathbf{r} \cdot \frac{\partial \mathbf{U}}{\partial t} = \mathbf{r} \cdot R_U + \mu \mathbf{r} \cdot \mathbf{r}, \quad (3.10)$$

where $R_U = -\nabla \cdot \left(\frac{\mathbf{U} \otimes \mathbf{U}}{\phi} + \frac{\phi^2}{2} \mathcal{J}_3 \right) - \phi \nabla \phi_s - f \hat{\mathbf{r}} \times \mathbf{U}$. Using the conditions $\mathbf{u} \cdot \mathbf{r} = 0$ and $\frac{\partial \mathbf{r}}{\partial t} = 0$, we obtain the Lagrange multiplier $\mu = -\frac{\mathbf{r} \cdot R_U}{a^2}$. Substituting μ into the momentum equation yields

$$\frac{\partial \mathbf{U}}{\partial t} = (\mathcal{J}_3 - \hat{\mathbf{r}} \hat{\mathbf{r}}^T) R_U, \quad (3.11)$$

which maps the momentum equation onto the local tangential plane. Note that $\hat{\mathbf{r}}\hat{\mathbf{r}}^T$ is the orthogonal projector that takes vectors to the direction normal to the sphere and, consequently, $(\mathcal{I}_3 - \hat{\mathbf{r}}\hat{\mathbf{r}}^T)$ is the complementary projector which takes all vectors along the tangent to the spherical surface.

Similar to Section 3.2.1, we extract the fast gravity wave by linearizing the flux tensor (3.9) around the lake at rest condition, i.e., background geopotential height $\phi = \phi_B$ and zero horizontal velocity $\mathbf{U} = 0$. We obtain the linearized flux \mathcal{F}_L containing the fast gravity waves:

$$\mathcal{F}_L = \begin{pmatrix} U & V & W \\ \phi_B \phi & 0 & 0 \\ 0 & \phi_B \phi & 0 \\ 0 & 0 & \phi_B \phi \end{pmatrix}. \quad (3.12)$$

We now show that the dynamics corresponding to the linearized differential operator (associated with the fast waves) either in (3.7) or (3.12) is well-defined.

Lemma 10 (Stability). Consider the following linear system of PDEs:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathcal{F}_L = 0, \text{ in } \Omega, \quad (3.13)$$

where \mathcal{F}_L is either from (3.7) or (3.12). Suppose (3.13) is equipped with either wall boundary conditions, i.e. $\mathbf{U} \cdot \mathbf{n} = 0$ on $\partial\Omega$ where \mathbf{n} is the unit outward normal vector, or periodic boundary conditions, then it is well-defined in the following sense

$$\frac{\partial E}{\partial t} = 0, \quad (3.14)$$

where the energy E is defined as $E = \int_{\Omega} \phi_{\eta}^2 d\Omega + \int_{\Omega} \phi_B^{-1} \mathbf{U} \cdot \mathbf{U} d\Omega$.

Proof. We proceed by an energy approach. Specifically, taking the L^2 -inner product of the mass conservation equation with ϕ_{η} and the momentum equation with $\phi_B^{-1} \mathbf{U}$, and then adding the resulting equations together we have

$$\frac{1}{2} \frac{\partial E}{\partial t} + \int_{\Omega} \phi_{\eta} \nabla \cdot \mathbf{U} d\Omega + \int_{\Omega} \mathbf{U} \cdot \nabla \phi_{\eta} d\Omega = 0,$$

which yields (3.14) after integrating the second term by parts and applying the boundary conditions. That is, the energy of the linearized shallow water system (3.13) remains constant over time. \square

3.3 Euler equations

In this section, we are interested in the compressible Euler equation, called *non-hydrostatic equations* in the atmospheric science community, where the acoustic wave is the fastest wave in the systems. The governing equation for the non-hydrostatic atmosphere is written as follows [86, 206, 3]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (3.15a)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p \mathcal{I}) = -\rho g \mathbf{k} + \nabla \cdot (\nu \rho \nabla \mathbf{u}), \quad (3.15b)$$

$$\frac{\partial \rho \theta}{\partial t} + \nabla \cdot (\rho \theta \mathbf{u}) = \nabla \cdot (\nu \rho \nabla \theta), \quad (3.15c)$$

where ρ is the density, \mathbf{u} is the velocity, \mathcal{I} is the rank-2 or 3 identity matrix, g is the gravitational constant, \mathbf{k} is the unit normal vector along the vertical direction, ν is the kinematic viscosity, and θ is the potential temperature². If a parcel of air at temperature T and pressure p is adiabatically brought to a reference pressure p_0 , then its potential temperature is given by

$$\theta = T \left(\frac{p_0}{p} \right)^{1-1/\gamma}.$$

Here, $\gamma = \frac{c_p}{c_v}$ is the ratio of specific heats, c_p and c_v are the specific heats for constant pressure and volume. The pressure p is derived from the equation of state

$$p = p_0 \left(\frac{\rho \theta R}{p_0} \right)^\gamma$$

²One can consider the potential temperature and the reference pressure as the total temperature and the total pressure for the isentropic flows.

with R is the gas constant for dry air. We can rewrite (3.15) in terms of conserved variables [86]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{U} = 0, \quad (3.16a)$$

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \left(\frac{\mathbf{U} \otimes \mathbf{U}}{\rho} + p\mathbf{J} \right) = -\rho g \mathbf{k} + \nabla \cdot \left(\nu \rho \nabla \frac{\mathbf{U}}{\rho} \right), \quad (3.16b)$$

$$\frac{\partial \Theta}{\partial t} + \nabla \cdot \left(\frac{\Theta \mathbf{U}}{\rho} \right) = \nabla \cdot \left(\nu \rho \nabla \frac{\Theta}{\rho} \right), \quad (3.16c)$$

where $\mathbf{U} = \rho \mathbf{u}$ and $\Theta = \rho \theta$. We introduce the splitting variables for simplicity,

$$\rho(\mathbf{x}) = \rho'(\mathbf{x}) + \bar{\rho}(z),$$

$$\theta(\mathbf{x}) = \theta'(\mathbf{x}) + \bar{\theta}(z),$$

$$p(\mathbf{x}) = p'(\mathbf{x}) + \bar{p}(z).$$

This leads to

$$\rho\theta = (\rho' + \bar{\rho})(\theta' + \bar{\theta}) = (\rho'\theta' + \rho'\bar{\theta} + \bar{\rho}\theta') + \bar{\rho}\bar{\theta}.$$

Similar to [23], we define $(\rho\theta)' := \rho'\theta' + \rho'\bar{\theta} + \bar{\rho}\theta'$ and $\bar{\rho\theta} := \bar{\rho}\bar{\theta}$. In addition, from the isentropic relation, the pressure term can be written in terms of $\rho\theta$ as

$$p(\rho\theta) = p(\bar{\rho\theta} + (\rho\theta)') = p(\bar{\rho\theta}) + \frac{\partial p}{\partial \rho\theta}(\rho\theta)' + \mathcal{O}((\rho\theta)')^2.$$

So we define $\bar{p} := p(\bar{\rho\theta})$ and $p' := p - \bar{p}$. For linearized pressure, we define $p'_L := \frac{\partial p}{\partial \rho\theta}(\rho\theta)' = \frac{\bar{a}^2}{\theta}$. With the splitting variables and the hydrostatic balance equation, i.e.,

$$\frac{d\bar{p}}{dz} = -\bar{\rho}g,$$

we can simplify (3.16) as

$$\frac{\partial \rho'}{\partial t} = -\nabla \cdot \mathbf{U}, \quad (3.17a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\nabla \cdot \left(\frac{\mathbf{U} \otimes \mathbf{U}}{\rho} + p'\mathbf{J} \right) - \rho'g\mathbf{k} + \nabla \cdot \left(\nu \rho \nabla \frac{\mathbf{U}}{\rho} \right), \quad (3.17b)$$

$$\frac{\partial \Theta'}{\partial t} = -\nabla \cdot \left(\frac{\Theta \mathbf{U}}{\rho} \right) + \nabla \cdot \left(\nu \rho \nabla \frac{\Theta}{\rho} \right). \quad (3.17c)$$

For the IMEX scheme, we define a linear operator which contains the fast waves. To this end, we linearize the flux tensor

$$\mathcal{F} = \underbrace{\begin{pmatrix} \mathbf{U} \\ \frac{\mathbf{U} \otimes \mathbf{U}}{\rho} + p' \mathcal{J} \\ \frac{\Theta \mathbf{U}}{\rho} \end{pmatrix}}_{\text{inviscid}} - \underbrace{\begin{pmatrix} 0 \\ \nu \rho \nabla \frac{\mathbf{U}}{\rho} \\ \nu \rho \nabla \frac{\bar{\Theta}}{\rho} \end{pmatrix}}_{\text{viscous}} \quad (3.18)$$

around a hydrostatic balanced solution [88], which leads to

$$\frac{\partial \rho'}{\partial t} = -\nabla \cdot (\mathbf{U}), \quad (3.19a)$$

$$\frac{\partial \mathbf{U}}{\partial t} = -\nabla \cdot (p'_L \mathcal{J}) - \rho' g \mathbf{k}, \quad (3.19b)$$

$$\frac{\partial \Theta'}{\partial t} = -\nabla \cdot (\mathbf{U} \bar{\theta}), \quad (3.19c)$$

where $p'_L := \frac{\bar{a}^2}{\bar{\theta}}$, $\bar{\theta} := \frac{\bar{\rho} \bar{\theta}}{\bar{\rho}} = \frac{\Theta'}{\bar{\rho}}$ and $\bar{a}^2 = \frac{\gamma \bar{p}}{\bar{\rho}}$. (Note that we only linearize the inviscid Euler flux.) Now, the linearized flux and its Jacobian read

$$\mathcal{F}_L = \begin{pmatrix} \mathbf{U} \\ p'_L \mathcal{J} \\ \frac{\Theta \mathbf{U}}{\bar{\rho}} \end{pmatrix} \text{ and } \mathcal{A} := \frac{\partial \mathbf{n} \cdot \mathcal{F}_L}{\partial q} = \begin{pmatrix} 0 & \mathbf{n}^T & 0 \\ 0 & 0 & \mathbf{n} \gamma \frac{\bar{p}}{\bar{\Theta}} \\ 0 & \mathbf{n}^T \frac{\bar{\Theta}}{\bar{\rho}} & 0 \end{pmatrix}. \quad (3.20)$$

We observe that the eigenvalues of (3.20) are $-\bar{a}$, 0 and \bar{a} .

3.4 Spatial Discretization

3.4.1 DG and HDG spatial discretization

The DG discretization [128, 96, 89] for either (3.5) or (3.8) can be written in the following form: seek $\mathbf{q} \in \mathbf{V}_h(K)$ such that the weak formulation

$$\left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v} \right)_K - (\mathcal{F}(\mathbf{q}), \nabla \mathbf{v})_K + \langle \mathcal{F}^*(\mathbf{q}^\pm) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial K} = (\mathbf{s}, \mathbf{v})_K, \quad (3.21)$$

holds for each element $K \in \Omega_h$, where $\mathcal{F}^*(\mathbf{q}^\pm)$ is a numerical flux [126] such as the Lax-Friedrichs (i.e., Rusanov) [181] or Roe [179] flux. Note that the standard numerical flux $\mathcal{F}^*(\mathbf{q}^\pm)$ is a function of the solution traces \mathbf{q}^\pm from both sides of ∂K .

For convenience, we have ignored the fact that (3.21) must hold for all test functions $\mathbf{v} \in \mathbf{V}_h(K)$; throughout this chapter, this should be implicitly understood.

The key idea of the HDG framework is to introduce a new single-valued numerical trace $\hat{\mathbf{q}}$ on the mesh skeleton [155, 47, 147, 30] so that the numerical flux is now the function of the solution in element K and $\hat{\mathbf{q}}$. In particular, the weak formulation for the HDG discretization (compared with the DG discretization in (3.21)) reads

$$\left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v} \right)_K - (\mathcal{F}(\mathbf{q}), \nabla \mathbf{v})_K + \left\langle \hat{\mathcal{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}, \mathbf{v} \right\rangle_{\partial K} = (\mathbf{s}, \mathbf{v})_K, \quad (3.22)$$

where $\hat{\mathcal{F}}$ is a hybridization of the numerical flux $\mathcal{F}^*(\mathbf{q}^\pm)$ in (3.21), and $\hat{\mathbf{q}}$ approximates \mathbf{q} on \mathcal{E}_h . In other words, we have hybridized the DG formulation (3.21) to obtain the HDG formulation (3.22). Since we introduce a new variable, $\hat{\mathbf{q}}$, we need one more equation to close the system. To that end, we note that for the HDG discretization (3.22) to be conservative the HDG flux $\hat{\mathcal{F}}$ needs to be continuous across the mesh skeleton. Thus, a natural equation (a sufficient condition for conservation) is a weak continuity of the HDG normal flux on each interface $e \in \partial K$, i.e.,

$$\left\langle \left[\hat{\mathcal{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_e = 0, \quad (3.23)$$

for all $\boldsymbol{\mu} \in \boldsymbol{\Lambda}_h(e)$. By summing (3.22) over all elements and (3.23) over the mesh skeleton we obtain the complete HDG discretization: find the approximate solution $(\mathbf{q}, \hat{\mathbf{q}}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$ such that

$$\left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v} \right)_{\Omega_h} - (\mathcal{F}(\mathbf{q}), \nabla \mathbf{v})_{\Omega_h} + \left\langle \hat{\mathcal{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}, \mathbf{v} \right\rangle_{\partial \Omega_h} = (\mathbf{s}, \mathbf{v})_{\Omega_h}, \quad (3.24a)$$

$$\left\langle \left[\hat{\mathcal{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.24b)$$

for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$, where the numerical flux $\hat{\mathcal{F}}$ can be defined as [46, 156, 30]

$$\hat{\mathcal{F}}(\mathbf{q}, \hat{\mathbf{q}}) = \mathcal{F}(\mathbf{q}) + \tau (\mathbf{q} - \hat{\mathbf{q}}) \otimes \mathbf{n}, \quad (3.25)$$

with τ as the stabilization parameter (to be described in detail later).

3.4.2 Coupled HDG-DG spatial discretization

As discussed in Section 3.2 we decompose the nonlinear differential operator associated with the shallow water equations into a linear (stiff) part $\nabla \cdot \mathcal{F}_L$ and a nonlinear (non-stiff) part $\nabla \cdot (\mathcal{F} - \mathcal{F}_L)$. Unlike most of the existing literature, our decomposition is on the continuous level instead of the discrete one. The advantage of this strategy is that it allows one to employ two separate spatial discretizations for the stiff and non-stiff parts, respectively. In this chapter, we choose HDG for the former and DG for the latter. Clearly, we can choose DG [24, 152, 33, 30, 128, 4, 89, 87] for the former as well but, as will be shown, HDG provides several advantages over DG including lower storage and more efficiency. The coupled HDG-DG discretization (see section 3.4.1) of the decomposed system reads: seek $(\mathbf{q}, \hat{\mathbf{q}}) \in \mathbf{V}_h(\Omega_h) \times \mathbf{\Lambda}_h(\mathcal{E}_h)$ such that

$$\left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v} \right)_{\Omega_h} = \mathcal{NL}(\mathbf{q}) + \mathcal{L}(\mathbf{q}, \hat{\mathbf{q}}), \quad (3.26a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.26b)$$

for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \mathbf{\Lambda}_h(\mathcal{E}_h)$, where

$$\begin{aligned} \mathcal{NL}(\mathbf{q}) &= (\mathcal{F}_{NL}(\mathbf{q}), \nabla \mathbf{v})_{\Omega_h} - \langle \mathcal{F}_{NL}^*(\mathbf{q}^\pm) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h}, \\ \mathcal{L}(\mathbf{q}, \hat{\mathbf{q}}) &= (\mathcal{F}_L(\mathbf{q}), \nabla \mathbf{v})_{\Omega_h} + (\mathbf{s}(\mathbf{q}), \mathbf{v})_{\Omega_h} - \left\langle \hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}, \mathbf{v} \right\rangle_{\partial\Omega_h}. \end{aligned}$$

Here, $\mathcal{F}_{NL} := \mathcal{F} - \mathcal{F}_L$, $\mathcal{F}_{NL}^* := \mathcal{F}^* - \mathcal{F}_L^*$ is a nonlinear DG numerical flux, and $\hat{\mathcal{F}}_L$ is a linear HDG numerical flux. We now present a choice for these numerical fluxes.

3.4.2.1 Shallow water equations

For the two-dimensional shallow water equations on a plane, we choose the Lax-Friedrichs numerical flux [181, 204] for the DG discretization and the upwind

HDG flux [33]:

$$\mathcal{F}^*(\mathbf{q}) = \{\!\!\{ \mathcal{F}(\mathbf{q}) \}\!\!\} + \frac{\tau^*}{2} [\mathbf{q} \otimes \mathbf{n}], \quad (3.27a)$$

$$\mathcal{F}_L^*(\mathbf{q}) = \{\!\!\{ \mathcal{F}_L(\mathbf{q}) \}\!\!\} + \frac{\tau_L^*}{2} [\mathbf{q} \otimes \mathbf{n}], \quad (3.27b)$$

$$\hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) = \mathcal{F}_L(\mathbf{q}) + \hat{\tau}(\mathbf{q} - \hat{\mathbf{q}}) \otimes \mathbf{n}, \quad (3.27c)$$

where $\tau^* = \max \left((|\mathbf{u} \cdot \mathbf{n}| + \sqrt{\phi})^+, (|\mathbf{u} \cdot \mathbf{n}| + \sqrt{\phi})^- \right)$, $\tau_L^* = \max \left(\sqrt{\phi_B^+}, \sqrt{\phi_B^-} \right)$, and $\hat{\tau} = \tau_L^*$. Note that τ^* is the (advection + gravity) wave speed of the shallow water equations, while τ_L^* is the (gravity) wave speed of the stiff term. Here, a hybridized Lax-Friedrichs flux³, (3.27c), is defined [33] as

$$\mathbf{n} \cdot \hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) = \begin{pmatrix} n_x U + n_y V + \sqrt{\phi_B} (\phi_\eta - \hat{\phi}_\eta) \\ n_x \phi_B \phi_\eta + \sqrt{\phi_B} (U - \hat{U}) \\ n_y \phi_B \phi_\eta + \sqrt{\phi_B} (V - \hat{V}) \end{pmatrix}. \quad (3.28)$$

For the two-dimensional shallow water equations on a sphere, the Lax-Friedrichs flux for DG methods has the same form as (3.27a) and (3.27b), while the hybridized Lax-Friedrichs flux is defined as

$$\mathbf{n} \cdot \hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) = \begin{pmatrix} n_x U + n_y V + n_z W + \sqrt{\phi_B} (\phi - \hat{\phi}) \\ n_x \phi_B \phi + \sqrt{\phi_B} (U - \hat{U}) \\ n_y \phi_B \phi + \sqrt{\phi_B} (V - \hat{V}) \\ n_z \phi_B \phi + \sqrt{\phi_B} (W - \hat{W}) \end{pmatrix}. \quad (3.29)$$

For periodic boundary condition (or similarly no boundary in spherical cases), all faces are interior faces, and hence no special treatment is needed. To enforce the wall boundary condition, we use a reflection principle. In particular, for an element

³Note that for polygonal domain Ω , (3.27b) and (3.27c) are the same [30, 33], and hence there is no splitting error. Otherwise, the splitting error is of order $\mathcal{O} \left(h^{p+\frac{1}{2}} \right)$, which is the same as the convergence order, and thus not affecting the convergence rate of the whole scheme.

K^- that is adjacent to the domain boundary, i.e. $\partial K^- \cap \partial\Omega \neq \emptyset$, we assume that there is an imaginary neighbor element K^+ whose state $\mathbf{q}^+ = (\phi^+, \mathbf{U}^+)^T$ is determined as

$$\phi^+ = \phi^-, \quad (3.30a)$$

$$\mathbf{U}^+ = \mathbf{U}^- - 2(\mathbf{U}^- \cdot \mathbf{n}^-) \mathbf{n}^-, \quad (3.30b)$$

which, together with the conservation condition (3.26b) and the HDG flux (3.28) (or (3.29)) on boundary faces $e \in \partial K^- \cap \partial\Omega$, leads to

$$\left\langle \mathbf{U} \cdot \mathbf{n} + \sqrt{\phi_B} (\phi - \hat{\phi}), \mu \right\rangle_{\partial K^- \cap \partial\Omega} = 0, \quad (3.31a)$$

$$\left\langle \sqrt{\phi_B} (\mathbf{U}^t - \hat{\mathbf{U}}^t), \boldsymbol{\mu}^t \right\rangle_{\partial K^- \cap \partial\Omega} = 0, \quad \hat{\mathbf{U}} \cdot \mathbf{n} = 0, \quad (3.31b)$$

where the superscript “t” denotes the tangential part.

3.4.2.2 Euler equations

For Euler equations, we take $\tau^* = \max((|\mathbf{u} \cdot \mathbf{n}| + a)^+, (|\mathbf{u} \cdot \mathbf{n}| + a)^-)$ and $\tau_L^* = \hat{\tau} = \bar{a}$ as the stabilization parameters in (3.27). A hybridized Lax-Friedrich flux is defined as

$$\mathbf{n} \cdot \hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) = \begin{pmatrix} \mathbf{n} \cdot \mathbf{U} + \hat{\tau}(\rho' - \hat{\rho}') \\ \mathbf{n} p'_L + \hat{\tau}(\mathbf{U} - \hat{\mathbf{U}}) \\ \mathbf{n} \cdot \mathbf{U} \frac{\bar{\theta}}{\bar{\rho}} + \hat{\tau}(\Theta' - \hat{\Theta}') \end{pmatrix}. \quad (3.32)$$

As for the wall boundary condition, similar to (3.30), we use the ghost element approach as

$$\rho'^+ = \rho'^-, \quad (3.33a)$$

$$\mathbf{U}^+ = \mathbf{U}^- - 2(\mathbf{U}^- \cdot \mathbf{n}^-) \mathbf{n}^-, \quad (3.33b)$$

$$\Theta'^+ = \Theta'^-. \quad (3.33c)$$

3.5 Temporal Discretization

In this section, we adapt the general IMEX-RK idea in Section 3.1 to the semi-discrete system (3.26). In particular, the i th stage IMEX-RK stated in (3.2), when specified to (3.26), reads

$$\mathbf{Q}^{(i)} = \mathbf{q}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{M}^{-1} \mathcal{N} \mathcal{L}_j + \Delta t \sum_{j=1}^i \tilde{a}_{ij} \mathbf{M}^{-1} \mathcal{L}_j, \quad i = 1, \dots, s, \quad (3.34a)$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \sum_{i=1}^s b_i \mathbf{M}^{-1} \mathcal{N} \mathcal{L}_i + \Delta t \sum_{i=1}^s \tilde{b}_i \mathbf{M}^{-1} \mathcal{L}_i, \quad (3.34b)$$

where $\mathcal{N} \mathcal{L}_i := \mathcal{N} \mathcal{L}(\mathbf{Q}^{(i)})$ and $\mathcal{L}_i := \mathcal{L}(\mathbf{Q}^{(i)}, \hat{\mathbf{Q}}^{(i)})$; \mathbf{M} is a mass matrix. Due to the last term on the right-hand side, the i th stage equation (3.34a) is implicit in both $\mathbf{Q}^{(i)}$ and $\hat{\mathbf{Q}}^{(i)}$. They can be solved by combining (3.34a) and (3.26b). Since \mathcal{L}_j is a result of the HDG discretization, this combination is nothing more than an HDG discretization with the local equation and the conservation condition defined as

$$\mathbf{Q}^{(i)} - \Delta t \tilde{a}_{ii} \mathbf{M}^{-1} \mathcal{L}_i = \mathcal{R}es_0, \quad (3.35a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{Q}^{(i)}, \hat{\mathbf{Q}}^{(i)}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\varepsilon_h} = 0, \quad (3.35b)$$

where $\mathcal{R}es_0 = \mathbf{q}^n + \Delta t \mathbf{M}^{-1} \sum_{j=1}^{i-1} (a_{ij} \mathcal{N} \mathcal{L}_j + \tilde{a}_{ij} \mathcal{L}_j)$.

To solve the HDG system (3.35), we note that both equations are linear in $\mathbf{Q}^{(i)}$ and $\hat{\mathbf{Q}}^{(i)}$, and can be written as a coupled linear system. We define

$$\mathcal{R}es(\mathbf{Q}, \hat{\mathbf{Q}}) = \mathbf{Q} - \Delta t \tilde{a}_{ii} \mathbf{M}^{-1} \mathcal{L}(\mathbf{Q}, \hat{\mathbf{Q}}) - \mathcal{R}es_0, \quad (3.36a)$$

$$\mathcal{F}lx(\mathbf{Q}, \hat{\mathbf{Q}}) = \left\langle \left[\hat{\mathcal{F}}_L(\mathbf{Q}, \hat{\mathbf{Q}}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\varepsilon_h}. \quad (3.36b)$$

The HDG system (3.35) can be written algebraically as

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} \begin{pmatrix} \mathbf{Q}^{(i)} \\ \hat{\mathbf{Q}}^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{pmatrix}, \quad (3.37)$$

where $\mathbf{A} = I - \Delta t \tilde{a}_{ii} \mathbf{M}^{-1} \frac{\partial \mathcal{L}}{\partial \mathbf{Q}}$, $\mathbf{B} = -\Delta t \tilde{a}_{ii} \mathbf{M}^{-1} \frac{\partial \mathcal{L}}{\partial \mathbf{Q}}$, $\mathbf{C} = \frac{\partial \mathcal{F}Lx}{\partial \mathbf{Q}}$, $\mathbf{D} = \frac{\partial \mathcal{F}Lx}{\partial \mathbf{Q}}$, $\mathbf{R}_1 = \mathcal{R}es_0$ and $\mathbf{R}_2 = 0$. Note that the term $\frac{\partial \mathcal{L}}{\partial \mathbf{Q}}$ does not involve the computation of a Jacobian. Since L is linear, $\frac{\partial \mathcal{L}}{\partial \mathbf{Q}}$ is a constant matrix.

To solve (3.37), we can first eliminate the volume unknowns $\mathbf{Q}^{(i)}$

$$\mathbf{Q}^{(i)} = \mathbf{A}^{-1} \left(\mathbf{R}_1 - \mathbf{B} \hat{\mathbf{Q}}^{(i)} \right). \quad (3.38)$$

Since \mathbf{A} is block-diagonal (each block corresponding to one element in the mesh), the inversion in (3.38) is actually done in an element-by-element fashion, completely independent of each other. This Schur complement step allows us to condense $\mathbf{Q}^{(i)}$ to arrive at a much smaller linear system of equations in terms of $\hat{\mathbf{Q}}^{(i)}$:

$$(\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B}) \hat{\mathbf{Q}}^{(i)} = \mathbf{R}_2 - \mathbf{C} \mathbf{A}^{-1} \mathbf{R}_1. \quad (3.39)$$

Once $\hat{\mathbf{Q}}^{(i)}$ is computed, the volume unknowns $\mathbf{Q}^{(i)}$ can be obtained using (3.38), in an element-by-element fashion. Compared to IMEX DG schemes [178, 70, 174, 229], our IMEX HDG-DG scheme has a smaller number of coupled unknowns. On quadrilateral meshes with $n \times n$ elements and polynomial order p , for example, the number of coupled IMEX HDG-DG unknowns is $2n(n+1)(p+1)$, whereas that of the IMEX DG is $n^2(p+1)^2$. The ratio of the IMEX DG unknowns to the IMEX HDG-DG counterparts is $\frac{p+1}{2(1+1/n)}$. The IMEX HDG-DG schemes thus become beneficial in terms of the number of coupled degrees of freedom, and hence the size of the linear system, when the solution order $p \geq 1 + 2/n$. In particular, IMEX HDG-DG becomes advantageous starting from second order approximations. A detailed complexity comparison between HDG and DG can be found in [33]. Once all the intermediate solutions are computed, the next time-step solution \mathbf{q}^{n+1} is determined through (3.34b). Algorithm 1 summarizes all the steps of our proposed IMEX scheme.

Algorithm 1 IMEX HDG-DG scheme for s -stages.

Ensure: Given solution state q^n , compute its next solution state q^{n+1} .

```

1: for  $i = 1$  to  $s$  do
2:   if  $\tilde{a}_{ii} = 0$  then
3:      $Q^{(i)} \leftarrow \mathbf{q}^n$ 
4:      $\mathcal{L}_i \leftarrow \mathcal{L}(\mathbf{Q}^{(i)})$ 
5:   else
6:      $Res0 \leftarrow q^n + \Delta t \mathbf{M}^{-1} \sum_{j=1}^{i-1} (a_{ij} \mathcal{N} \mathcal{L}_j + \tilde{a}_{ij} \mathcal{L}_j)$ 
7:     Solve for  $\hat{\mathbf{Q}}^{(i)}$  using (3.39)
8:     Obtain the volume unknowns  $\mathbf{Q}^{(i)}$  using (3.38)
9:      $\mathcal{L}_i \leftarrow \mathcal{L}(\mathbf{Q}^{(i)}, \hat{\mathbf{Q}}^{(i)})$ 
10:  end if
11:   $\mathcal{NL}_i \leftarrow \mathcal{NL}(\mathbf{Q}^{(i)})$ 
12: end for
13: Update the solution  $q^{n+1} \leftarrow q^n + \Delta t \mathbf{M}^{-1} \sum_{i=1}^s (b_i \mathcal{NL}_i + \tilde{b}_i \mathcal{L}_i)$ 

```

The IMEX methods considered in this chapter are the ARS2(2,3,2) and ARS3(4,4,3) [12] methods, which have the singly diagonally implicit Runge-Kutta (SDIRK) property. (ARK methods [85, 42] with the same order of accuracy behave similarly and hence are not shown in the chapter.) Here, the triplet (s, σ, p) denotes the s stages of the implicit scheme, σ stages of the explicit scheme, and the order of accuracy of the scheme.

3.6 Numerical Results

In this section, we demonstrate the accuracy and efficiency of the proposed coupled IMEX HDG-DG methods for the shallow water equations and the Euler equations through several numerical experiments. For planar shallow water flow, two test cases are considered: the translating vortex test case and the water height perturbation problem. For the former, in which an exact solution exists, we present the numerical convergence for both the spatial and temporal discretizations. For the latter, in which no analytical solution is available, we perform a comparison with

explicit schemes. For the spherical shallow water equations, the well-known standard test cases proposed by [225] and the barotropic instability phenomenon [77] are chosen to verify the IMEX HDG-DG scheme. For the Euler equations, three examples are tested: the inertia-gravity wave [192], the rising thermal bubble [86] and the density current flows [196].

3.6.1 Moving vortex

We consider the vortex translation test [177] in which the initial condition in the domain $\Omega = [-2, 2] \times [-2, 2]$ is chosen in such a way that the pressure gradient force and the centrifugal force are balanced. This allows the initial vortex to translate across the domain without changing its shape. The exact solution for the vortex at any time t is given by

$$H = H_\infty - \frac{\beta^2}{32\pi^2} e^{-2(r^2-1)}, \quad (3.40a)$$

$$(u, v) = (u_\infty, v_\infty) + \frac{\beta}{2\pi} e^{-(r^2-1)} (-y_t, x_t), \quad (3.40b)$$

where β is the vortex strength, (x_c, y_c) the center of the vortex, (u_∞, v_∞) the reference horizontal velocity, $x_t = x - x_c - u_\infty t$, $y_t = y - y_c - v_\infty t$, $r^2 = x_t^2 + y_t^2$, and H_∞ the reference water depth. For the numerical results in this section, we choose $H_\infty = 1$ and $(u_\infty, v_\infty) = (1, 0)$, $\beta = 5$, and $g = 2$. We use the exact solution to impose the boundary condition. Initially the vortex is located at $(x_c, y_c) = (0, 0)$. Figure 3.1 shows numerical results for the free surface elevation $\eta := H - H_\infty$ at different times. Here, the solution order is $p = 6$ and the results are computed on a uniform mesh with 32×32 elements.

We compute the errors of the free surface elevation and the velocity using the L_2 norm defined as $L_2(q) := \left(\sum_{K \in \Omega_h} \int_K (q - q_T)^2 dK \right)^{\frac{1}{2}}$, where q_T is the exact solution at the final time T . For the spatial convergence test, we use a sequence

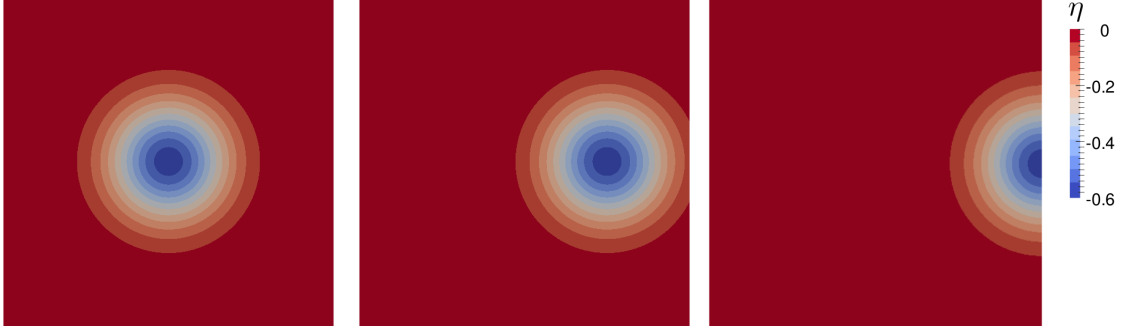


Figure 3.1: The moving vortex test case: time evolution of the free surface elevation η at times $t = 0$, $t = 1$ and $t = 2$ computed using ARS3 HDG-DG. The contour levels are from -0.6m to 0m with the step-size of 0.05m .

of nested meshes with $N_e = \{8^2, 12^2, 16^2, 20^2\}$ for $p = \{2, 3, 4, 5\}$ and measure the errors at $T = 0.1$. We choose a time-step size such that spatial discretization error is dominant over time discretization error (e.g., $\Delta t = 5.0 \times 10^{-5}$ for $p = 5$, $\Delta t = 2.0 \times 10^{-4}$ for $p = 4$, $\Delta t = 1.0 \times 10^{-3}$ for $p = 3$, $\Delta t = 1.0 \times 10^{-3}$ for $p = 2$). Table 3.1 shows the spatial convergence results of the free surface elevation η and the velocity \mathbf{u} for ARS3 HDG-DG scheme, where the energy norm is defined as $E := \frac{1}{2}(\|\eta\|_{\Omega_h}^2 + \|\mathbf{u}\|_{\Omega_h}^2)$. As can be seen, the predicted convergence rate of $(p + 1/2)$ is observed for all cases.

To numerically compute the temporal convergence for ARS2 HDG-DG and ARS3 HDG-DG, we simulate the translational vortex with a $6th$ -order solution on the 32×32 -element mesh. The time-step size Δt varies from 6.25×10^{-5} to 5×10^{-3} , which corresponds to Courant numbers (Cr) from 0.07 to 5.6 . We compute the error at $T = 0.2$. The mean water depth H_∞ is set to be 50 so that the reference Froude number, $Fr = \frac{u_\infty}{\sqrt{gH_\infty}}$, is 0.1 , that is, the gravity wave dominates the convection.

In Figure 3.2(a), we observe the correct second-order and third-order convergence in time for ARS2 HDG-DG and ARS3 HDG-DG, respectively. We also see the second-order convergence in time for RK2 DG. As for RK3 DG, we notice that

Table 3.1: Spatial convergence for the traveling vortex test using ARS3 HDG-DG.

p	N_e	$L_2(\eta)$		$L_2(\mathbf{u})$		$L_2(\sqrt{E})$	
		error	order	error	order	error	order
2	8×8	9.983e-03	—	2.585e-02	—	1.959e-02	—
	12×12	3.816e-03	2.372	1.168e-02	1.959	8.689e-03	2.006
	16×16	1.851e-03	2.515	6.248e-03	2.174	4.608e-03	2.205
	20×20	1.046e-03	2.557	3.758e-03	2.278	2.758e-03	2.300
3	8×8	1.102e-03	—	6.138e-03	—	4.410e-03	—
	12×12	3.500e-04	2.829	1.484e-03	3.502	1.078e-03	3.475
	16×16	1.281e-04	3.494	5.509e-04	3.443	4.000e-04	3.446
	20×20	5.869e-05	3.498	2.491e-04	3.557	1.810e-04	3.554
4	8×8	2.344e-04	—	7.030e-04	—	5.240e-04	—
	12×12	3.658e-05	4.581	1.475e-04	3.851	1.074e-04	3.908
	16×16	1.047e-05	4.348	4.508e-05	4.120	3.272e-05	4.133
	20×20	4.072e-06	4.233	1.727e-05	4.299	1.255e-05	4.295
5	8×8	2.340e-05	—	1.873e-04	—	1.335e-04	—
	12×12	3.557e-06	4.647	2.171e-05	5.315	1.556e-05	5.301
	16×16	7.820e-07	5.265	3.986e-06	5.892	2.872e-06	5.873
	20×20	2.303e-07	5.479	1.094e-06	5.793	7.908e-07	5.781

its error levels are saturated near 10^{-10} . This is because the spatial discretization error starts to dominate the time discretization error. To demonstrate the stability benefit of the IMEX HDG-DG scheme we perform simulations for a range of Courant numbers (Cr) from 0.28 (the point over which the second order RKDG, denoted as RK2 DG, blows up) to 5.6.

Clearly, the IMEX HDG-DG approaches are more economical than our previous work on IMEX DG [87, 85] due to the fewer number of coupled degrees of freedom in the context of a direct solver. Compared to standard fully implicit methods, they are much more advantageous since only one linear solve is needed for each stage per

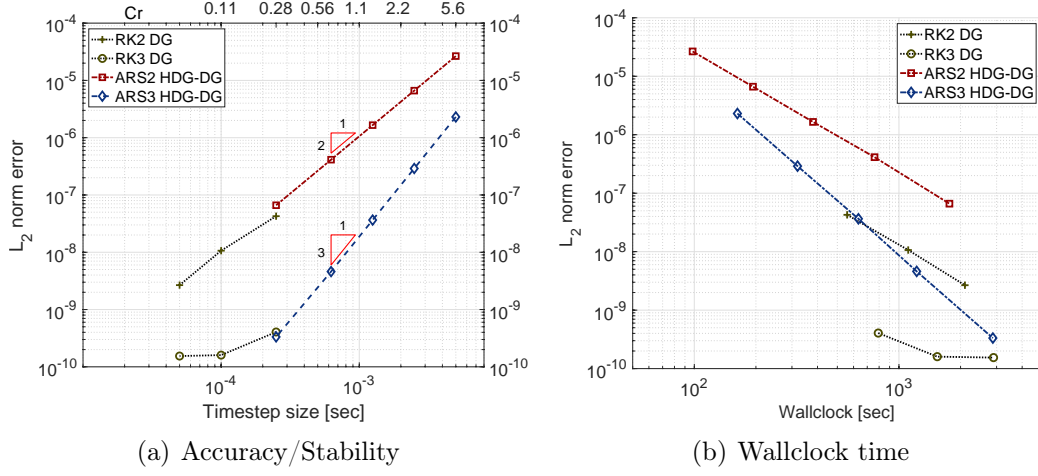


Figure 3.2: Comparison between IMEX HDG-DG and RKDG for the moving vortex test case: (a) accuracy/stability and (b) wallclock time.

time-step. For this chapter, our methods are in fact “optimal” in the sense that the HDG matrix in (3.37), and hence the matrices \mathbf{A} in (3.38) and $(\mathbf{D} - \mathbf{CA}^{-1}\mathbf{B})$ in (3.39), is the same for any time-step and any stage (since \tilde{a}_{ii} are the same at any stage for the chosen schemes). Thus, we need to perform the LU factorization (here we use UMFPACK [57]) of the HDG-trace matrix $(\mathbf{D} - \mathbf{CA}^{-1}\mathbf{B})$ once, and the same LU factors can be recycled (via a forward substitution followed by a backward substitution) for all subsequent computations involving (3.39). We also store the inverse of each diagonal block of \mathbf{A} in (3.38) once and reuse it for recovering the DG unknowns in an element-by-element fashion.

Still, it is challenging to compete with fully explicit methods in terms of wall clock time because our approach needs to solve (3.38) and (3.39) for each time-step. However, there are cases when the proposed IMEX HDG-DG methods are more economical than explicit schemes. To demonstrate this we plot in Figure 3.2(b) the L^2 error of the free surface height against the wall clock time for ARS2 HDG-DG, ARS3 HDG-DG, RK2 DG, and RK3 DG (the third order RKDG). Given an

error level, RK2 DG and RK3 DG outperform ARS2 HDG-DG and ARS3 HDG-DG, respectively. However, ARS3 HDG-DG becomes comparable to RK2 DG and outperforms RK2 DG for tighter error levels. This means that high-order IMEX HDG-DG methods can be beneficial compared to low-order explicit methods.

Not only that, when a desired level of accuracy is relaxed, IMEX HDG-DG can be efficient. For example, when the desired accuracy is $\mathcal{O}(10^{-6})$, ARS3 HDG-DG method is three times faster than RK2 DG methods in Figure 3.2(b). Indeed, the error level of $\mathcal{O}(10^{-10})$ will not be achieved when the solution is not smooth or is not well resolved in spatial discretization. In Table 3.2, we lower the solution order from $p = 6$ to $p = 4$ so that the spatial discretization error affects the temporal convergence rate. As expected, the saturation level increases from $\mathcal{O}(10^{-10})$ to $\mathcal{O}(10^{-7})$. When the desired accuracy is $\mathcal{O}(10^{-7})$, we observe that ARS3 HDG-DG with $Cr = 1.38$ is comparable to RK2 DG with $Cr = 0.28$. As shown in Table 3.3, we can obtain further computational gain when considering the case with $Fr = 0.01$ (here, we take $H_\infty = 5000$), where the saturation level is $\mathcal{O}(10^{-6})$ for $p = 4$. ARS3 HDG-DG and ARS2 HDG-DG with $Cr = 4.72$ are four times and six times faster than RK2 DG with $Cr = 0.24$, respectively. This is due to the virtue of the scale-separation in low Mach or subcritical flows, where there is a chance to take a larger time-step size, with which IMEX schemes can outperform explicit schemes [22, 231]. Clearly, the smaller the Froude Number, the better it is for IMEX methods in the shallow water systems.

To improve the wall clock time we can, for example, develop iterative solvers for (3.39) and solve (3.38); Recently, block Jacobi type preconditioner [151] and multi-grid solver [224] have been developed for HDG methods. Woopen et al. (2014)[228] have shown that HDG methods give a faster solution than DG methods for compressible flow with iterative solvers. Based on that, we can conjecture the IMEX HDG methods would be more efficient, but we need to explore iterative solvers in various

Table 3.2: Moving vortex for $Fr = 0.1$: the errors and the convergence rates for the free surface elevation η , and wallclock time at $T = 0.2$. The errors are computed over the domain $\tilde{\Omega} \in [-2, 2]^2$ with 32×32 elements.

32×32	$p = 6$				$p = 4$			
	Cr	error	order	wc [s]	Cr	error	order	wc [s]
ARS2	5.60	2.657E-05	-	98.1	5.52	1.069E-04	-	16.1
	2.80	6.627E-06	2.003	193.5	2.76	2.657E-05	2.009	31.6
	1.40	1.655E-06	2.001	380.3	1.38	6.639E-06	2.001	62.2
	0.70	4.130E-07	2.003	759.8	0.67	1.704E-06	1.962	122.8
	0.28	6.618E-08	1.998	1764.4	0.28	4.828E-07	1.376	305.7
ARS3	5.60	2.304E-06	-	162.4	5.52	1.832E-05	-	25.9
	2.80	2.901E-07	2.990	319.5	2.76	2.339E-06	2.969	51.1
	1.40	3.650E-08	2.990	633.8	1.38	4.971E-07	2.234	101.1
	0.70	4.598E-09	2.989	1220.9	0.67	4.054E-07	0.294	201.6
	0.28	3.328E-10	2.866	2876.9	0.28	4.038E-07	0.004	502.5
RK2	0.28	4.264E-08	-	558.3	0.28	4.384E-07	-	98.3
	0.14	1.065E-08	2.001	1109.8	0.14	4.060E-07	0.111	205.5
	0.07	2.668E-09	1.998	2101.0	0.07	4.039E-07	0.008	413.5
RK3	0.28	4.054E-10	-	792.4	0.28	4.038E-07	-	134.2
	0.14	1.601E-10	1.341	1539.0	0.14	4.038E-07	0.000	281.2
	0.07	1.542E-10	0.054	2898.0	0.07	4.038E-07	0.000	547.3

aspects because their performance depends on not only the size of the system but also preconditioning techniques, matrix properties and so on. We will consider this topic in our future work.

3.6.2 Water height perturbation

The IMEX methods are designed to find accurate solutions for the slow modes while damping out the high-frequency modes. [12]. This design philosophy fits atmospheric models because they simulate slow advection, Rossby waves, and slow gravity

Table 3.3: Moving vortex for $Fr = 0.01$: the errors and the convergence rates for the free surface elevation η , and wall clock time at $T = 0.1$. The errors are computed over the domain $\tilde{\Omega} \in [-2, 2]^2$ with 32×32 elements.

32×32	$p = 6$				$p = 4$			
	Cr	error	order	wc [s]	Cr	error	order	wc [s]
ARS2	10.17	9.466E-07	-	255.3	9.44	5.666E-06	-	41.5
	5.08	2.172E-07	2.124	508.7	4.72	3.294E-06	0.783	87.8
	2.54	5.269E-08	2.044	1001.4	2.36	3.162E-06	0.059	171.8
	1.27	1.306E-08	2.012	1904.2	1.18	3.155E-06	0.003	339.4
	0.64	3.407E-09	1.939	3710.1	0.59	3.155E-06	0.000	696.1
ARS3	10.17	1.902E-06	-	372.9	9.44	1.357E-05	-	70.3
	5.08	2.462E-07	2.949	732.0	4.72	3.684E-06	1.881	140.7
	2.54	3.108E-08	2.986	1458.7	2.36	3.164E-06	0.219	279.6
	1.27	4.034E-09	2.946	2941.1	1.18	3.155E-06	0.004	556.9
	0.64	1.158E-09	1.801	6505.6	0.59	3.155E-06	0.000	1118.6
RK2	0.25	1.106E-09	-	2928.6	0.24	3.155E-06	-	561.9
	0.13	1.054E-09	0.071	5467.6	0.12	3.155E-06	-0.000	1142.8
	0.06	1.050E-09	0.005	10490.8	0.06	3.155E-06	-0.000	2174.5
RK3	0.25	1.051E-09	-	4029.8	0.24	3.155E-06	-	769.7
	0.13	1.050E-09	0.001	7588.3	0.12	3.155E-06	0.000	1531.9
	0.06	1.050E-09	0.000	15329.5	0.06	3.155E-06	0.000	3022.3

waves accurately while inaccurately treating acoustic waves and fast gravity waves [220, 85, 180]. This is because the fast modes have a negligible effect on the nonlinear convected motion for low Mach number flows (e.g., compressible fluid dynamics contain acoustic waves, which play no essential role on the motion of interest in the atmosphere or the ocean).

In this section, gravity waves are the fast modes in the shallow water systems, and we examine how IMEX HDG-DG methods suppress the gravity waves.

We consider the propagation of smooth gravity waves [65] over the domain

$\Omega = [-1, 1] \times [-1, 1]$. The initial condition is given as

$$H = H_\infty + e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad \text{and} \quad u = v = 0,$$

where $H_\infty = 100$. We set the gravitational acceleration g to be unity. The domain is discretized with 20×20 finite elements and with 8th order polynomials. The time horizon is $T = 0.5$.

We choose different time-step sizes for RK2 DG and ARS2 HDG-DG. Since RK2 DG blows up after a few iterations with $\Delta t = 0.0002$ (see Figure 3.3), we take $\Delta t = 0.0001$. The time-step sizes of $\Delta t = 0.002$ and $\Delta t = 0.02$ are chosen for ARS2 HDG-DG. Figure 3.4 quantitatively shows a three-dimensional plot of the evolution of the free surface elevation using RK2 DG and ARS2 HDG-DG at times $t = 0$, $t = 0.06$, $t = 0.1$, and $t = 0.5$. Since the speed of the reference gravity wave is $\sqrt{gH_\infty} = 10$, the wave crest propagates radially and reaches to the wall boundary at $t = 0.1$. We observe that ARS2 HDG-DG with $\Delta t = 0.002$ is in good agreement with RK2 DG. While IMEX-RK methods allow us to increase the time-step size without being penalized by stability constraints, the accuracy of the fast modes is reduced due to the truncation error in the time discretization. This can be observed in the last column of Figure 3.4 in which ARS2 HDG-DG with $\Delta t = 0.02$ shows damped (inaccurate) but stable solutions.

In Figure 3.5, we compare the wall clock times and the errors of ARS2 HDG-DG and ARS3 HDG-DG with the DG counterparts. Since no analytical solution is available in this case, we take the numerical solution of 10th order RK4 DG with $\Delta t = 0.00005$ as a reference solution and measure an L_2 norm. The norm is defined as $L_2(q, q_R) := (\sum_{K \in \Omega_h} \int_K (q - \Pi q_R)^2 dK)^{\frac{1}{2}}$, where Π is the elementwise projection operator from 10th to 8th order polynomial space. As can be seen, ARS2 HDG-DG and ARS3 HDG-DG show correct second-order and third-order convergence in time

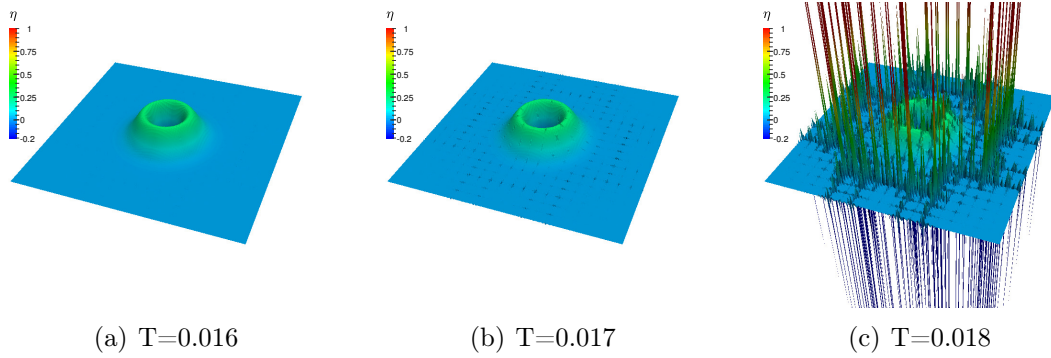


Figure 3.3: Water height perturbation test: free surface evolution for RK2 DG ($\Delta t=0.0002$) at $t = 0.016$, $t = 0.017$ and $t = 0.018$.

for decreasing time-step size. For increasing time-step size, we see that the error level is bounded above. This is because the fast gravity waves are effectively damped out as shown in the last column in Figure 3.4. This implies that we can view the IMEX methods as a high-frequency filter.

From an efficiency standpoint, in Figure 3.2(b), ARS2 HDG-DG and ARS3 HDG-DG do not compete with RK2 DG and RK3 DG at the same error level. However, high-order ARS3 HDG-DG is more economical than low-order RK2 DG below the error level of $\mathcal{O}(10^{-4})$. Also, in view of a high-frequency filter, ARS2 HDG-DG and ARS3 HDG-DG methods are attractive because they provide fast and stable solutions. For example, at $\mathcal{O}(10^{-2})$ error level, they are two times faster than RK2 DG, and at $\mathcal{O}(10^{-1})$ error level, they are ten times faster than RK2 DG.

3.6.3 Steady-state geostrophic flow

We consider the steady-state geostrophic flow in [225]. This flow admits an analytical solution for the shallow water equations on the sphere [152]. The initial

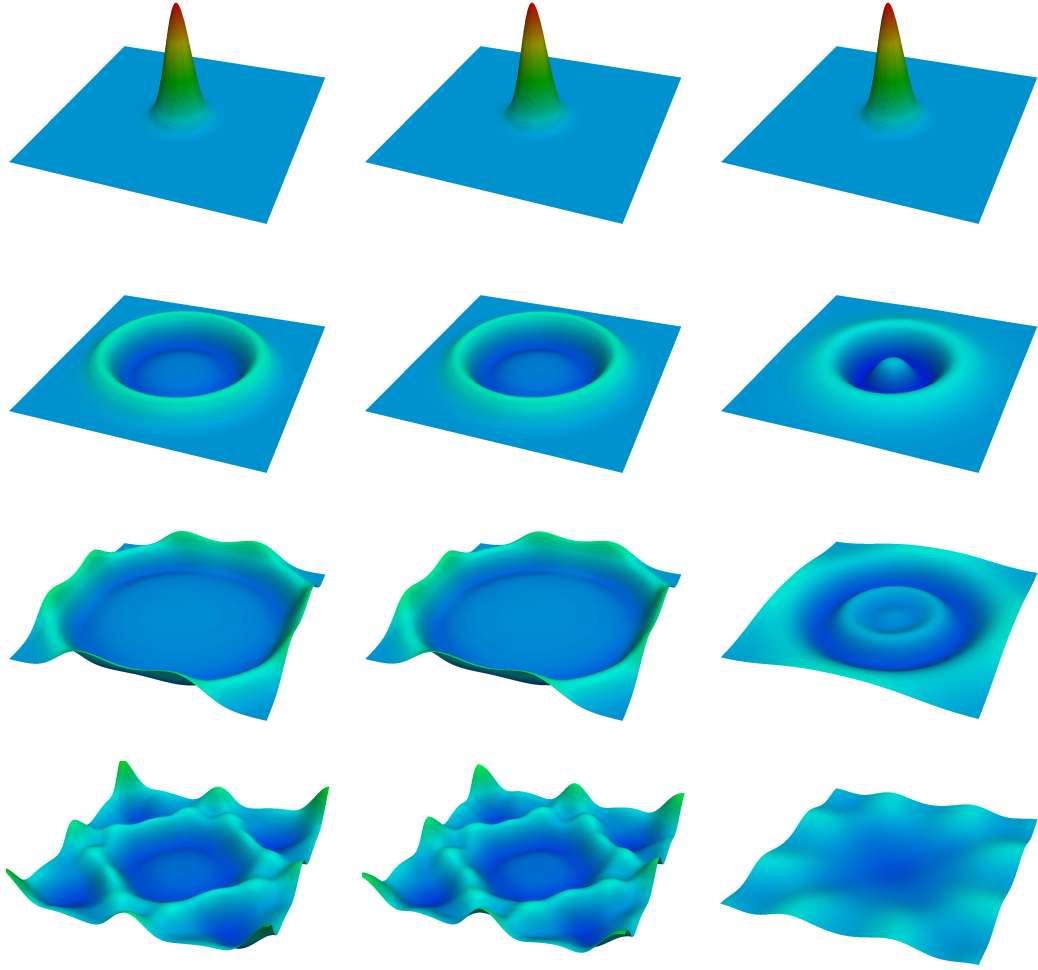


Figure 3.4: Water height perturbation test: free surface evolution for RK2 DG ($\Delta t=0.0001$, left), ARS2 HDG-DG ($\Delta t=0.002$, center) and ARS2 HDG-DG ($\Delta t=0.02$, right) at times $t = 0$, $t = 0.06$, $t = 0.1$ and $t = 0.5$.

condition is given by

$$H = H_{\infty} - \frac{1}{g} \left(a\Omega + \frac{u_{\infty}^2}{2} \right) \cos^2 \theta, \quad (3.41a)$$

$$(u_{\lambda}, u_{\theta}) = (u_{\infty} \cos \theta, 0), \quad (3.41b)$$

where $(u_{\lambda}, u_{\theta})$ is the local tangential velocity in latitude-longitude coordinates (λ, θ) . We take $gH_{\infty} = 2.94 \times 10^4 \text{ m}^2 \text{ s}^{-2}$, and $u_{\infty} = 38.61 \text{ m s}^{-1}$, which corresponds to

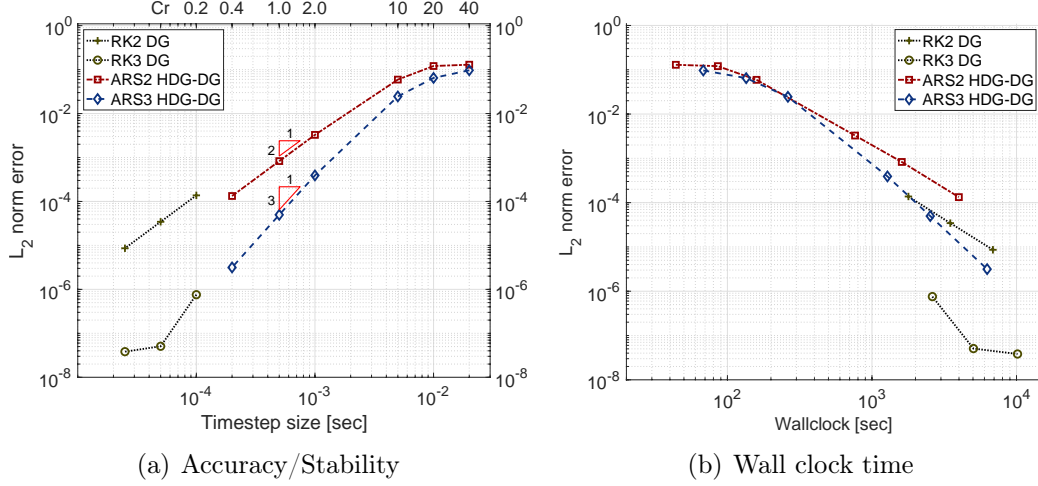


Figure 3.5: Comparison between IMEX HDG-DG and RKDG for the smooth gravity wave: (a) the accuracy/stability and (b) the wall clock time.

$Fr = 0.23$. The numerical experiment is performed on a grid with $N_e = 1536$ elements (16×16 elements on each of the six faces of the cubed-sphere) and solution order $p = 4$. The time-step size for ARS2 HDG-DG is 864 s. Figure 3.6 shows the snapshot of the height field from the ARS2 HDG-DG approach (Figure 3.6(a)) and the exact field (Figure 3.6(b)) after 12 days. The height field from ARS2 HDG-DG is almost the same as the exact solution. Indeed, we show in Figure 3.6(c) the relative error of the height field, $|\frac{H_{num} - H_{exact}}{H_{exact}}|$, and the maximum relative error of $\mathcal{O}(10^{-7})$ is observed (see also Figure 3.7(a)).

Figure 3.7 shows the time series of the height field error, mass, and energy loss in L_1 , L_2 and L_∞ norms, where L_1 and L_∞ norms are defined as $L_1(q) := \sum_{K \in \Omega_h} \int_K |q - q_T| dK$ and $L_\infty(q) := \max_{q \in \Omega_h} |q - q_T|$, respectively.

Here, the mass and energy losses are defined as

$$\text{mass loss} = \left| \frac{\text{mass}(t) - \text{mass}(0)}{\text{mass}(0)} \right|, \quad \text{energy loss} = \left| \frac{\text{energy}(t) - \text{energy}(0)}{\text{energy}(0)} \right|,$$

where $\text{mass} := \|H\|_{\Omega_h}^2$ and $\text{energy} := \|H\mathbf{u} \cdot \mathbf{u} + gH^2\|_{\Omega_h}^2$. We observe that the mass has $\mathcal{O}(10^{-14})$ loss level, whereas the energy has $\mathcal{O}(10^{-13})$ loss level. This is a

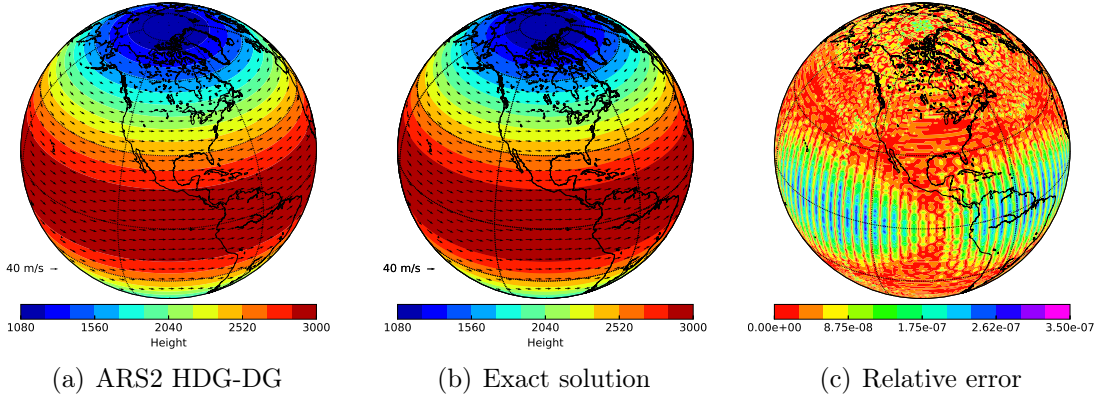


Figure 3.6: Geostrophic flow test case: (a) total height field from ARS2 HDG-DG at day 12 with $Cr = 1.36$, (b) the exact solution, and (c) the relative error of the height field.

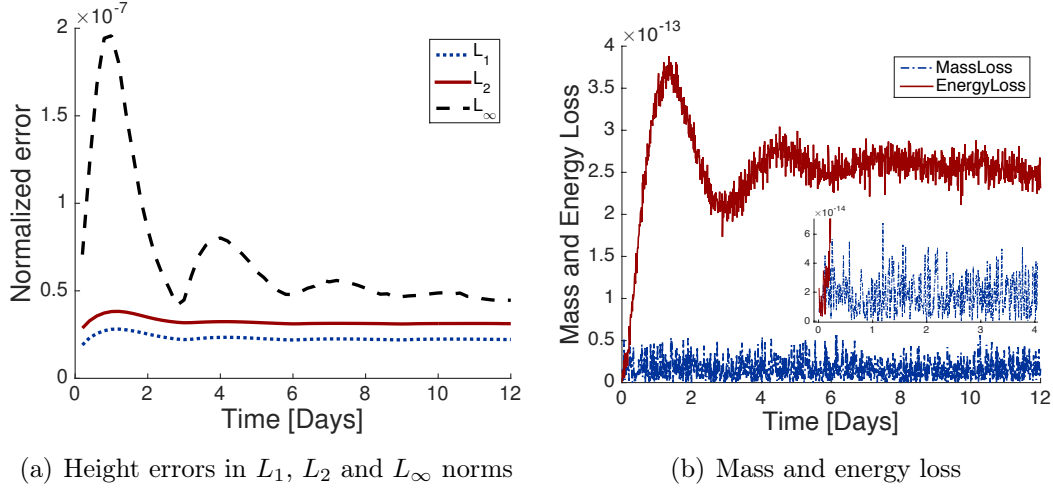


Figure 3.7: Geostrophic flow test case: (a) time evolution of height field error in L_1 , L_2 and L_∞ norms, and (b) mass and energy loss of the ARS2 HDG-DG solution with timestep size of 864 s (i.e. $Cr = 1.36$).

direct consequence of the fact that both DG and HDG are conservative discretizations for the mass.

3.6.4 Steady-state geostrophic flow with compact support

This case is similar to the steady-state geostrophic flow in Section 3.6.3. The difference is that it is equipped with a compactly supported wind field, considered as a high latitude jet in the northern hemisphere. The initial condition is given as

$$H = H_\infty - \frac{a}{g} \int_{-\pi/2}^{\theta} \left(f + \frac{u(\tau) \tan \tau}{a} \right) u(\tau) d\tau, \quad (3.42a)$$

$$(u_\lambda, u_\theta) = (u_\infty b(x) b(x_e - x) e^{4/x_e}, 0), \quad (3.42b)$$

where $\theta_b = -\pi/6$, $\theta_e = \pi/2$, $x_e = 0.3$, $x = x_e \frac{\theta - \theta_b}{\theta_e - \theta_b}$, and $b(x) = \begin{cases} 0 & \text{for } x \leq 0, \\ e^{-1/x} & \text{for } 0 < x. \end{cases}$

For the spatial convergence test, we conduct both h -convergence and p -convergence studies in Figure 3.8.

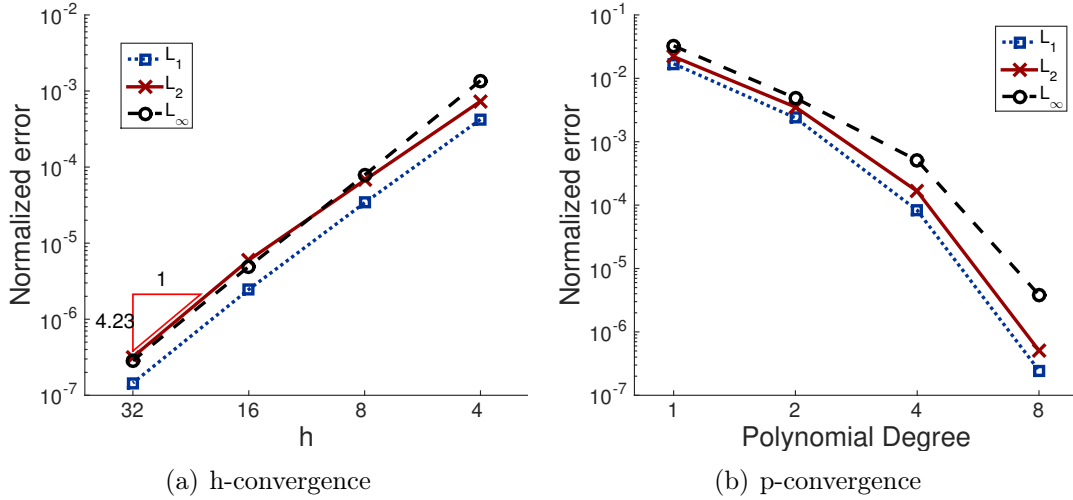


Figure 3.8: Convergence studies for the ARS2 HDG-DG scheme when applied to the steady-state geostrophic flow with compact support with $Cr = 0.7$: (a) h -convergence with $p = 3$ and (b) p -convergence with $N_e = 6144$.

The errors are measured at $T = 9.6$ hour with $Cr = 0.7$. For h -convergence, the height field error in L_1 , L_2 and L_∞ norms are computed for $p = 3$ and the total number of elements is given by $N_e = 6n^2$, where $n = \{4, 8, 16, 32\}$. As can be seen in

Figure 3.8(a), the convergence rate is $p + 1$. For p -convergence, an exponential rate is observed in Figure 3.8(b).

3.6.5 Zonal flow over an isolated mountain

We consider the zonal flow over an isolated mountain test proposed in [225]. The height and wind fields are similar to those of the steady-state geostrophic flow, but now $H_\infty = 5960$ m and $u_\infty = 20$ m s⁻¹ with $Fr = 0.08$. A mountain with height $H_s = 2000(1 - r/r_s)$ m, located at $(\lambda_c, \theta_c) = (3\pi/2, \pi/6)$, is introduced into the flow, where $r_s = \frac{\pi}{9}$ and $r^2 = \min(r^2, (\lambda - \lambda_c)^2 + (\theta - \theta_c)^2)$.

We plot the height field at days 5, 10 and 15 in Figure 3.9 on a grid with $N_e = 384$ elements ($8 \times 8 \times 6$ elements on the cubed-sphere) and solution order $p = 8$. The timestep size of 432 s is taken. As can be seen, the height fields are smooth and comparable to the corresponding results in [152, 207] (note that this problem has no analytic solution).

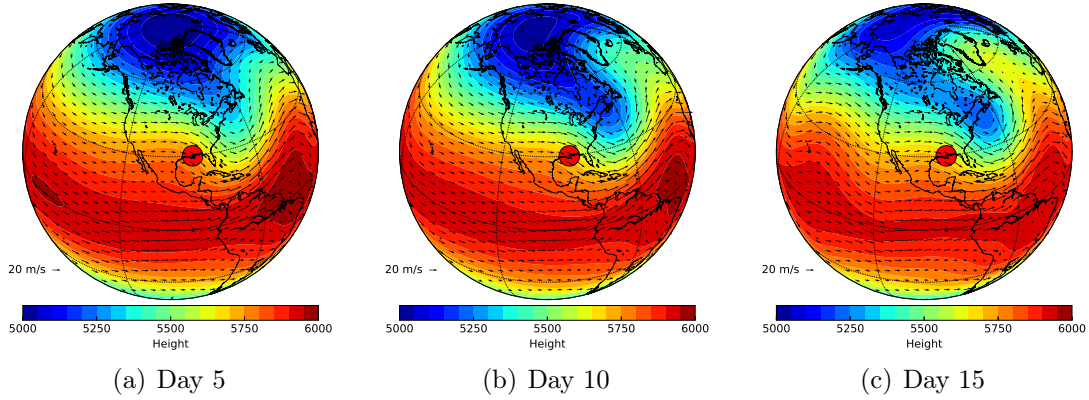


Figure 3.9: Flow over an isolated mountain (red circle) computed with the ARS2 HDG-DG scheme: shown are the total water height after (a) 5 days, (b) 10 days, and (c) 15 days. The numerical experiments are performed on a grid with $N_e = 384$ elements, solution order $p = 8$, and $Cr = 1.46$. Contour levels are from 5000 m to 6000 m with 21 levels.

We compare the height field of ARS2 HDG-DG with that of RK2 DG in Figure

3.10. We take $\Delta t = 43.2$ s ($\text{Cr}=0.15$) for RK2 DG. The height field of ARS2 HDG-DG is in good agreement with that of RK2 DG: the relative difference in the height field is $\mathcal{O}(10^{-3})$.

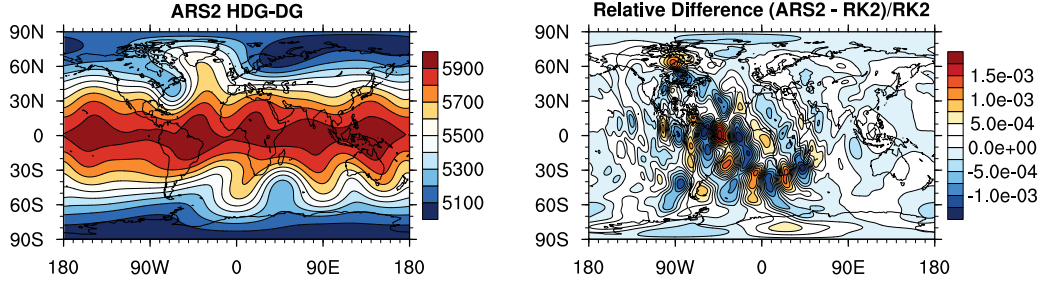


Figure 3.10: The zonal flow over an isolated mountain: total height field computed from ARS2 HDG-DG (left) at day 15 with Cr of 1.46; the relative difference (right) with RK2 in the height field.

3.6.6 Rossby-Haurwitz wave

We next consider the Rossby-Haurwitz wave test in [225]. The Rossby-Haurwitz wave is an exact solution of the nonlinear barotropic vorticity equation [93], but not an exact solution of the shallow water system [152]. The wave number is chosen to be 4. This wave pattern propagates from the west to the east keeping its shape at an angular velocity of 12° per day. With the maximum speed 100 m s^{-1} and the minimum depth $8 \times 10^3 \text{ m}$, we have the Froude number of 0.35.

To simulate this test case, we use the ARS2 HDG-DG scheme on a grid with $N_e = 864$ elements ($12 \times 12 \times 6$), solution order $p = 5$, and with a time-step size of 345.6 s (i.e. $\text{Cr} = 1.2$). The height fields at days 0, 7 and 14 are shown in Figure 3.11. We also compare the results of ARS2 HDG-DG with those of RK2 DG in Figure 3.12 after 14 days. For RK2 DG, we take the timestep size to be 43.2 s ($\text{Cr}=0.14$) for stability. The height field of ARS2 HDG-DG is in good agreement with that of RK2 DG. In particular, the relative difference in the height field is $\mathcal{O}(10^{-3})$.

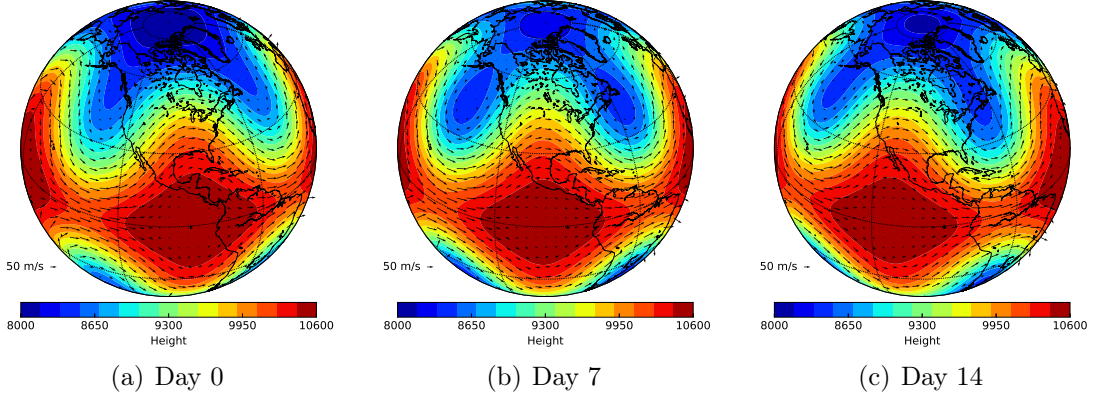


Figure 3.11: Rossby-Haurwitz wave: the total height field computed from ARS2 HDG-DG after a) 0 days, (b) 7 days, and (c) 14 days. The numerical experiment is performed on the grid with $N_e = 864$ elements, solution order $p = 5$, and $Cr = 1.2$. Contour levels are from 8000 m to 10 600 m with the step size of 173 m.

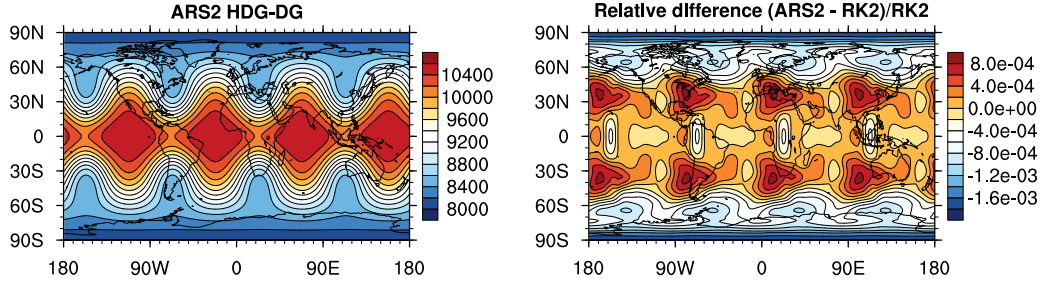


Figure 3.12: Rossby-Haurwitz wave: the total height field computed from ARS2 HDG-DG (left) after 14 days with Cr of 1.2; the relative difference (right) with RK2 in the height field.

3.6.7 Barotropic instability test

In this section, we consider the barotropic instability test in [77]. A zonal jet, a wind field along a latitude line and geostrophically balanced height, is initialized in the northern hemisphere. Then, the height field is perturbed by adding a smoothly localized bump to the center of the jet, which causes barotropic waves to evolve in time. The maximum velocity of the jet is 80 m s^{-1} and $H_\infty = 1 \times 10^4 \text{ m}$ is chosen, which corresponds to $Fr = 0.26$.

Figure 3.13 shows the relative vorticity field of the barotropically unstable flow at days 4, 5 and 6. The numerical experiment is conducted on the grid with $N_e = 5400$ elements ($30 \times 30 \times 6$), solution order $p = 4$, and time-step size of 173 seconds. The vorticity field computed from ARS2 HDG-DG is comparable to that of [138], which use high-order continuous and discontinuous Galerkin methods with explicit time-integration.

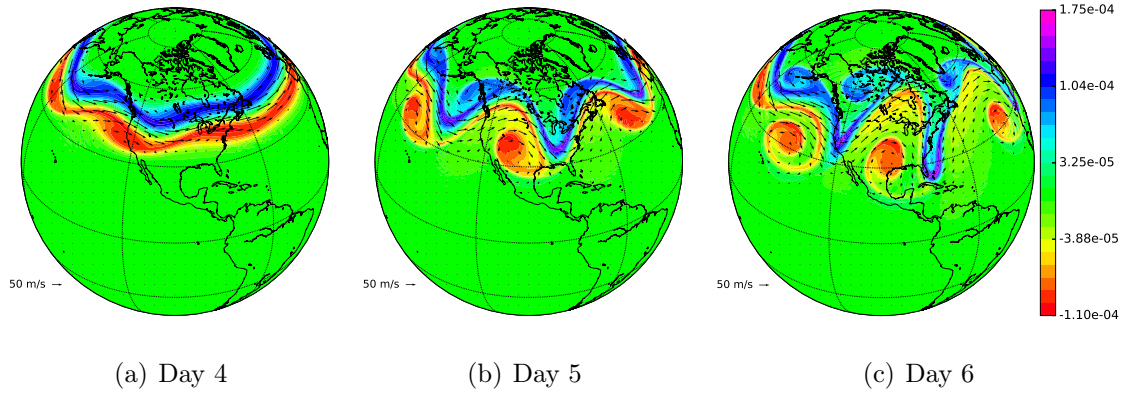


Figure 3.13: Barotropic instability test: relative vorticity field of the ARS2 HDG-DG at (a) 4 days, (b) 5 days, and (c) 6 days. The numerical experiment was performed on the grid of $N_e = 5400$ and $p = 4$ with a time-step size of 173 s ($Cr=0.94$). The vorticity ranges from $-1.1 \times 10^{-4} \text{ s}^{-1}$ to $1.8 \times 10^{-4} \text{ s}^{-1}$.

We also compare the results of ARS2 HDG-DG with those of RK2 DG in Figure 3.14 after 6 days. For RK2 DG, we take the timestep size to be 21.6 s (i.e $Cr=0.12$) for stability. The vorticity field of ARS2 HDG-DG is in good agreement with that of RK2 DG. Indeed, the difference in the vorticity field is $O(10^{-6})$.

3.6.8 Inertia-gravity waves

As a test for the non-hydrostatic equations, we consider the propagation of the inertia-gravity waves [192, 86]. We consider the uniformly stratified atmosphere with a

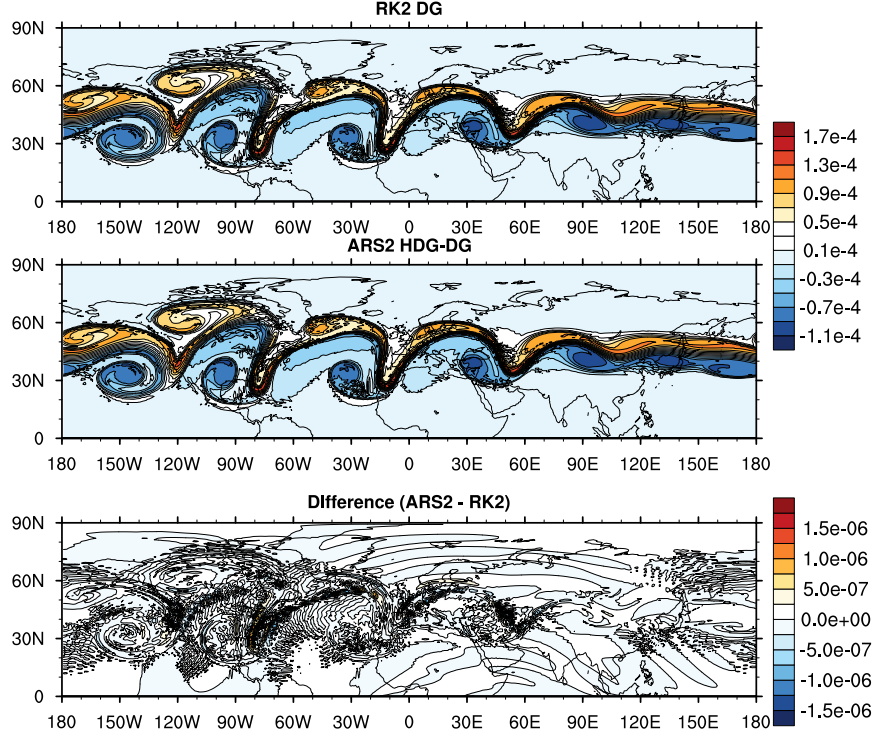


Figure 3.14: Barotropic instability test: relative vorticity fields of RK2 DG (top) and ARS2 HDG-DG (middle) at 6 days with Cr of 0.94 and 0.12, respectively; the difference (bottom) in the vorticity field between ARS2 HDG-DG and RK2 DG.

constant Brunt-Väisälä frequency ⁴ of $\mathcal{N} = 10^{-2}\text{s}^{-1}$. We construct the hydrostatically balanced potential temperature profile, i.e.,

$$\bar{\theta}(z) = \bar{\theta}_o \exp\left(\frac{\mathcal{N}^2 z}{g}\right)$$

⁴ Brunt-Väisälä frequency is the frequency of vertical acceleration on an air parcel. For an ideal gas, the frequency can be written in terms of the potential temperature as $\mathcal{N}^2 = \frac{g}{\bar{\theta}} \frac{\partial \bar{\theta}}{\partial z}$. For an isothermal atmosphere \bar{T} , the frequency becomes a constant, i.e., $\mathcal{N} = g (C_p \bar{T})^{-\frac{1}{2}}$ with C_p specific heat at constant pressure. With the hydrostatic balanced equation and the equation of state, we denote the density and the pressure by

$$\bar{p} = \bar{p}_o \exp\left(-\frac{gz}{R\bar{T}}\right) \text{ and } \bar{\rho} = \bar{\rho}_o \exp\left(-\frac{gz}{R\bar{T}}\right)$$

with $\bar{p}_o = R\bar{T}\bar{\rho}_o$. The potential temperature is denoted as $\bar{\theta} = \bar{T}\Pi^{-1}$ with the Exner pressure. Here, the Exner pressure is defined as $\Pi = \left(\frac{\bar{p}}{\bar{p}_o}\right)^{1-1/\gamma} = \left(\frac{\bar{\rho}}{\bar{\rho}_o}\right)^{\gamma-1}$.

with $\bar{\theta}_o = 300$ K. Then we add an initial perturbation of the potential temperature

$$\theta'(x, z) = \theta_c \left(\frac{\sin\left(\frac{\pi z}{z_l}\right)}{1 + \left(\frac{x-x_c}{\sigma}\right)^2} \right)$$

on top of the hydrostatically balanced atmosphere. Here, we choose $\theta_c = 0.01$ K, $x_c = 100$ km, $\sigma = 5$ km and $z_c = 10$ km. When air parcel moves up slightly from its rest position, the displaced air parcel returns to its original position by the gravity. When it reaches the rest position, it moves downwards because of negative vertical velocity. However, at some point, due to the pressure force, the air parcel stops and return to its original position. This creates a wave. It is described by inertia and gravity. As you can see in Figure 3.15, the inertia-gravity wave propagates in a channel $(x, z) \in [0, 300]\text{km} \times [0, 10]\text{km}$ for $t \in [0, 3000]\text{s}$. Since a uniform velocity $u_0 = 20 \text{ m s}^{-1}$ is enforced, the center of the inertia-gravity wave moves to the right. In Figure 3.16, ARS2 with $dt = 10$ s solution matches well with RK4 with $dt = 0.1$ s solution. The sampled plot of the potential temperature perturbation at the height of 5 km reveals that both the RK4 with $dt = 0.1$ s and ARS2 with $dt = 10$ s results agree well.

3.6.9 Rising thermal bubble

The rising thermal bubble test [86] mimics the convection system of the atmosphere. A warm bubble is located in a constant potential temperature environment. Because the bubble is warmer than the ambient temperature, the air bubble rises up and the shear motion of the velocity field transform the circular shape into a mushroom shape.

We take a constant potential temperature, $\bar{\theta} = 300$ K as a background field,

and add a small perturbation of the form

$$\theta' = \begin{cases} 0 & \text{for } r > r_c, \\ \frac{1}{2}\theta_c (1 + \cos(\pi r r_c^{-1})) & \text{for } r \leq r_c \end{cases}$$

with $\theta_c = 0.5$ K, $r = \|\mathbf{x} - \mathbf{x}_c\|_2$, $r_c = 250$ m, $\mathbf{x}_c = (500, 350)$ m. The simulation runs for $t \in [0, 900]$ s on $(x, z) = [0, 1]^2$ km, which is discretized with $(N_{ex}, N_{ez}) = (10, 10)$ elements and solution order $k = 10$. The snapshots of the perturbation of the potential temperatures are plotted in Figure 3.18 for ARS2 with $Cr = 52.62$, where the contour levels range from 0 K to 0.5 K with 11 isosurfaces. Compared to RK4 with $Cr = 0.5262$, ARS2 with $Cr = 52.62$ can increase one hundred times larger timestep size, but produce a comparable result as shown in Figure 3.19.

3.6.10 Density current

The density current test was proposed by [196]. This example is similar to the rising thermal bubble test except that the cold bubble falls into the viscous fluid. The perturbation of the potential temperature is given by

$$\theta' = \begin{cases} 0 & \text{for } r > 1, \\ \frac{1}{2}\theta_c (1 + \cos(\pi r)) & \text{for } r \leq 1 \end{cases}$$

with $\theta_c = -15$ K and $r = \left\| \frac{\mathbf{x} - \mathbf{x}_c}{\mathbf{x}_r} \right\|_2$. The center of the bubble is initially located at $\mathbf{x}_c = (0, 3)$ km and its size is controlled by $\mathbf{x}_r = (4, 2)$ km. The computational domain is defined as $\mathbf{x} \in [0, 25.6] \text{ km} \times [0, 6.4] \text{ km}$ and is discretized with $(N_{ex}, N_{ez}) = (32, 8)$ elements and $k = 8$ solution order. A uniform viscosity $75 \text{ m}^2 \text{ s}^{-1}$ is applied to the whole domain. A slip boundary condition is imposed on all the directions. The cold bubble drops, hit the bottom, and propagates to the right. Because of the strong viscosity, the cold front creates a shear force at the upper boundary of the front. This results in Kelvin-Helmholtz rotors. The evolution of the cold bubble is shown in Figure 3.20 for ARS2 with $Cr = 1.7$. We compare the result of ARS2 to that of

RK4 in Figure 3.21. Both the results seem to match well and 3.22 also shows that the sampled profiles of ARS2 and RK4 at $z = 1.2$ km are on top of each other. In this example, ARS2 HDG-DG method uses $dt = 0.2$ s whereas RK4 DG does $dt = 0.04$ s. The timestep size of ARS2 HDG-DG is only five times larger than that of RK4 DG. This is because we treat the artificial viscosity term explicitly. We consider the implicit treatment of the viscosity terms in the future study.

3.7 Discussions

In this chapter, we are interested in subcritical shallow water and subsonic Euler systems. We start by decomposing the original flux into a linear part (obtained from linearizing the flux at the lake at rest condition) containing the fast wave and a nonlinear part for which the fastest wave is removed. We spatially discretize the former using an HDG method, and the latter using a DG approach. This enables us to develop an IMEX HDG-DG framework in which we integrate the DG discretization explicitly and the HDG discretization implicitly.

The purpose of our coupled approach is fivefold: (i) to step over the fast waves using larger time step sizes (compared to fully explicit methods) without facing instability; (ii) to avoid expensive Newton-type iterations (compared to fully implicit methods) for each time step; (iii) to take advantage of the DOF reduction in HDG method (relative to DG approaches) to further reduce the cost of linear solves; (iv) to preserve high-order accuracy in both space and time; and (v) to be applicable for applications where slow modes are accurately treated while permitting less accurate handling fast modes.

Numerical results have shown that while fully explicit DG approaches are stable with small timestep sizes, our IMEX HDG-DG method is stable for orders of

magnitude larger time-step sizes. In the IMEX HDG-DG framework, the resulting HDG system is linear. That means only the right-hand side of the HDG system needs to be updated throughout the computation, which significantly reduces the computational cost. We have shown that dense but smaller linear system can be constructed with HDG methods in discontinuous Galerkin discretization. We also numerically observe that the proposed approach achieves the expected high-order accuracy both in time and space. Especially, the high-order IMEX HDG-DG methods can be beneficial compared to the low-order explicit RK method with a fixed error level. Not only that, with a low Froude number, the IMEX HDG-DG methods can be more economical than the explicit RK when the desired accuracy is relaxed. We also numerically confirm that the IMEX HDG-DG methods damp out the fast gravity waves with large time-step size. This implies that IMEX HDG-DG method can be used as a tool to suppress the high-frequency modes such as acoustic waves or fast gravity waves shown in atmospheric or ocean models.

These observations make IMEX HDG-DG methods attractive for applications in which slow modes are accurately treated while permitting less accurate handling fast modes.

Note that our proposed methods work for subcritical flows, which means the speed of gravity waves is faster than the speed of nonlinear advection. Considering the speed of gravity wave is proportional to the water depth, i.e. $c_g \sim \sqrt{gH}$, our IMEX scheme is not appropriate for simulations requiring the wetting and drying treatment where the water depth is shallow. Similarly, our IMEX method works for subsonic flows in Euler systems.

Ongoing work focuses on further improving the efficiency of the IMEX HDG-DG approach by developing preconditioned iterative methods for the linear solve and

to implement the scheme on parallel computing systems. Of interest will be the rigorous convergence analysis of the IMEX-HDG-DG scheme.

3.8 Auxilliary results

3.8.1 Hybridized upwind HDG flux with a one-trace variable

The hybridization is based on a numerical flux form. For example, the numerical flux in (3.28) is composed of the three-trace variables, i.e., $\hat{\phi}_\eta$, \hat{U} and \hat{V} . The resulting HDG system in (3.39) has $2n(n+1)(p+1) \times 3$ degrees of freedom.

To reduce the degrees of freedom further, we can use other forms of numerical flux. For example, the upwind (UP) HDG flux [33] is comprised of a one-trace variable, $\hat{\phi}_\eta$,

$$\mathbf{n} \cdot \hat{\mathcal{F}}_L(\mathbf{q}, \hat{\mathbf{q}}) = \begin{pmatrix} n_x U + n_y V + \sqrt{\phi_B} (\phi_\eta - \hat{\phi}_\eta) \\ n_x \phi_B \hat{\phi}_\eta \\ n_y \phi_B \hat{\phi}_\eta \end{pmatrix} \quad (3.43)$$

The resulting Schur complemented system is smaller, $2n(n+1)(p+1)$, than that with the Lax-Friedrich (LF) flux in (3.28).

As for the moving vortex case, we compare the performance of the two hybridizations (with UP and LF fluxes) in terms of accuracy and efficiency. In Table 3.4, we observe the theoretical second-order and third-order convergence in time for ARS2 HDG-DG and ARS3 HDG-DG with both three-trace variables and one-trace variable. Compared to the ARS HDG-DG scheme with three-trace variables, the one-trace counterpart has almost the same accuracy but is slightly more economical. For this reason, we choose the ARS HDG-DG scheme with the one-trace variable to compare with the explicit RKDG methods for the planar shallow water equations.

Clearly, the linear solver with the three-trace variables is more expensive than that with the one-trace variable. However, in the proposed IMEX HDG-DG method,

we perform LU factorization of the HDG system in (3.39) one time and reuse the same LU factors for the succeeding computations. The difference in wall clock time is not significant for the moving vortex case.

3.8.2 Well-balancing property

We consider the first order IMEX scheme, which is composed of the forward and backward Euler schemes, to the semi-discrete system (3.26) for the planar shallow water equation (3.5),

$$\left(\frac{\mathbf{q}^{n+1} - \mathbf{q}^n}{\Delta t}, \mathbf{v} \right)_{\Omega_h} = \mathcal{NL}(\mathbf{q}^n) + \mathcal{L}(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}), \quad (3.44a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{q}^n, \hat{\mathbf{q}}^n) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.44b)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.44c)$$

for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$.

Our IMEX HDG-DG scheme is well-balanced in the sense that the solution preserves the lake at rest condition. In other words, for a given still-water solution at t^n , $\mathbf{q}^n = (\phi_\eta^n, \mathbf{U}^n) = (0, 0)$, the solution at t^{n+1} becomes zero, $\mathbf{q}^{n+1} = (\phi_\eta^{n+1}, \mathbf{U}^{n+1}) = (0, 0)$. From the conservation condition (3.44b), we notice $\hat{\mathbf{q}}^n = (\hat{\phi}_\eta^n, \hat{\mathbf{U}}^n) = (0, 0)$. At t^n step, the nonlinear term $\mathcal{NL}(\mathbf{q}^n)$ in (3.44a) vanishes because both the flux terms ((3.6) and (3.7)), and the numerical flux terms ((3.27a) and (3.28)) become zero. Thus (3.44a) and (3.44c) yield the linear system as

$$(\mathbf{q}^{n+1}, \mathbf{v})_{\Omega_h} - \Delta t \mathcal{L}(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) = 0, \quad (3.45a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.45b)$$

for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$. This is indeed the HDG system shown in (3.35) with $\mathcal{R}es_0 = 0$, $\mathbf{Q}^{(i)} = \mathbf{q}^{n+1}$ and $\tilde{a}_{ii} = 1$. Since the HDG system has a unique solution, the solution at the next time-step becomes zero, $\mathbf{q}^{n+1} = (0, 0)$.

The IMEX HDG-DG methods is also well-balanced with a constant height perturbation, $\mathbf{q}^n = (C, 0)$. Applying integration by parts to (3.44a) with a zero forcing (i.e., $\mathbf{s} = 0$), we have the nonlinear and linear term as

$$\begin{aligned}\mathcal{NL}(\mathbf{q}^n) &= (\nabla \cdot \mathcal{F}_{NL}(\mathbf{q}^n), \mathbf{v})_{\Omega_h} - \langle (\mathcal{F}_{NL}^*((\mathbf{q}^n)^\pm) - \mathcal{F}_{NL}((\mathbf{q}^n)^\pm)) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h}, \\ \mathcal{L}(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) &= (\nabla \cdot \mathcal{F}_L(\mathbf{q}^{n+1}), \mathbf{v})_{\Omega_h} - \langle \hat{\tau}_L(\mathbf{q}^{n+1} - \hat{\mathbf{q}}^{n+1}), \mathbf{v} \rangle_{\partial\Omega_h}.\end{aligned}$$

Since ϕ_η^n is a constant and $\mathbf{U}^n = 0$, the volume flux term in $\mathcal{NL}(\mathbf{q}^n)$ vanishes because of the divergence operator. From the conservation condition in (3.44b), we have $\hat{\phi}_\eta^n = \{\{\phi_\eta^n\}\} = C$ and $\hat{U}^n = \hat{V}^n = 0$. The surface flux term in $\mathcal{NL}(\mathbf{q}^n)$ also disappears because the penalty terms in the numerical flux in (3.27) become zero. The resulting HDG system yields

$$(\mathbf{q}^{n+1}, \mathbf{v})_{\Omega_h} - \Delta t \mathcal{L}(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) = (\mathbf{q}^n, \mathbf{v})_{\Omega_h}, \quad (3.46a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.46b)$$

for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$. Next, we define $\tilde{\mathbf{q}}^{n+1} := \mathbf{q}^{n+1} - (C, 0)$. From (3.46b), we have

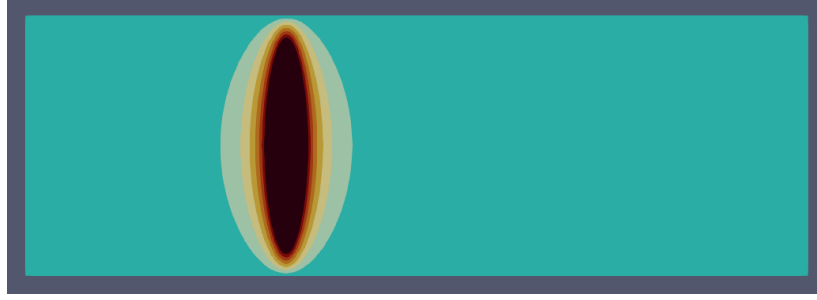
$$\hat{\mathbf{q}}^{n+1} = \hat{\tau}^{-1} \left[\mathbf{n} \cdot \mathcal{F}(\mathbf{q}^{n+1}) \right] + \{\{\mathbf{q}^{n+1}\}\} = \hat{\tau}^{-1} \left[\mathbf{n} \cdot \mathcal{F}(\tilde{\mathbf{q}}^{n+1}) \right] + \{\{\tilde{\mathbf{q}}^{n+1}\}\} + (C, 0). \quad (3.47)$$

By plugging (3.47) to the linear operator $\mathcal{L}(\mathbf{q}^{n+1}, \hat{\mathbf{q}}^{n+1})$, we can rewrite the HDG linear system (3.46) as

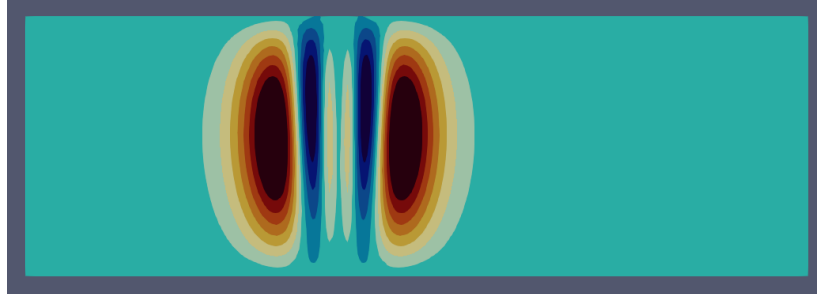
$$(\tilde{\mathbf{q}}^{n+1}, \mathbf{v})_{\Omega_h} - \Delta t \mathcal{L}(\tilde{\mathbf{q}}^{n+1}, \hat{\mathbf{q}}^{n+1}) = 0, \quad (3.48a)$$

$$\left\langle \left[\hat{\mathcal{F}}_L(\tilde{\mathbf{q}}^{n+1}, \hat{\mathbf{q}}^{n+1}) \cdot \mathbf{n} \right], \boldsymbol{\mu} \right\rangle_{\mathcal{E}_h} = 0, \quad (3.48b)$$

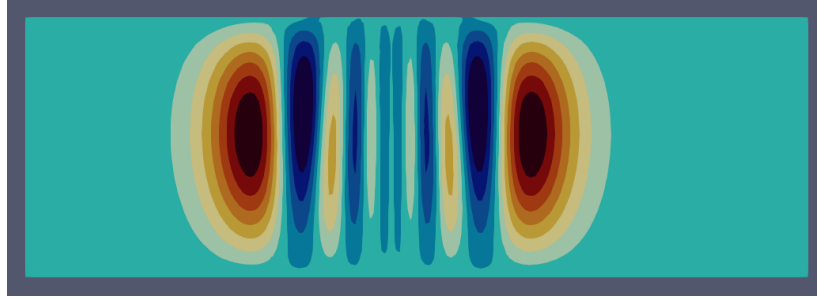
for all $(\mathbf{v}, \boldsymbol{\mu}) \in \mathbf{V}_h(\Omega_h) \times \boldsymbol{\Lambda}_h(\mathcal{E}_h)$. Since the right-hand side is zero and the HDG system has a unique solution, we have $\tilde{\mathbf{q}}^{n+1} = (0, 0)$, which means that $\mathbf{q}^{n+1} = (C, 0)$.



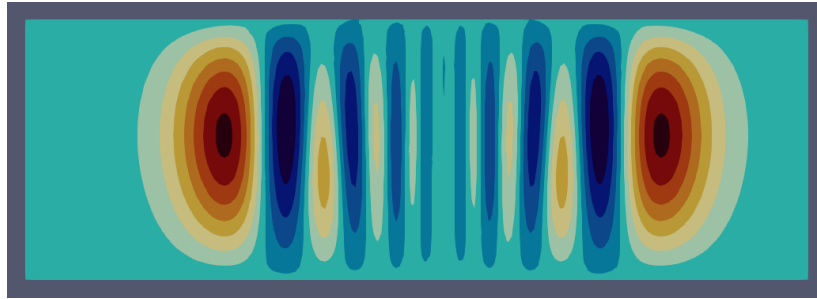
(a) $t = 0$ s



(b) $t = 1000$ s

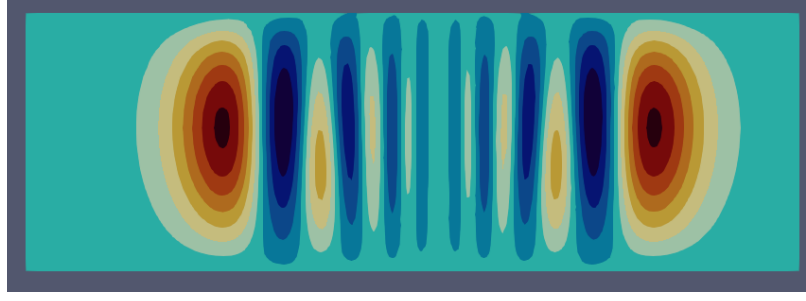


(c) $t = 2000$ s

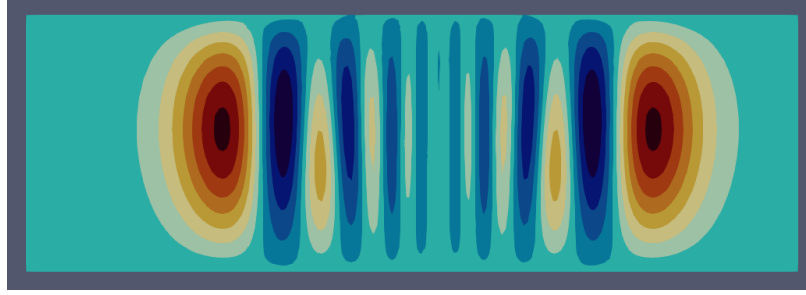


(d) $t = 3000$ s

Figure 3.15: Inertia-gravity wave: the snapshots of the potential temperature perturbations at $T = [0, 1000, 2000, 3000]$ s. ARS2 with $dt = 10$ s solutions are plotted on the grid with $(N_{ex}, N_{ez}) = (30, 2)$ elements and solution order $k = 10$. Contour levels range from -0.0015 K to 0.003 K with 13 isosurfaces.



(a) RK4 ($dt = 0.1$ s)



(b) ARS2 HDG-DG ($dt = 10$ s)

Figure 3.16: Inertia-gravity wave: the potential temperature perturbations at $T = 3000$ s are plotted for RK4 with $dt = 0.1$ s and ARS2 with $dt = 10$ s.

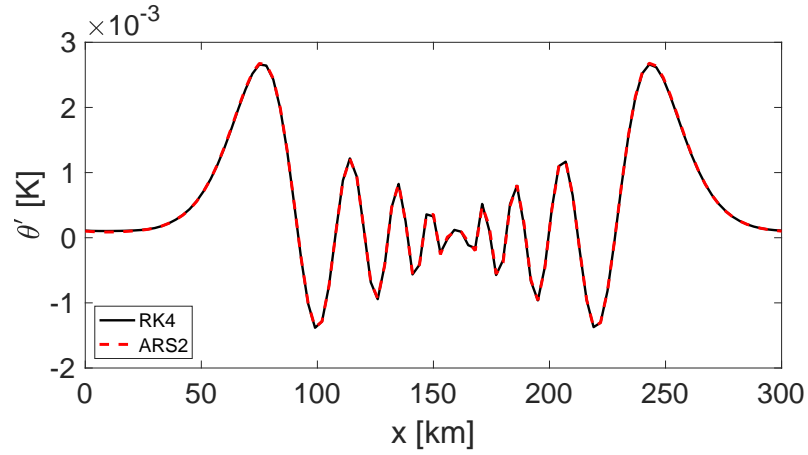


Figure 3.17: The inertia-gravity wave: the potential temperature perturbations are sampled at $z = 5$ km along $x \in [0, 300]$ km at $T = 3000$ s. ARS2 of $dt = 10$ s shows a good agreement with RK4 of $dt = 0.1$ s.

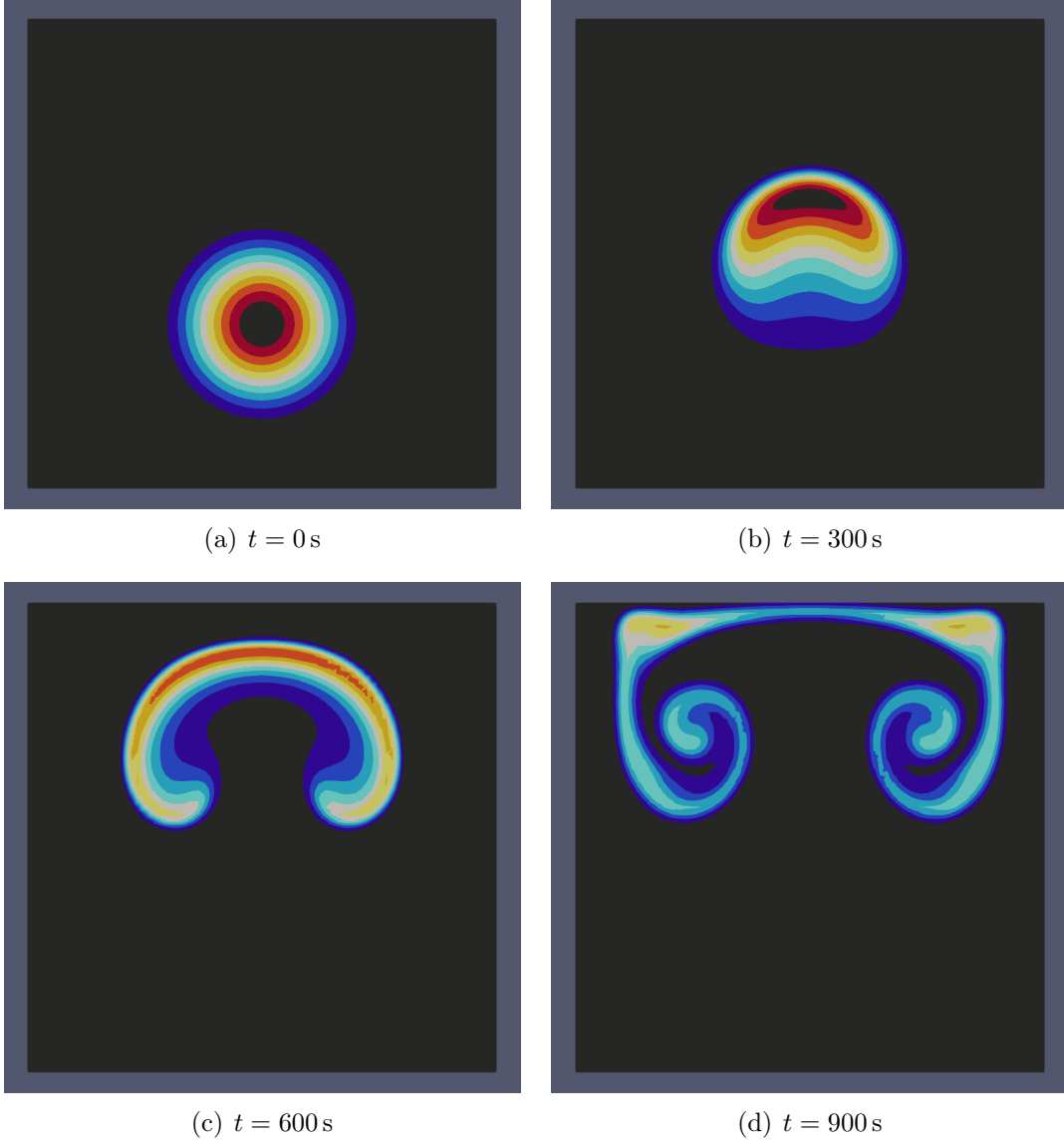
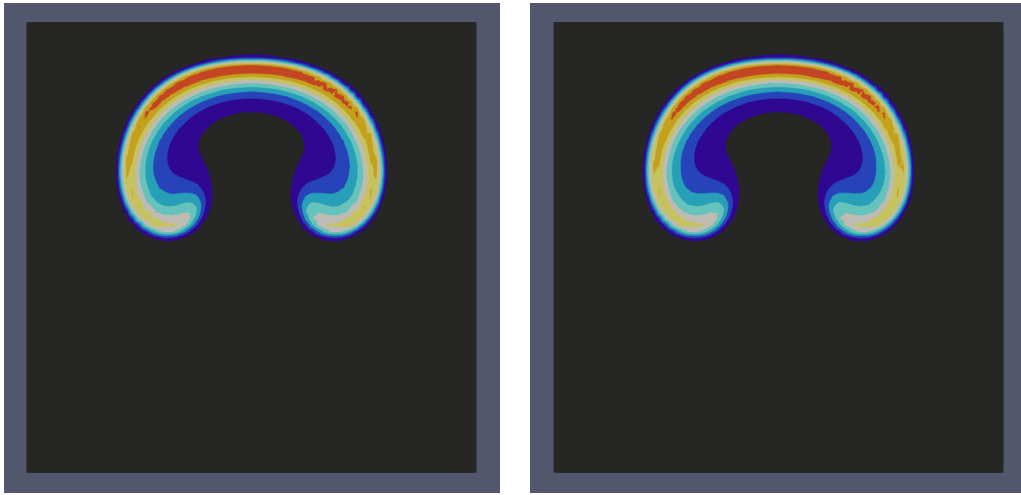


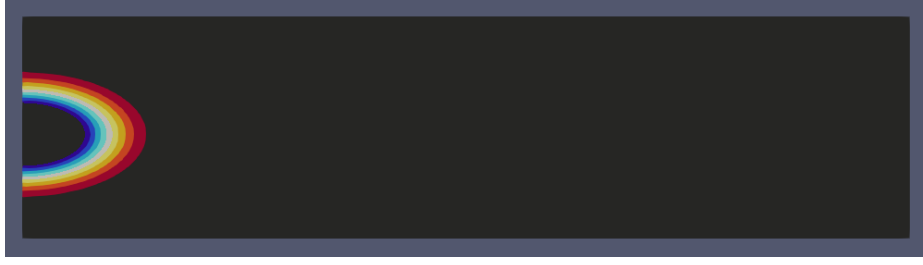
Figure 3.18: Rising thermal bubble: the snapshots of the potential temperature perturbations at $T = [0, 300, 600, 900]$ are plotted for ARS2 HDG-DG with $dt = 0.5(Cr = 52.62)$. The numerical experiment is performed on the grid with $(N_{ex}, N_{ez}) = (10, 10)$ elements and solution order $k = 10$. Contour levels range from 0 K to 0.5 K with 11 isosurfaces.



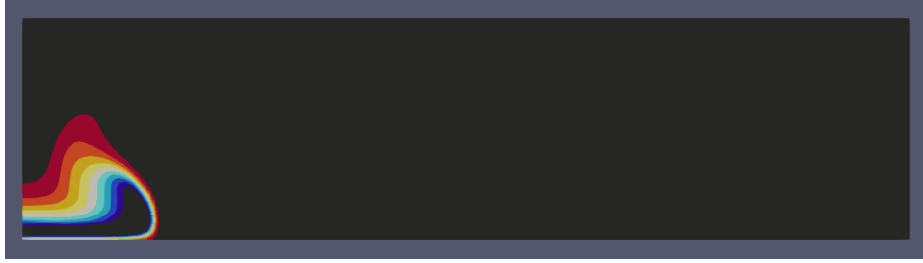
(a) RK4 (CFL=0.5262)

(b) ARS2 (CFL=52.62)

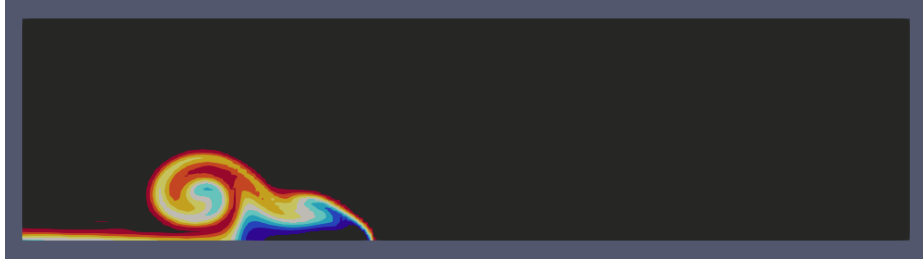
Figure 3.19: Rising thermal bubble: RK4 with $dt = 0.005 \text{ s}$ ($Cr = 0.5262$) and ARS2 with $dt = 0.5 \text{ s}$ ($Cr = 52.62$) solutions at $T = 600 \text{ s}$.



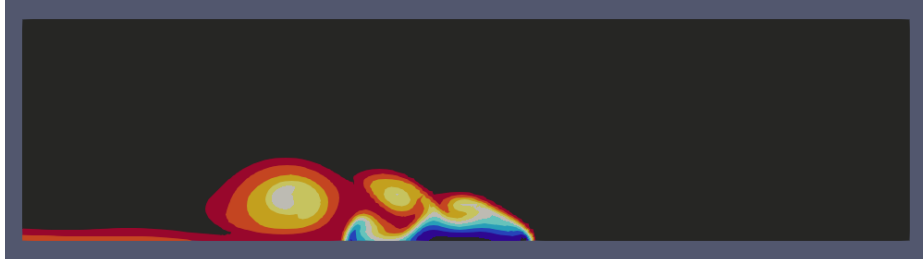
(a) $t = 0 \text{ s}$



(b) $t = 300 \text{ s}$

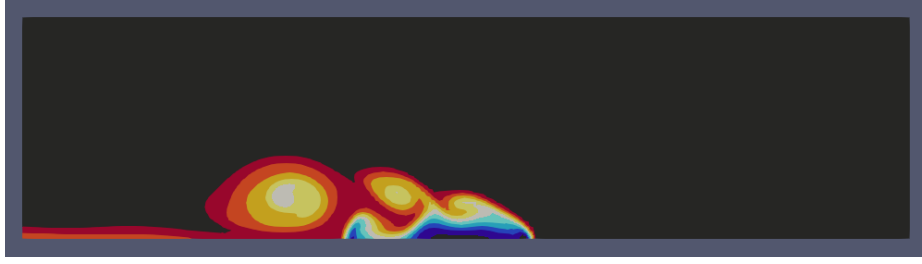


(c) $t = 600 \text{ s}$

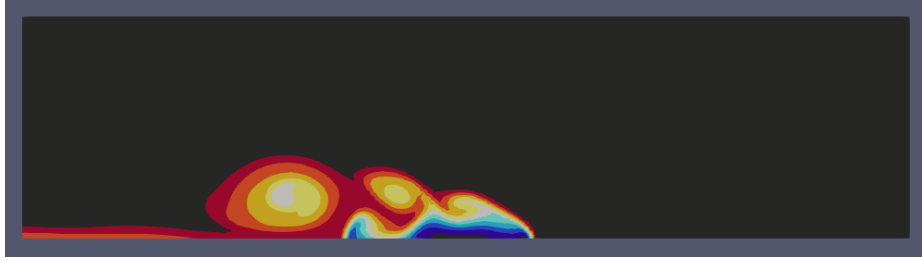


(d) $t = 900 \text{ s}$

Figure 3.20: Density current: the potential temperature perturbations is evolved from $t = 0 \text{ s}$ to $t = 900 \text{ s}$. ARS2 with $dt = 0.2 \text{ s}$ ($Cr = 1.7$) is used for the simulation on the grid with $(N_{ex}, N_{ez}) = (32, 8)$ elements and solution order $k = 8$. Contour levels range from -9.5 K to 0.5 K with 11 isosurfaces.



(a) RK4 ($dt = 0.04$ s)



(b) ARS2 HDG-DG ($dt = 0.2$ s)

Figure 3.21: Density current: RK4 with $dt = 0.04$ s ($Cr = 0.378$) and ARS2 with $dt = 0.2$ s ($Cr = 1.7$) solutions at $t = 900$ s.

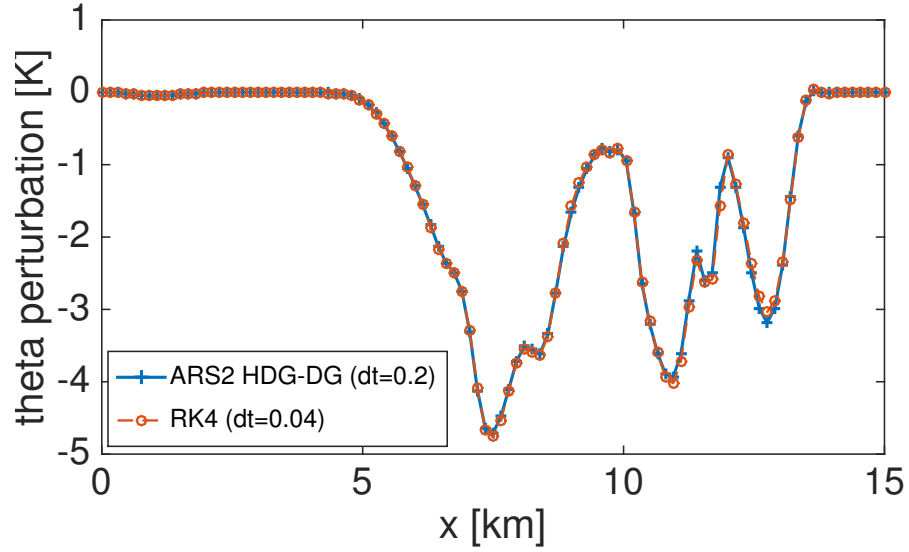


Figure 3.22: Density current: the potential temperature perturbation is sampled at $z = 1.2$ km along $x \in [0, 15]$ km at $T = 900$ s.

Table 3.4: Comparison of the ARS HDG-DG schemes with the three-trace variables $(\hat{\phi}_\eta, \hat{U}, \hat{V})$ in (3.28) and the one-trace variable $(\hat{\phi}_\eta)$ in (3.43) for the moving vortex for $Fr = 0.1$: the errors and the convergence rates for the free surface elevation η and the velocity \mathbf{u} , and wall clock time at $T = 0.2$. The errors are computed over the domain $\Omega \in [-2, 2]^2$ with 32×32 elements and $p = 6$ solution order.

$p = 6, 32 \times 32$	Cr	$\ \eta^e - \eta\ _2$	order	$\ \mathbf{u}^e - \mathbf{u}\ _2$	order	wc [s]
ARS2 $(\hat{\phi}_\eta)$	5.60	2.657E-05	-	1.717E-05	-	98.1
	2.80	6.627E-06	2.003	4.294E-06	2.000	193.5
	1.40	1.655E-06	2.001	1.074E-06	2.000	380.3
	0.70	4.130E-07	2.003	2.684E-07	2.000	759.8
	0.28	6.618E-08	1.998	4.296E-08	2.000	1764.4
ARS3 $(\hat{\phi}_\eta)$	5.60	2.304E-06	-	6.644E-07	-	162.4
	2.80	2.901E-07	2.990	7.602E-08	3.128	319.5
	1.40	3.650E-08	2.990	9.633E-09	2.980	633.8
	0.70	4.598E-09	2.989	1.691E-09	2.510	1220.9
	0.28	3.328E-10	2.866	9.180E-10	0.667	2876.9
ARS2 $(\hat{\phi}_\eta, \hat{U}, \hat{V})$	5.52	2.658E-05	-	1.717E-05	-	112.7
	2.76	6.630E-06	2.003	4.293E-06	2.000	202.9
	1.38	1.656E-06	2.001	1.073E-06	2.000	407.7
	0.67	4.140E-07	2.000	2.683E-07	2.000	758.8
	0.28	6.623E-08	2.000	4.297E-08	1.999	1898.1
ARS3 $(\hat{\phi}_\eta, \hat{U}, \hat{V})$	5.52	2.305E-06	-	6.655E-07	-	192.5
	2.76	2.902E-07	2.990	7.701E-08	3.111	348.7
	1.38	3.648E-08	2.992	1.029E-08	2.904	692.9
	0.67	4.600E-09	2.988	2.603E-09	1.983	1355.0
	0.28	4.331E-10	2.579	2.017E-09	0.278	3189.0

Chapter 4

EXPONENTIAL DG

As we have seen in Chapter 3, the IMEX HDG-DG method requires a linear solver, which means an appropriate preconditioner needs to be equipped for achieving decent performance. However, developing such a preconditioner is not an easy task, especially, for hyperbolic problems. This is why we propose an EXPONENTIAL DG scheme ¹, as an alternative method to the IMEX HDG-DG scheme. Similar to the IMEX HDG-DG scheme, we separate governing equations into linear and nonlinear parts, to which we apply the DG spatial discretization. The former is integrated exactly, whereas the latter is approximated. Since the method does not require any linear solve, it has a potential to represent the phase of the fast modes accurately and to be scalable in a modern massively parallel computing architecture. The proposed EXPONENTIAL DG method: 1) exploits high-order solutions both in time and space; 2) allows a large Courant number ($Cr > 1$); 3) does not require any linear solve; 4) provides promising weak and strong scalable solutions.

We briefly describe the exponential time integrators including an adaptive Krylov subspace approach in Section 4.1. In Section 4.2, we present the construction of the linear operator for Euler equations by introducing a flux Jacobian. The computational performance of the method is discussed in Section 4.3 with several numerical examples.

¹ A manuscript is in preparation for the contents of this chapter. The contributions of the author ranged from numerical implementation of the algorithm and writing the manuscript.

4.1 Exponential time integrations

In this section, we briefly describe the key ideas behind exponential time integration methods. The readers are referred to [101] for more details.

We consider a linear initial value problem,

$$\frac{dq}{dt} = Lq, \quad t \in (0, T), \quad (4.1)$$

where q is a quantity of interest; L is a linear operator and $q(t=0) = q_o$ is the initial condition. The exact solution is given as

$$q(t) = q_o \exp(tL). \quad (4.2)$$

For a nonlinear initial value problem,

$$\frac{dq}{dt} = R(q), \quad t \in (0, T), \quad (4.3)$$

by splitting the right-hand side into linear and nonlinear parts, i.e.,

$$\frac{dq}{dt} = Lq + \mathcal{N}(q), \quad t \in (0, T), \quad (4.4)$$

and multiplying an integrating factor $e^{-\Delta t L}$, we arrive at

$$q(t^{n+1}) = e^{\Delta t L} q(t^n) + \int_0^{\Delta t} e^{(\Delta t - \sigma)L} \mathcal{N}(q(t^n + \sigma)) d\sigma \quad (4.5)$$

with the method of the variation of constants. This formula (4.5) is exact. The first term $e^{\Delta t L} q(t^n)$ is the homogeneous solution, whereas the second term is the particular solution that involves a convolution integral with the matrix exponential. Various exponential integrators are proposed by approximating (4.5) in different ways. For example, the nonlinear term $\mathcal{N}(q(t^n + \sigma))$ can be approximated by Taylor expansions, i.e.,

$$\mathcal{N}(q(t^n + \sigma)) = \sum_{m=0}^{\infty} \left(\frac{\mathcal{N}^{(m)}(q(t^n))}{m!} \right) \sigma^m. \quad (4.6)$$

By substituting (4.6) into (4.5), we have

$$q(t^{n+1}) = e^{\Delta t L} q(t^n) + \sum_{k=1}^{\infty} (\Delta t)^k \varphi_k(\Delta t L) \mathcal{N}^{(k-1)}(q(t^n)) \quad (4.7)$$

with φ_k -functions². When $k = 1$, the first-order exponential time differencing (ETD1) method [54] reads

$$q(t^{n+1}) = e^{\Delta t L} q(t^n) + \Delta t \varphi_1(\Delta t L) \mathcal{N}^{(0)}(q(t^n)). \quad (4.9)$$

When $k = 2$, the second-order ETD2 scheme [54] is obtained,

$$\begin{aligned} q(t^{n+1}) = & e^{\Delta t L} q(t^n) + \Delta t \varphi_1(\Delta t L) \mathcal{N}^{(0)}(q(t^n)) \\ & + \Delta t \varphi_2(\Delta t L) (\mathcal{N}^{(0)}(q(t^n)) - \mathcal{N}^{(0)}(q(t^{n+1}))), \end{aligned} \quad (4.10)$$

where the first-order derivative of $\mathcal{N}^{(1)}$ is approximated by the finite difference method, i.e., $\mathcal{N}^{(1)} \simeq \frac{\mathcal{N}^{(0)}(q(t^n)) - \mathcal{N}^{(0)}(q(t^{n-1}))}{\Delta t}$.

On the other hand, one can use a polynomial to approximate (4.6). The work in [191] shows that the solution of

$$\frac{dq}{dt} = Lq + \sum_{j=0}^{p-1} \frac{t^j}{j!} N_j, \quad t \in (0, T) \quad (4.11)$$

is

$$q(t^{n+1}) = \varphi_0(\Delta t L) q(t^n) + \sum_{j=0}^{p-1} \sum_{l=0}^j \frac{(t^n)^{j-l}}{(j-l)!} (\Delta t)^{l+1} \varphi_{l+1}(\Delta t L) N_j. \quad (4.12)$$

With $v_{j+1} := N_j$, we rearrange (4.12) as

$$q(t^{n+1}) = \varphi_0(\Delta t L) q(t^n) + \sum_{i=1}^p (\Delta t)^i \varphi_i(\Delta t L) \left(\sum_{j=0}^{p-i} \frac{(t^n)^j}{j!} v_{i+j} \right). \quad (4.13)$$

² φ_k -function for a scalar τ argument is defined as

$$\varphi_k(\tau) := \int_0^1 e^{(1-z)\tau} \frac{z^{k-1}}{(k-1)!} dz \quad (4.8)$$

with $\varphi_0(z) := e^z$, $\varphi_k(0) = \frac{1}{k!}$ and the recurrence relation of $(\varphi_k(\tau) - \varphi_k(0)) \tau^{-1} = \varphi_{k+1}(\tau)$.

We clearly see that the solution $q(t^{n+1})$ is a linear combination of φ -functions, i.e.,

$$q(t^{n+1}) = \sum_{i=0}^p (\Delta t)^i \phi_i(\Delta t L) b_i \quad (4.14)$$

with $b_0 := q(t^n)$ and $b_i := \sum_{j=0}^{p-i} \frac{(t^n)^j}{j!} v_{i+j}$. That means $(p+1)$ φ -function evaluations are needed. To reduce the computational burden on the evaluations, the authors [162] have proposed the alternative form based on the recurrence relation (4.8),

$$q(t^{n+1}) = \sum_{m=0}^{p-1} \frac{(\Delta t)^m}{m!} w_m + (\Delta t)^p \varphi_p(\Delta t L) w_p \quad (4.15)$$

with $w_0 = q(t^n)$ and $w_i = L w_{i-1} + \sum_{l=0}^{p-i} \frac{(t^n)^l}{l!} v_{l+i}$ for $i = 1, 2, \dots, p$. Only the action of $\varphi_p(\Delta t L)$ on a vector w_p is required. To compute $\varphi_p(\Delta t L) w_p$, we let $L \in \mathbb{R}^{\ell \times \ell}$, $w_p, e_1 \in \mathbb{R}^\ell$ and $J = \begin{pmatrix} 0 & J_{p-1} \\ 0 & 0 \end{pmatrix}$, and introduce an augmented matrix $\tilde{L} \in \mathbb{R}^{(\ell+p) \times (\ell+p)}$,

$$\tilde{L} = \begin{pmatrix} L & w_p e_1^T \\ 0 & J \end{pmatrix}. \quad (4.16)$$

From the Theorem 1 of [190], the exponential of the augmented matrix yields

$$e^{\Delta t \tilde{L}} = \begin{pmatrix} e^{\Delta t L} & \Delta t \varphi_1(\Delta t L) w_p & (\Delta t)^2 \varphi_2(\Delta t L) w_p & \cdots & (\Delta t)^p \varphi_p(\Delta t L) w_p \\ 0 & 1 & \frac{\Delta t}{1!} & \cdots & \frac{(\Delta t)^{p-1}}{(p-1)!} \\ 0 & 0 & 1 & \cdots & \frac{(\Delta t)^{p-2}}{(p-2)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & \frac{\Delta t}{1!} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (4.17)$$

That is, the first ℓ entries of the last column of $e^{\Delta t \tilde{L}}$ gives the vector $(\Delta t)^p e^{\Delta t L} w_p$, i.e.,

$$\underbrace{\begin{pmatrix} (\Delta t)^p \varphi_p(\Delta t L) w_p \\ \tilde{w} \end{pmatrix}}_{\in \mathbb{R}^{(\ell+p)}} = \underbrace{e^{\Delta t \tilde{L}}}_{\in \mathbb{R}^{(\ell+p) \times (\ell+p)}} \underbrace{\begin{pmatrix} 0 \\ e_p \end{pmatrix}}_{\in \mathbb{R}^{(\ell+p)}} \quad (4.18)$$

where $e_p = (0, 0, \dots, 1)^T \in \mathbb{R}^p$ and $\tilde{w} = \left(\frac{(\Delta t)^{p-1}}{(p-1)!}, \dots, \Delta t, 1 \right)^T$.

However, the procedure in (4.15) may become sensitive to rounding errors when a large coefficient of φ -function is involved [82]. Thus, authors [82] have suggested to compute (4.14) simultaneously with the generalized augmented matrix [5],

$$\tilde{L} = \begin{pmatrix} L & B \\ 0 & J \end{pmatrix}, \quad (4.19)$$

where $B := (b_1, b_2, \dots, b_p) = b_1 e_1^T + b_2 e_2^T + \dots + b_p e_p^T$. The matrix exponential of the augmented matrix (4.19) yields

$$e^{\Delta t \tilde{L}} = \begin{pmatrix} e^{\Delta t L} & \Delta t \varphi_1(\Delta t L) b_1 & \sum_{i=1}^2 (\Delta t)^i \varphi_i(\Delta t L) b_i & \cdots & \sum_{i=1}^p (\Delta t)^i \varphi_i(\Delta t L) b_i \\ 0 & 1 & \frac{\Delta t}{1!} & \cdots & \frac{(\Delta t)^{p-1}}{(p-1)!} \\ 0 & 0 & 1 & \cdots & \frac{(\Delta t)^{p-2}}{(p-2)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & \frac{\Delta t}{1!} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (4.20)$$

Now we can obtain the solution $q(t^{n+1})$ by taking the top ℓ elements from the matrix exponential,

$$\begin{pmatrix} q(t^{n+1}) \\ \tilde{w} \end{pmatrix} = e^{\Delta t \tilde{L}} \begin{pmatrix} b_0 \\ e_p \end{pmatrix}. \quad (4.21)$$

4.1.1 Krylov subspace projection-based methods

Now, we need to explain how to evaluate the action of φ -functions on a vector. Since φ -functions can be solved simultaneously using the augmented matrix (4.19), we focus on how $e^A b$ is approximated in Krylov subspace projection-based methods [182]. Given a matrix $A \in \mathbb{R}^{n \times n}$ and a vector $b \in \mathbb{R}^n$, we seek an approximation of $e^A b \in \mathcal{K}_m$,

$$\mathcal{K}_m := \text{span} \{b, Ab, A^2 b, \dots, A^{m-1} b\}$$

in m dimensional Krylov subspace whose dimension is $m \ll n$. To that end, we generate an orthonormal basis set

$$V_m := (v_1, v_2, \dots, v_m)$$

through Arnoldi iterations [11], and project the action of A to the Krylov subspace,

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^T, \quad (4.22)$$

where $H_m = V_m^T A V_m \in \mathbb{R}^{m \times m}$ is the Hessenberg matrix; h_{ij} ³ is the element of H_m ; and $e_m \in \mathbb{R}^m$ is the m th canonical unit vector. Now, the action of e^A on b can be approximated with the small matrix H_m by

$$e^A b \simeq V_m e^{H_m} V_m^T b = \|b\|_2 V_m e^{H_m} e_1 \quad (4.23)$$

with $v_1 := \frac{b}{\|b\|_2}$ and $V_m^T v_1 = e_1$. Note that the size of H_m is m -by- m . The evaluation of e^{H_m} is less expensive than that of e^A . Next, we evaluate the matrix exponential of e^{H_m} using Padé approximation via the scaling and squaring method [5].

The accuracy of the approximation (4.23) relies on the size of the Krylov subspace, m and the distribution of the eigenvalues of A . By Theorem 2 in [190]⁴, the error (4.23) satisfies

$$\underbrace{e^A b - \|b\|_2 V_m e^{H_m} e_1}_{\epsilon} = \|b\|_2 h_{m+1,m} \sum_{j=1}^{\infty} e_m^T \varphi_j(H_m) e_1 A^{j-1} v_{m+1}. \quad (4.25)$$

This suggests *a posteriori* error estimate,

$$\|\epsilon\|_2 \simeq \|b\|_2 \|h_{m+1,m} e_m^T \varphi_1(H_m) e_1\|_2 \quad (4.26)$$

under the assumption that the magnitude of φ_1 dominates the error. The *a posteriori* error estimate is easily computed by introducing an augmented matrix $\tilde{H}_m \in$

³ $h_{m+1,m}$ can be considered as a residual of the projection of A onto the Krylov subspace \mathcal{K}_m [82].

⁴ We obtain (4.25) when $p = 0$ from Theorem 2 in [190], which states that

$$\varphi_p(\tau A) b - \|b\|_2 V_m \varphi_p(\tau H_m) e_1 = \|b\|_2 \tau h_{m+1,m} \sum_{j=p+1}^{\infty} e_m^T \varphi_j(\tau H_m) e_1 (\tau A)^{j-p-1} v_{m+1}. \quad (4.24)$$

$\mathbb{R}^{(m+1) \times (m+1)}$

$$\tilde{H}_m = \begin{pmatrix} H_m & e_1 \\ 0 & 0 \end{pmatrix}, \quad (4.27)$$

and computing its matrix exponential

$$e^{\tilde{H}_m} = \begin{pmatrix} e^{H_m} & \varphi_1(H_m)e_1 \\ 0 & 1 \end{pmatrix}. \quad (4.28)$$

Once we know the error estimate, we can control the accuracy of the approximation (4.23). If the spectrum of A is broad, then we can either expand the Krylov subspace or downsize the eigenvalues of A . When $A = \Delta t \tilde{L}$ (in (4.20)), we can split Δt into a sum of smaller intervals, i.e., $\Delta t = \tau_1 + \tau_2 + \dots + \tau_k$, and approximate the matrix exponential iteratively ⁵,

$$e^{\Delta t L} b = e^{\tau_k L} \left(\dots \left(e^{\tau_2 L} \left(e^{\tau_1 L} b \right) \right) \right). \quad (4.29)$$

Based on *a posteriori* error in (4.26), the authors [162] have developed **phim** algorithm to carry out the matrix exponential efficiently by adaptively adjusting the size of the Krylov subspace and the number of sub-steps. Along the line, recently, **KIOPS** in [82] and **phipm/IOM2** in [133] have been developed to improve the adaptive Krylov algorithm **phim** by replacing the Arnoldi iteration with the incomplete orthogonalization procedure [119].

4.2 Governing equations and spatial discretization

In this section, we focus on the construction of the linear L and the nonlinear operator \mathcal{N} in (4.4). The linear operator should capture the rapidly changing characteristics of the system for accuracy and stability of exponential integrators. Due to

⁵This is equivalent to a time-stepping method with the small time intervals.

this reason, we utilize Jacobian $\frac{\partial R(q)}{\partial q}$ of the right-hand side $R(q)$ in (4.3). By adding and subtracting the Jacobian, we construct the linear and the nonlinear operator as follows: $Lq = \frac{\partial R(q)}{\partial q}$ and $\mathcal{N}(q) := R(q) - Lq$. Indeed, this choice transforms the first-order ETD1 in (4.9) to the second-order EPI2 [202],

$$q^{n+1} = q^n + \Delta t \varphi_1(L\Delta t)R^n$$

where $q^{n+1} = q(t^{n+1})$, $q^n = q(t^n)$, $\mathcal{N}^n := \mathcal{N}(q^n)$ and $R^n := R(q^n)$. We will numerically demonstrate the benefit of using Jacobian as the linear operator in Section 4.3.1.

4.2.1 Constructing a linear and a nonlinear operator in DG

Given a partial differential equation (PDE),

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathcal{F} = \mathbf{s}, \quad (4.30)$$

we split the full flux tensor into a linear (stiff) part $\nabla \cdot \mathcal{F}_L$ and a nonlinear (non-stiff) part $\nabla \cdot (\mathcal{F} - \mathcal{F}_L)$ at a continuous level ⁶,

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathcal{F}_L + \nabla \cdot (\mathcal{F} - \mathcal{F}_L) = \mathbf{s}, \quad (4.31)$$

then we apply the DG spatial discretization to (4.31). The DG discretization of the decomposed system reads: seek $\mathbf{q} \in \mathbf{V}_h(\Omega_h)$ such that

$$\left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v} \right)_{\Omega_h} = L\mathbf{q} + \mathcal{N}(\mathbf{q}), \quad (4.32a)$$

for all $\mathbf{v} \in \mathbf{V}_h(\Omega_h)$, where

$$\begin{aligned} \mathcal{N}(\mathbf{q}) &= (\mathcal{F}_N(\mathbf{q}), \nabla \mathbf{v})_{\Omega_h} - \langle \mathcal{F}_N^*(\mathbf{q}^\pm) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h}, \\ L\mathbf{q} &= (\mathcal{F}_L(\mathbf{q}), \nabla \mathbf{v})_{\Omega_h} + (\mathbf{s}(\mathbf{q}), \mathbf{v})_{\Omega_h} - \langle \mathcal{F}_L^*(\mathbf{q}^\pm) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h} \end{aligned}$$

⁶ This approach avoids taking a partial derivative of the stabilization parameter in a numerical flux.

for the weak form and

$$\begin{aligned}\mathcal{N}(\mathbf{q}) &= -(\nabla \mathcal{F}_{\mathcal{N}}(\mathbf{q}), \mathbf{v})_{\Omega_h} - \langle (\mathcal{F}_{\mathcal{N}}^*(\mathbf{q}^\pm) - \mathcal{F}_{\mathcal{N}}(\mathbf{q})) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h}, \\ L\mathbf{q} &= -(\nabla \mathcal{F}_L(\mathbf{q}), \mathbf{v})_{\Omega_h} + (\mathbf{s}(\mathbf{q}), \mathbf{v})_{\Omega_h} - \langle (\mathcal{F}_L^*(\mathbf{q}^\pm) - \mathcal{F}_L(\mathbf{q})) \cdot \mathbf{n}, \mathbf{v} \rangle_{\partial\Omega_h}\end{aligned}$$

for the strong form. Here, $\mathcal{F}_{\mathcal{N}} := \mathcal{F} - \mathcal{F}_L$ is a nonlinear flux; $\mathcal{F}_{\mathcal{N}}^* := \mathcal{F}^* - \mathcal{F}_L^*$ is a nonlinear DG numerical flux; and \mathcal{F}_L^* is a linear DG flux.

4.2.2 Euler equations

We consider Euler equations

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = s_\rho, \quad (4.33a)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + p \mathcal{J}) = s_{\rho \mathbf{u}}, \quad (4.33b)$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho \mathbf{u} H) = s_{\rho E}, \quad (4.33c)$$

where ρ is the density; \mathbf{u} is the velocity; p is the pressure; $\rho E = \rho e + \frac{1}{2} \rho \|\mathbf{u}\|^2$ is the total energy; $e = \frac{p}{\gamma-1}$ is the internal energy; $H = E + \frac{p}{\rho} = \frac{a^2}{\gamma-1} + \frac{1}{2} \|\mathbf{u}\|^2$ is the total specific enthalpy; $a = \sqrt{\frac{p}{\rho}}$ is the sound speed; γ is the ratio of the specific heat; \mathcal{J} is the rank-1,2 or 3 identity matrix; and \mathbf{s} is a source term.

For simplicity, we focus on one-dimensional Euler equations. The core of the linearization is to introduce a flux Jacobian. In a compact form, (4.33) can be written as

$$\frac{\partial \mathbf{q}}{\partial t} = -\frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \mathbf{s} \quad (4.34)$$

with $\mathbf{q} = (\rho, \rho u, \rho E)^T$, $\mathbf{s} = (s_\rho, s_{\rho u}, s_{\rho E})^T$ and $\mathbf{F}(\mathbf{q}) = (\rho u, \rho u u + p, \rho u H)^T$. We define the flux Jacobian $A := \frac{\partial \mathbf{F}}{\partial \mathbf{q}}$ by

$$A = \begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{2}(\gamma-3)u^2 & (3-\gamma)u & \gamma-1 \\ u(\frac{1}{2}(\gamma-1)u^2 - H) & H + (1-\gamma)u^2 & \gamma u \end{pmatrix}, \quad (4.35)$$

evaluate A at a reference state $\tilde{\mathbf{q}}$, and construct the following linear flux, $\mathbf{F}_L := A(\tilde{\mathbf{q}})\mathbf{q} = \tilde{A}\mathbf{q}$. By adding and subtracting the linear flux, we decompose (4.34) into the following form,

$$\frac{\partial \mathbf{q}}{\partial t} = \underbrace{\left(-\frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \frac{\partial \tilde{A}\mathbf{q}}{\partial x}\right)}_{\mathcal{N}(\mathbf{q})} + \underbrace{\left(-\frac{\partial \tilde{A}\mathbf{q}}{\partial x} + \mathbf{s}\right)}_{L\mathbf{q}}. \quad (4.36)$$

By multiplying a test function \mathbf{v} to (4.36), integrating by parts twice for each element and summing all the elements, we arrive at the strong formulation: seek $\mathbf{q} \in \mathbf{V}_h(\Omega_h)$ such that

$$\begin{aligned} \left(\frac{\partial \mathbf{q}}{\partial t}, \mathbf{v}\right)_{\Omega_h} &= \left(-\frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \frac{\partial \tilde{A}\mathbf{q}}{\partial x}, \mathbf{v}\right)_{\Omega_h} \\ &\quad + \left\langle n \left(-(\mathbf{F}(\mathbf{q}^\pm))^* + (\tilde{A}\mathbf{q}^\pm)^* \right), \mathbf{v} \right\rangle_{\partial\Omega_h} \\ &\quad - \left\langle n \left(-\mathbf{F}(\mathbf{q}) + (\tilde{A}\mathbf{q}) \right), \mathbf{v} \right\rangle_{\partial\Omega_h} \\ &\quad + (\mathbf{s}, \mathbf{v})_{\Omega_h} - \left(\frac{\partial \tilde{A}\mathbf{q}}{\partial x}, \mathbf{v} \right)_{\Omega_h} \\ &\quad - \left(n \left((\tilde{A}\mathbf{q}^\pm)^* - \tilde{A}\mathbf{q} \right), \mathbf{v} \right)_{\partial\Omega_h} \end{aligned} \quad (4.37)$$

for all $\mathbf{v} \in \mathbf{V}_h(\Omega_h)$. As for the numerical flux, we use Roe flux [179] for both $\mathbf{F}^*(\mathbf{q}^\pm)$ and $(\tilde{A}\mathbf{q}^\pm)^*$, i.e.,

$$\mathbf{F}^*(\mathbf{q}^\pm) = \{\!\!\{ \mathbf{F}(\mathbf{q}) \}\!\!\} + \frac{1}{2}|A(\mathbf{q})|[\mathbf{q}], \quad (4.38a)$$

$$(\tilde{A}\mathbf{q}^\pm)^* = \{\!\!\{ \tilde{A}\mathbf{q} \}\!\!\} + \frac{1}{2}|\tilde{A}|[\mathbf{q}], \quad (4.38b)$$

where $|A| := R|\Lambda|R^{-1}$, $|\Lambda| = \text{diag}(|u-a|, |u|, |u+a|)$ absolute eigenvalues,

$$\tilde{R} = \begin{pmatrix} 1 & 1 & 1 \\ u-a & u & u+a \\ H-au & \frac{u^2}{2} & H+au \end{pmatrix}$$

right eigenvectors and its inverse

$$R^{-1} = \begin{pmatrix} \left(\frac{\gamma-1}{4}\right) \frac{u^2}{a^2} + \frac{u}{2a} & -\left(\frac{\gamma-1}{a}\right) \frac{u}{a^2} - \frac{1}{2a} & \frac{\gamma-1}{2a^2} \\ 1 - \left(\frac{\gamma-1}{2}\right) \frac{u^2}{a^2} & (\gamma-1) \frac{u}{a^2} & \frac{1-\gamma}{a^2} \\ \left(\frac{\gamma-1}{4}\right) \frac{u^2}{a^2} - \frac{u}{2a} & -\left(\frac{\gamma-1}{a}\right) \frac{u}{a^2} + \frac{1}{2a} & \frac{\gamma-1}{2a^2} \end{pmatrix}$$

whose rows are left eigenvectors. The readers can refer Appendix A for the flux Jacobian of two- or three-dimensional Euler equations.

4.3 Numerical Results

4.3.1 A system of two ordinary differential equations (ODE)

We consider the autonomous system of two ODEs proposed by [54]

$$\frac{du(t)}{dt} = cu - v + (\lambda v - cu)r^2, \quad (4.39a)$$

$$\frac{dv(t)}{dt} = u + cv - (\lambda v + cu)r^2 \quad (4.39b)$$

where $r^2 = u^2 + v^2$, $c > 0$ constant, $u = r \cos \theta$, $v = r \sin \theta$. The exact solution is given by

$$r^2(t) = r_0^{-2} \left(r_0^2 (1 - e^{-2ct}) + e^{-2ct} \right), \quad (4.40a)$$

$$\theta(t) = \theta_0 + (1 - \lambda)t - \frac{\lambda}{2c} \log \left(r_0^2 (1 - e^{-2ct}) \right) + e^{-2ct}. \quad (4.40b)$$

For a numerical simulation, we take $c = 100$, $\lambda = 1/2$, $u_0 = u(0) = 2$ and $v_0 = v(0) = 1$. The exact solution with the parameters is shown in Figure 4.1. The magnitude and the phase of the solution rapidly change within a short period.

How do we choose a linear operator to capture the characteristics of a rapidly changing system? To address the question, we consider the two linear operators: the dynamic Jacobian operator

$$L_d := \left(\frac{\partial R}{\partial(u, v)} \right) = \begin{pmatrix} c(1 - r^2) + 2(\lambda v - cu)u & -1 + \lambda r^2 + 2(\lambda v - cu)v \\ 1 - \lambda r^2 - 2(\lambda u + cv)u & c(1 - r^2) - 2(\lambda u + cv)v \end{pmatrix} \quad (4.41)$$

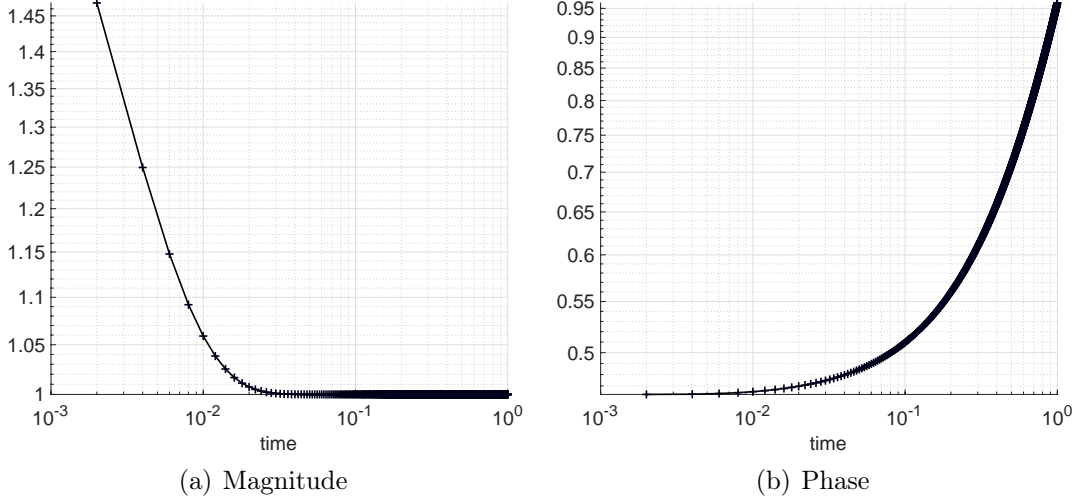


Figure 4.1: System of two ODEs: (a) magnitude and (b) phase of the exact solution of (4.40a).

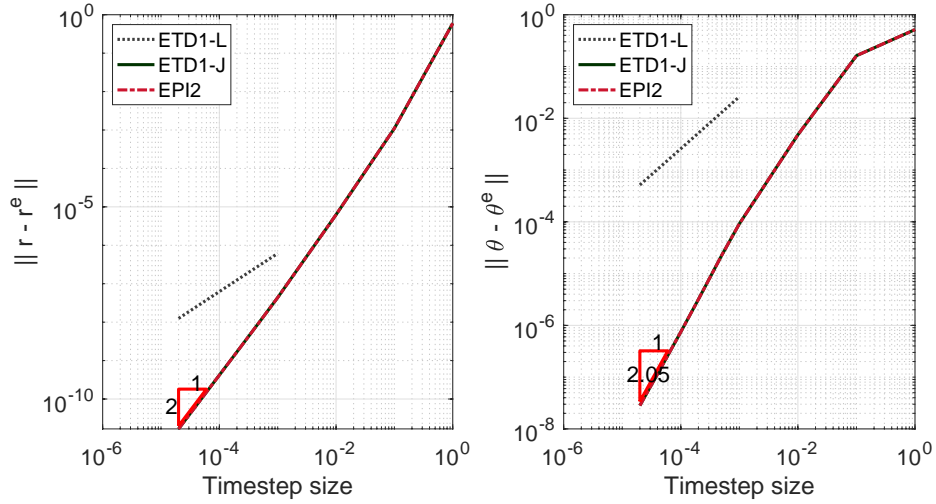


Figure 4.2: System of two ODEs: time convergence study of ETD1-L, ETD1-J and EPI2. We observe the first-order convergence rate for ETD1-L and the second-order convergence rate for ETD1-J and EPI2. The use of Jacobian improves not only the accuracy, but also the stability of ETD1.

evaluated at each timestep t^n , and the constant linear operator [54]

$$L_c := \begin{pmatrix} c & -1 \\ 1 & c \end{pmatrix}. \quad (4.42)$$

Figure 4.2 demonstrate the benefit of the dynamic Jacobian in terms of accuracy.

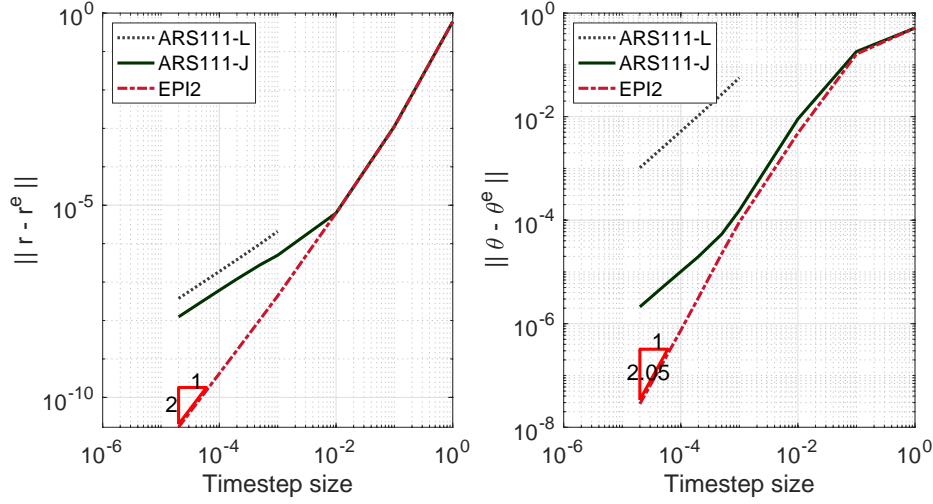


Figure 4.3: System of two ODEs: time convergence study of ARS111-L, ARS111-J and EPI2. ARS111-L shows the first-order convergence rate, whereas ARS111-J shows the first-order convergence rate near 10^{-4} timestep size, but the second-order convergence rate as loosen the timestep size.

We measure the errors of magnitude r and phase θ at $T = 1$ for ETD1 and EPI2 methods. ETD1-L and ETD1-J stand for ETD1 method with the constant linear operator (4.42) and the dynamic Jacobian (4.41), respectively. The error of ETD1-L shows the first-order convergence rate in both the magnitude and the phase errors, whereas ETD1-J counterpart shows the second-order convergence rate. Not only that, ETD1-J allows larger timestep size than ETD1-L. This demonstrates the importance of using the dynamic Jacobian.

We also perform the time convergence study for ARS111 in Figure 4.3. ARS111-L and ARS111-J stand for ARS111 method with the constant linear operator (4.42) and the dynamic Jacobian (4.41), respectively. ARS111-L shows the first-order convergence rate, whereas ARS111-J shows the first-order convergence rate near 10^{-4} timestep size, but the second-order convergence rate as loosen the timestep size. This implies that the error associated with the fast modes starts to dominate as tighten the timestep size. We also notice that the difference of the phase error between

Table 4.1: System of two ODEs: time convergence study of ARS111 and ETD1. Using L_d gives better stability and accuracy than using L_c . With the small timestep size ($dt < 10^{-3}$), ETD1 solutions are more accurate than ARS111 solutions.

L	dt	ETD1				ARS1			
		$\ r - r^e\ $	rate	$\ \theta - \theta^e\ $	rate	$\ r - r^e\ $	rate	$\ \theta - \theta^e\ $	rate
L_c	2.0E-5	1.3E-08	-	5.2E-04	-	3.8E-08	-	1.0E-03	-
	5.0E-5	3.1E-08	1.0	1.3E-03	1.0	9.4E-08	1.0	2.6E-03	1.0
	1.0E-4	6.2E-08	1.0	2.6E-03	1.0	1.9E-07	1.0	5.2E-03	1.0
	2.0E-4	1.3E-07	1.0	5.2E-03	1.0	3.8E-07	1.0	1.1E-02	1.0
	5.0E-4	3.1E-07	1.0	1.3E-02	1.0	9.9E-07	1.0	2.7E-02	1.0
	1.0E-3	6.1E-07	1.0	2.6E-02	1.0	2.1E-06	1.1	5.6E-02	1.1
L_d	2.0E-5	1.7E-11	-	2.8E-08	-	1.3E-08	-	2.1E-06	-
	5.0E-5	1.0E-10	2.0	1.8E-07	2.0	3.1E-08	1.0	5.2E-06	1.0
	1.0E-4	4.2E-10	2.0	7.5E-07	2.0	6.1E-08	1.0	1.0E-05	1.0
	2.0E-4	1.7E-09	2.0	3.2E-06	2.1	1.2E-07	1.0	2.0E-05	1.0
	5.0E-4	1.1E-08	2.0	2.3E-05	2.1	2.8E-07	0.9	5.4E-05	1.1
	1.0E-3	4.4E-08	2.0	9.2E-05	2.0	5.0E-07	0.8	1.6E-04	1.5
	1.0E-2	6.3E-06	2.2	4.8E-03	1.7	6.3E-06	1.1	9.0E-03	1.8
	1.0E-1	1.1E-03	2.2	1.6E-01	1.5	1.1E-03	2.3	1.8E-01	1.3
	1.0	6.0E-01	2.7	5.2E-01	0.5	6.0E-01	2.7	5.2E-01	0.5

ARS111-L and ARS111-J is relatively larger than that of the magnitude error. The same behavior is observed between ETD1-L and ETD1-J. The dynamic Jacobian L_d significantly improves the phase error. We tabulate the time convergence results in Table 4.1. We see that ETD1 solutions are more accurate than ARS111 solutions with the small timestep sizes ($dt < 10^{-3}$). This makes sense because the IMEX scheme is known to damp out the high-frequency modes in the system, whereas exponential integrators exactly treat the fast modes. In particular, exponential integrators subdivide a timestep size Δt into several sub-steps $\Delta t = \tau_1 + \tau_2 + \dots$ and march the approximation step-by-step similar to an explicit time integration method. Overall, exponential integrators are promising in the sense that it is not only accurate but

also stable with a large timestep size.

4.3.2 Isentropic vortex translation

We study an isentropic vortex example [234], where small perturbations of a vortex are added to a uniform mean flow and are translated without changing its shape. The mean flows are set to be $(u_\infty, v_\infty, \rho_\infty, T_\infty) = (0.2, 0, 1, 1)$ and the perturbations are given as

$$\begin{aligned} u' &= -\frac{\lambda}{2\pi} e^{(1-r^2)} (y - y_c), \\ v' &= +\frac{\lambda}{2\pi} e^{(1-r^2)} (x - x_c), \\ T' &= -\frac{(\gamma - 1) \lambda^2}{16\gamma\pi^2} e^{2(1-r^2)} \end{aligned}$$

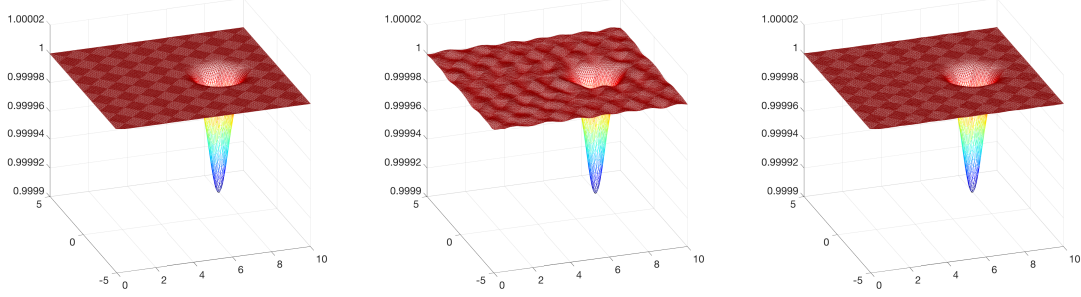
with $r = \|\mathbf{x} - \mathbf{x}_c\|$, $\mathbf{x}_c = (5, 0)$, and $\lambda = 0.05$ the vortex strength. The exact solutions are generated from the isentropic relation,

$$\rho = (T_\infty + T')^{1/(\gamma-1)}.$$

We take $\gamma = 1.4$ and the domain $\Omega = [0, 10] \times [-5, 5]$. Periodic boundary conditions apply to all directions. For a three-dimensional simulation, we take the zero vertical velocity, $w = 0$ and extrude domain vertically, $\Omega = [0, 10] \times [-5, 5] \times [0, 1]$.

We perform the simulation for $t \in [0, 10]$ on the grid with $k = 12$ solution order and $N_e = 256$ elements in Figure 4.4. For time discretization, RK4 with $Cr = 0.86$, EPI2 with $Cr = 85.56$ and EXPRB42 [133] with $Cr = 85.56$ are chosen. As can be seen in Figure 4.4(b) and Figure 4.4(c), EPI2 and EXPRB42 can use one-hundred times larger timestep size than RK4, and still capture the vortex movement. However, we see the solution of EPI2 degrades due to the time truncation error in Figure 4.4(b). To alleviate the error, we can use a high-order time integration method.

For example, in Figure 4.4(c), the time truncation error of EXPRB42 is significantly reduced compared to EPI2 counterpart.



(a) RK4 with $Cr = 0.86$ (b) EPI2 with $Cr = 85.56$ (c) EXPRB42 with $Cr = 85.56$

Figure 4.4: Isentropic vortex example to the two-dimensional Euler equation is simulated from $t \in [0, 10]$: (a) RK4 with $Cr = 0.86$, (b) EPI2 with $Cr = 8.56$ and (c) EXPRB42 with $Cr = 85.56$ are chosen for $k = 12$ and $N_e = 256$ spatial discretization.

To examine the error growth rate, time convergence studies are conducted with $k = \{8, 16\}$ solution order for $t \in [0, 1]$ on a uniform and a non-uniform mesh in Figure 4.5. The results for ρ on a uniform mesh are summarized in Figure 4.6 and Figure 4.7. In Figure 4.6(a), we observe the second-order convergence rate of EPI2 and the fourth-order convergence rate of EXPRB42 in a wide range of Courant number. The errors are saturated at $\mathcal{O}(10^{-13})$ error level, from which the spatial discretization error starts to dominate. In Figure 4.6(b), we see that the high-order EXPRB42 scheme is beneficial compared to the second-order EPI2 at $\mathcal{O}(10^{-10})$ error level. However, we still see that RK4 scheme is more economical than EXPONENTIAL DG methods at $\mathcal{O}(10^{-13})$ error level. When we increase a spatial discretization error, however, the wall clock time of EPI2 becomes comparable to that of RK4 in Figure 4.7.

On the non-uniform mesh, we start to see the benefit of EXPONENTIAL DG

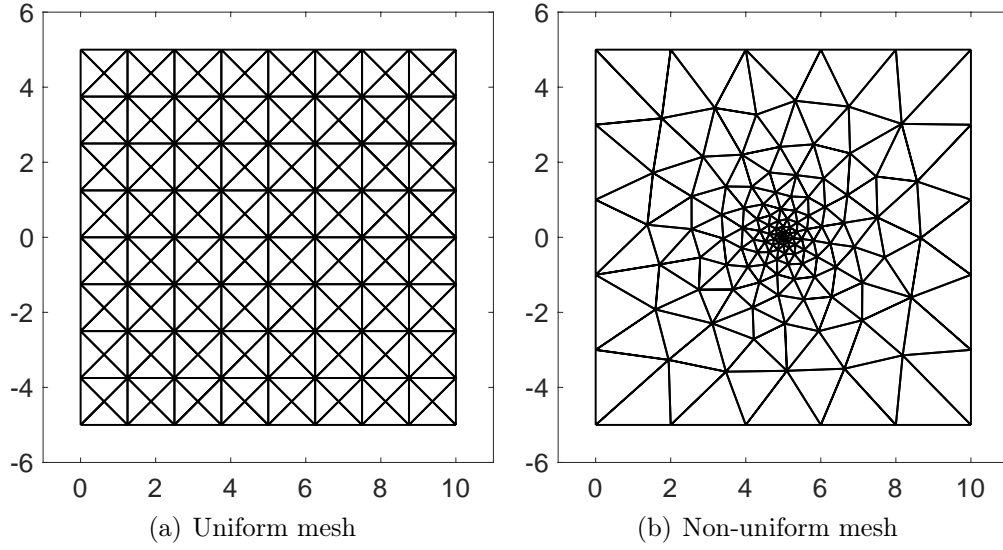


Figure 4.5: (a) Uniform mesh with $N_e = 256$ elements and (b) non-uniform meshes with $N_e = 250$ elements.

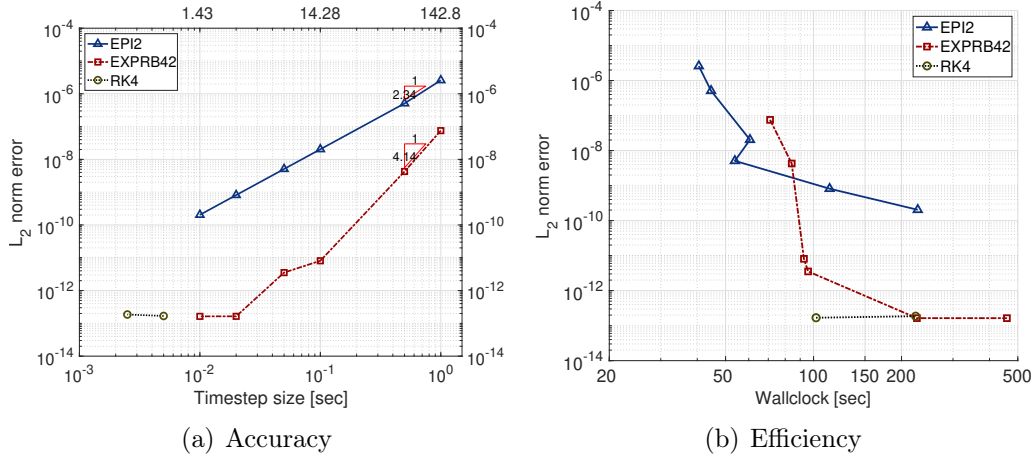


Figure 4.6: Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 16$.

schemes. In Figure 4.8(b), the wall clock time of EXPRB42 becomes comparable to RK4 at $\mathcal{O}(10^{-12})$ error level. We add a spatial discretization error by lowering the solution order from $k = 16$ to $k = 8$. We clearly see the computational gain in Figure 4.9(b). At $\mathcal{O}(10^{-9})$ error level, EPI2 is two times faster than RK4.

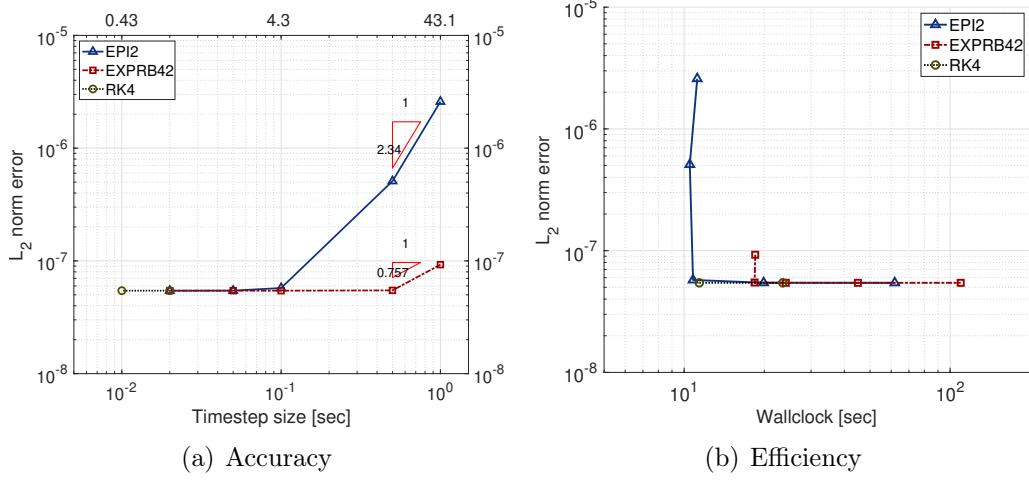


Figure 4.7: Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 8$.

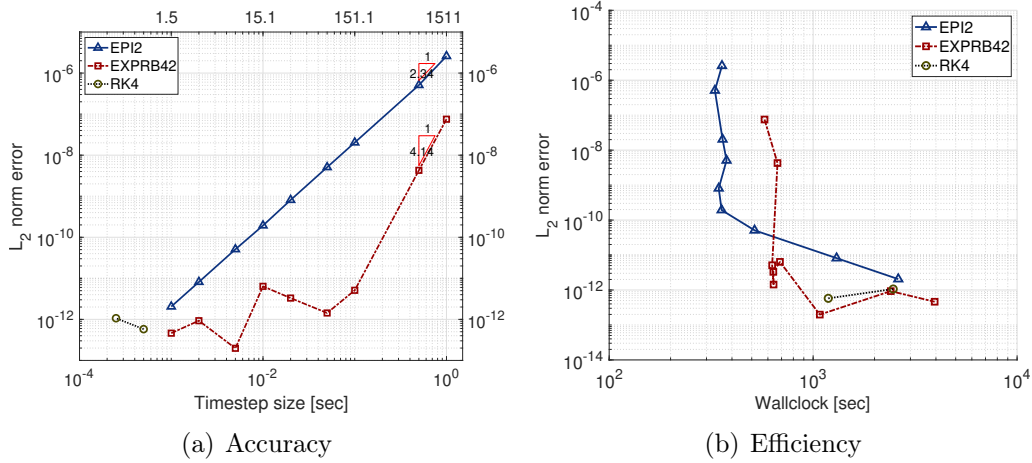


Figure 4.8: Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on non-uniform mesh. The computational domain is discretized with $N_e = 250$ and $k = 16$.

Figure 4.10 shows snapshots of advecting isentropic vortex at $t = [0, 10, 25]$. For spatial discretization, we take $N_e = 6400$ elements and $k = 8$ solution order. For time discretization, we choose EPI2 with $dt = 0.25$ ($Cr = 27.9$). We see the center of the vortex is located at lateral boundary at $t = 25$, because the mean speed of the

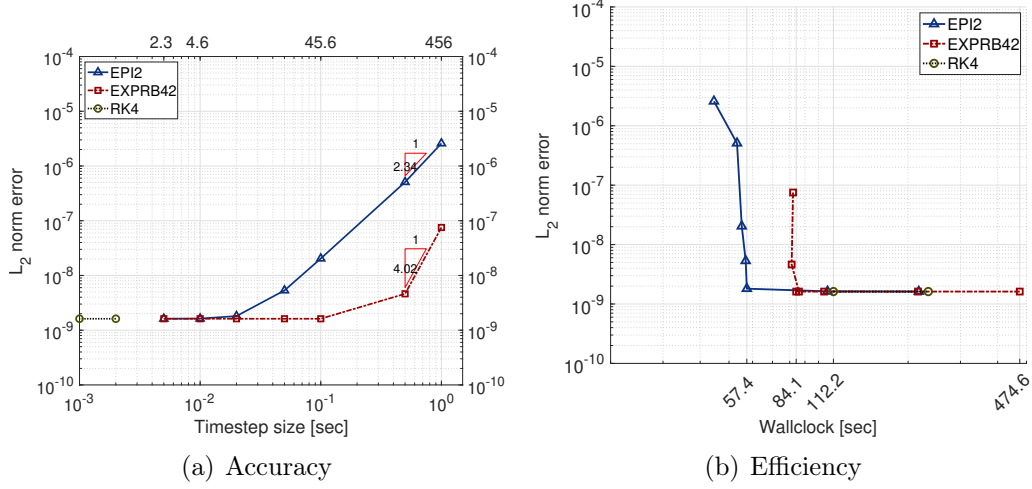


Figure 4.9: Isentropic vortex translation in two dimensions: time convergence study for (a) accuracy and (b) efficiency on a non-uniform mesh. The computational domain is discretized with $N_e = 256$ and $k = 8$.

flow is 0.2 and the half-length of the domain is 5.

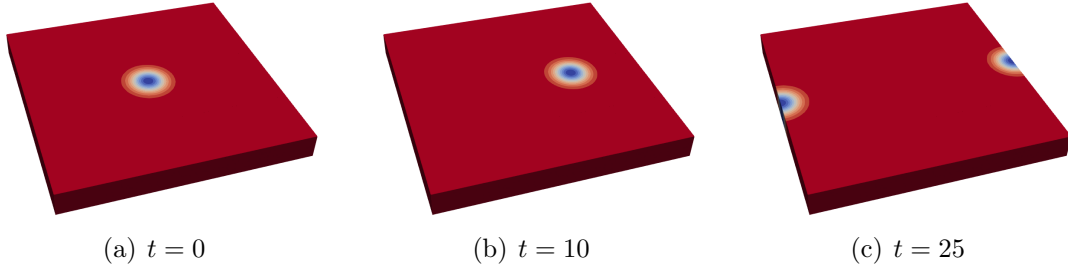


Figure 4.10: Isentropic vortex translation in three dimensions: snapshots for (a) EPI2 with $Cr = 27.9(dt = 0.25)$. Simulations are performed on the grid with $N_e = 6400$ and $k = 8$ solution order for $t \in [0, 20]$.

Now we study the parallel performance of the Exponential DG methods. To this end, we have implemented the adaptive Krylov subspace method [82] on top of the DG finite element library written in C++, which is the spin-off library from `mang11` [223]. For computing a matrix exponential, we use `EIGEN` C++ library. Parallel implementations mainly focus on vector-vector multiplication and matrix-vector multiplication. The matrix-vector multiplication is carried out in a matrix-free man-

ner by calling the linear operator $L\mathbf{q}$ in (4.32). By the virtue of the DG discretization, the computation of $L\mathbf{q}$ is embarrassingly parallel. For a dot product (vector-vector multiplication), we initially scatter a global vector to each processor, and then we perform the dot product with the local vector. Next, we call `MPI ALLreduce` function.

Parallel simulations are conducted on Stampede2 at the Texas Advanced Computing Center (TACC). Stampede 2 provides 4,200 Knights Landing (KNL) nodes and 1,736 Skylake (SKX) nodes. Each node of KNL consists of 68 cores of Intel Xeon Phi 7250 1.4GHz processors, 96 GB of traditional DDR4 RAM, and 16 GB high-speed MCDRAM. Each node of SKX consists of 48 cores of Intel Xeon Platinum 8160 2.1 GHz processors and 192GB DDR4 RAM. The interconnect is a 100 GB/s Intel Omni-Path (OPA) network with a fat-tree topology. We conduct strong and weak scaling studies on SKX system.

In Table 4.2, strong scaling results of EPI2 are compared with RK4 counterpart. Domain is discretized with $N_e = 6400$ elements and $k = 8$ solution order. We take $dt = 0.25(Cr = 27.9)$ for EPI2 and $dt = 0.005(Cr = 0.56)$ for RK4 and perform the simulations up to $t = 2$. In general, EPI2 is comparable to RK4 in terms of wall clock time. The efficiency ⁷ of EPI2 is slightly better than RK4 counterpart. For both EPI2 and RK4, we see the efficiency drops as increasing the number of processors. This is because the number of elements per core decreases as increasing cores so that the communication cost starts to dominate over the computational cost.

Next, we assign additional elements to each processor by refining the mesh. Table 4.3 compares the efficiencies of two different spatial discretization: (i) $N_e =$

⁷ Efficiency is defined by $\frac{T_s}{T_p np}$ with T_s serial wall clock time and T_p parallel wall clock time. Here, we approximate $T_s \simeq T_{base} \times np_{base}$ where T_{base} is parallel wall clock time with np_{base} cores.

Table 4.2: Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with $Cr = 27.9(dt = 0.25)$ and RK4 with $Cr = 0.56(dt = 0.005)$ are performed with $N_e = 6400$ and $k = 8$ solution order up to $t = 2$.

#cores	$N_e/core$	EPI2		RK4	
		Wall clock [s]	Efficiency	Wall clock [s]	Efficiency
32	200	1283	100	1014	100
64	100	537.3	119.4	501.6	101.1
128	50	424.8	75.5	315.5	80.3
256	25	148.6	107.9	225.9	56.1
512	12.5	114.1	70.3	118.1	53.7
1024	6.3	64.38	62.3	64.49	49.1
2048	3.1	39.38	50.9	37.11	42.7
4096	1.6	24.39	41.1	20.17	39.3

51200 elements and $k = 6$ solution order, and (ii) $N_e = 6400$ elements and $k = 8$ solution order. As expected, efficiency with the former discretization is higher than the latter.

To understand the effect of the timestep size on efficiency, we conduct strong scaling tests with two different timestep sizes, $dt = 0.125$ and $dt = 0.25$, on the mesh with $N_e = 51200$ elements and $k = 6$ solution order. As shown in Table 4.4, the efficiency with $dt = 0.125$ is a little bit higher than that with $dt = 0.25$. This is because the latter case involves more Krylov iterations than the former case. The total number of Krylov steps involved exponential time integrations is 561 for $dt = 0.125$ and 1024 for $dt = 0.25$. This implies that the spectrum of the linear operator becomes broad by increasing the timestep size.

Based on the above observations, we conduct the strong scaling test for EPI2 with $Cr = 8.1(dt = 0.03125)$, $N_e = 3,276,800$ and $k = 4$ solution order. We use the consecutive processors, $np = \{16, 32, 64, 128, 256, 512, 868\} \times 48$ so that the number of elements per core becomes $\{4266.7, 2133.3, 1066.7, 533.3, 266.7, 133.3, 78.6\}$. In Figure

Table 4.3: Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with different degrees-of-freedom (DOF). For the number of DOF (NDOF) of 17,561,600, simulations are performed on $N_e = 51200$ and $k = 6$ solution order with $dt = 0.25(Cr = 32.98)$ up to $t = 1$, whereas for NDOF of 4,665,600, simulations are performed on $N_e = 6400$ and $k = 8$ solution order with $dt = 0.25(Cr = 27.93)$ up to $t = 2$.

#cores	NDOF = 17,561,600		NDOF = 4,665,600	
	$N_e/core$	Efficiency	$N_e/core$	Efficiency
32	1600	100	200	100
64	800	97.9	100	119.4
128	400	82.7	50	75.5
256	200	85.7	25	107.9
512	100	61.9	12.5	70.3
1024	50	59.2	6.3	62.3
2048	25	55.0	3.1	50.9
4096	12.5	51.4	1.6	41.1

4.11, the speedup ⁸ factor shows a good strong scalability up to 41664 cores (which is the maximum cores in Skylake system in TACC).

Now we perform the weak scaling test. For the weak scaling test, we assign the same amount of work to each processor while increasing the number of processors. We discretize the domain with $N_e = \{1, 8, 64, 512, 4096\} \times 100$ elements and $k = 8$ solution order, and decompose the domain with $np = \{4, 32, 256, 2048, 16384\}$ processors. That is, we make the number of elements per core to be 25. We also keep the Courant number fixed so that the total number of Krylov steps, N_{Krylov} to be uniform. We have tabulated the weak scaling results in Table 4.5. We observe that N_{Krylov} tends to be linearly proportional to the Courant number. We also see that the increment of wall clock time is relatively small for a fixed Courant number. We plot the average time per timestep against the number of degrees-of-freedom in Figure 4.12, from which

⁸ Speedup is defined as $\frac{T_s}{T_p}$ with T_s serial wall clock time and T_p parallel wall clock time.

Table 4.4: Isentropic vortex translation in three dimensions: strong scaling results for EPI2 with $Cr = 16.49(dt = 0.125)$ and $Cr = 32.98(dt = 0.25)$ are performed with $N_e = 51200$ and $k = 6$ solution order up to $t = 1$.

#cores	$N_e/core$	$dt = 0.125$		$dt = 0.25$	
		Wall clock [s]	Efficiency	Wall clock [s]	Efficiency
32	1600	2346	100	4151	100
64	800	1199	97.8	2121	97.9
128	400	607.7	96.5	1255	82.7
256	200	306.9	95.6	605.7	85.7
512	100	237.3	61.8	419.1	61.9
1024	50	122.7	59.7	219.1	59.2
2048	25	61.33	59.5	118.0	55.0
4096	12.5	33.72	54.4	63.12	51.4

we see exponential DG methods shows a good weak scaling result.

Table 4.6 shows that the total number of Krylov steps linearly depends on the timestep size and the refining level. We refine the mesh by factor of two along x, y and z direction. That is, we add geometrically-induced stiffness to the system. This is why we see the wall clock time tends to increase with the number of degrees-of-freedom. The corresponding wall clock times are summarized in Figure 4.13.

4.4 Discussions

In this chapter, we have developed an EXPONENTIAL DG framework where we construct the decomposed system of (4.32) by adding and subtracting a linear term, and employ the adaptive Krylov subspace projection-based methods. In particular, to accommodate rapidly varying characteristics in the system, we take a dynamic flux Jacobian as the linear operator. Besides, we split the system at a continuous level instead of a discrete one. This approach avoids taking the complicated derivatives of the stabilization parameter in the DG formulation.

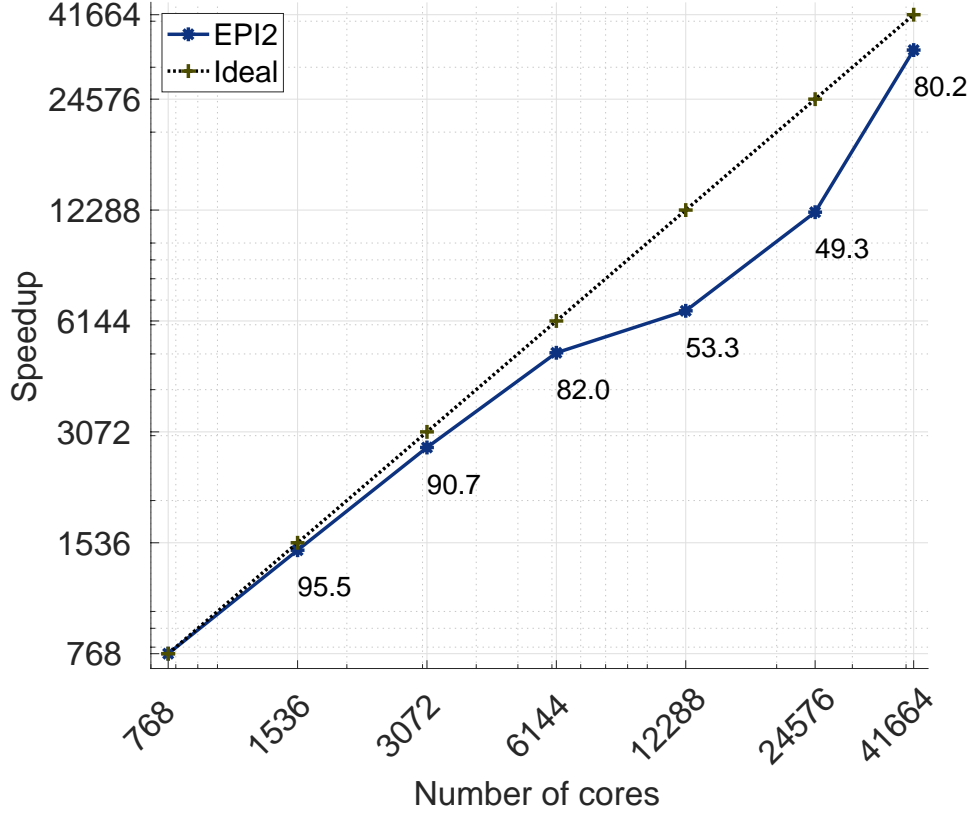


Figure 4.11: Isentropic vortex translation in three dimensions: strong scaling study for EPI2 with $Cr = 8.1$ ($dt = 0.03125$). The computational domain is discretized with $N_e = 3, 276, 800$ and $k = 4$ solution order for $np = \{16, 32, 64, 128, 256, 512, 868\} \times 48$.

The purpose of our approach is threefold: (i) to use larger timestep sizes (compared to explicit integrators); (ii) to accurately capture rapidly varying characteristics (compared to IMEX schemes); (iii) to achieve good scalable performance for large-scale simulations;

Numerical results demonstrate that exponential DG methods are stable and achieve larger timestep sizes. Depending on the spectrum of the dynamic Jacobian, the adaptive Krylov subspace methods automatically adjust the size of Krylov subspace and the sub-step intervals. Compared to the fourth-order explicit RK method, the EXPONENTIAL DG scheme can be beneficial when a non-uniform mesh is in-

Table 4.5: Isentropic vortex translation in three dimensions: weak scaling results for EPI2 with fixed Courant numbers. Each core has 18,225 degrees-of-freedom (=25 elements and 8th solution order).

Cr	N_e	dt	Final time	Wall clock [s]	N_{Krylov}
5.59	4	0.2	0.8	15.3	103
	32	0.1	0.4	17.8	90
	256	0.05	0.2	24.4	81
	2048	0.025	0.1	27.8	81
	16384	0.0125	0.05	29.3	82
22.4	4	0.2	0.8	50.4	397
	32	0.1	0.4	89.9	381
	256	0.05	0.2	85.8	355
	2048	0.025	0.1	86.9	348
	16384	0.0125	0.05	93.1	358
55.9	4	0.2	0.8	130.9	1024
	32	0.1	0.4	240.4	1023
	256	0.05	0.2	246.2	1024
	2048	0.025	0.1	248.8	1024
	16384	0.0125	0.05	342.2	1408

voled. Since a mesh refinement technique is necessary tool for complex simulations, the EXPONENTIAL DG scheme would be a viable option to overcome the timestep size restriction.

Parallelization of the adaptive Krylov subspace method involves matrix-vector and vector-vector multiplications. The matrix-vector product is performed in a matrix-free manner using the DG method, which is embarrassingly parallel. The vector-vector product (dot product) requires global communication, but the communication cost is relatively small compared to the computational cost. From the weak and the strong scaling tests, we observe that the EXPONENTIAL DG method provides a promising scalable solution to Euler systems.

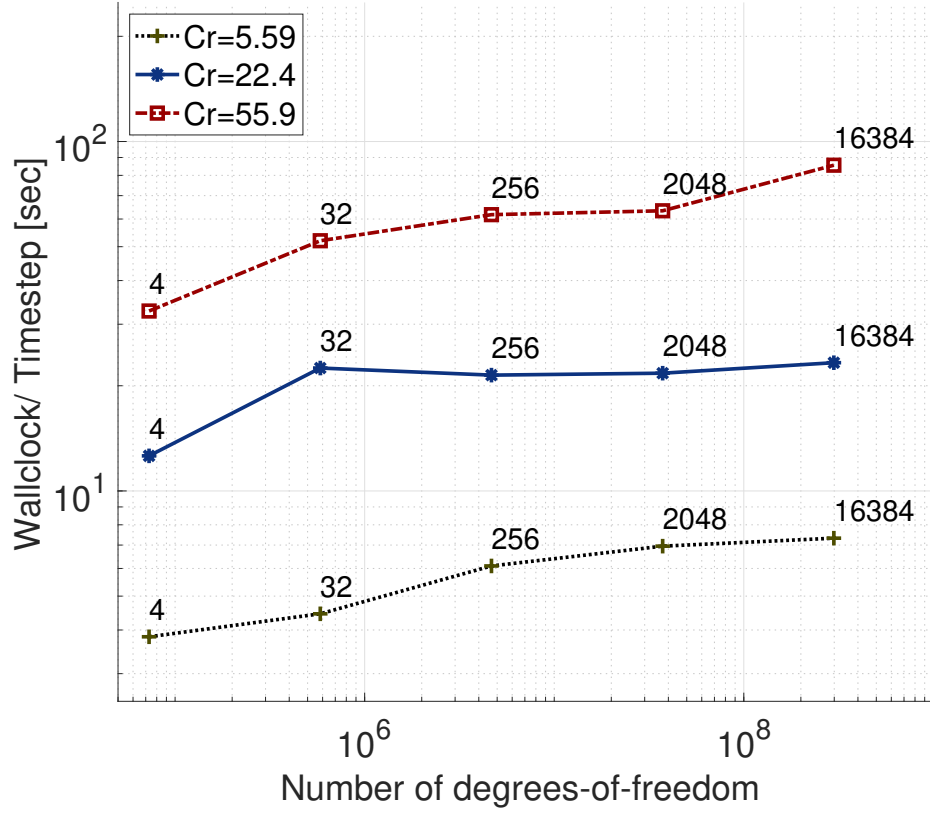


Figure 4.12: Isentropic vortex translation in 3D: weak scaling study with fixed Courant numbers. The computational domain is discretized with $N_e = \{1, 8, 64, 512, 4096\} \times 100$ and $k = 8$ solution order for $np = \{4, 32, 256, 2048, 16384\}$, respectively, so that each processor has 25 elements.

In future work, we focus on extending the idea to various PDE and optimizing the performance of exponential time integrators. Besides, the stability analysis of exponential DG schemes is also of interest.

Table 4.6: Isentropic vortex translation in three dimensions: the total number of Krylov steps, N_{Krylov} , linearly increases by (i) the refining level, and (ii) Courant number.

dt	$N_e = 100$		$N_e = 800$		$N_e = 6400$		$N_e = 51200$	
	Cr	N_{Krylov}	Cr	N_{Krylov}	Cr	N_{Krylov}	Cr	N_{Krylov}
0.025	0.7	40	1.4	40	2.8	45	5.6	81
0.25	7.0	128	14.0	233	27.9.	448	55.9	1024
2.5	69.8	1536	139.7	3072	279.32	5120	558.7	10112

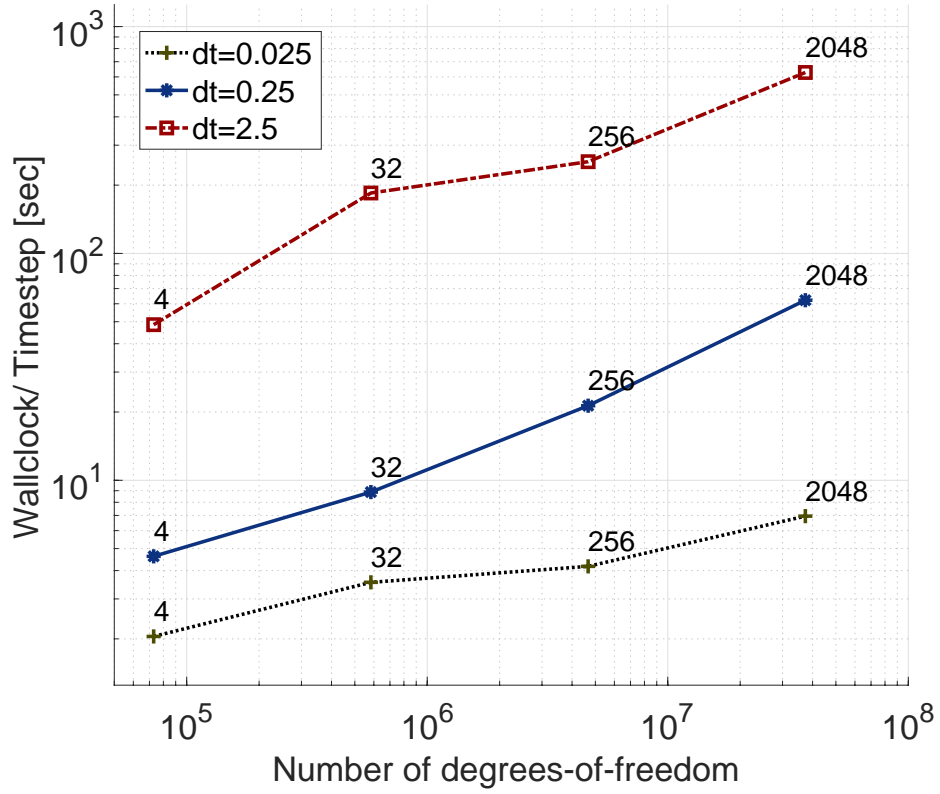


Figure 4.13: Isentropic vortex translation in three dimensions: weak scaling study with fixed timestep size. The computational domain is discretized with $N_e = \{100, 800, 6400, 51200\}$ and $k = 8$ solution order for $np = \{4, 32, 256, 2048\}$, respectively, so that each processor has 25 elements.

Chapter 5

Sliding-mesh Interface

In this chapter ¹, we are interested in simulating rotating flows. Inspired by [233], we develop a sliding-mesh interface in the context of (hybridized) DG spatial discretization. The key ingredients are to utilize mortar interfaces and Arbitrary Lagrangian-Eulerian (ALE) method. To handle nonconforming meshes, we introduce a mortar interface such that only two adjacent elements are allowed. In this manner, we only treat conforming mortar interfaces, where we construct projection operators to transfer data from a face to a mortar, vice-versa. When a mesh rotates, the size of the mortars between stationary and rotating meshes dynamically changes. To treat a moving mesh, the ALE method is employed. Within the ALE framework, the motion in the stationary and the rotating meshes can be described in terms of the deformation gradient and its Jacobian.

This chapter is organized as follows. Section 5.1 explains the curved nonconforming treatment in (hybridized) DG spatial discretization. In Section 5.2, we recall ALE methods for linear and curved elements. Section 5.3 combines the curved nonconforming mesh and ALE method to realize the sliding mesh interface. We perform several numerical experiments to demonstrate the idea.

¹ A manuscript is in preparation for the contents of this chapter. The contributions of the author ranged from numerical implementation of the algorithm and writing the manuscript.

5.1 Nonconforming mesh treatment

A (non) conforming interior interface Ξ is shared by two neighboring elements K^+ and K^- where we denote q^\pm as the trace of the solutions from K^\pm . Figure 5.1 shows an example of a nonconforming mesh. Mortar element Ξ_1 is shared by volume elements K_1 and K_2 . To compute a numerical flux on $\Xi_1 \cap K_2$, we project the solutions on K_1 and K_2 to Ξ_1 , compute the numerical flux on Ξ_1 , and then project it back to $\Xi_1 \cap K_2$. Similarly, we compute the numerical flux on $\Xi_2 \cap K_3$ by projecting the solutions on K_1 and K_3 to Ξ_2 . We construct the numerical flux such that the amount of the numerical flux on $\{\Xi_1, \Xi_2\} \cap K_1$ is the same as the sum of the numerical flux computed on $\Xi_1 \cap K_1$ and on $\Xi_2 \cap K_1$ in the sense of L^2 projection.

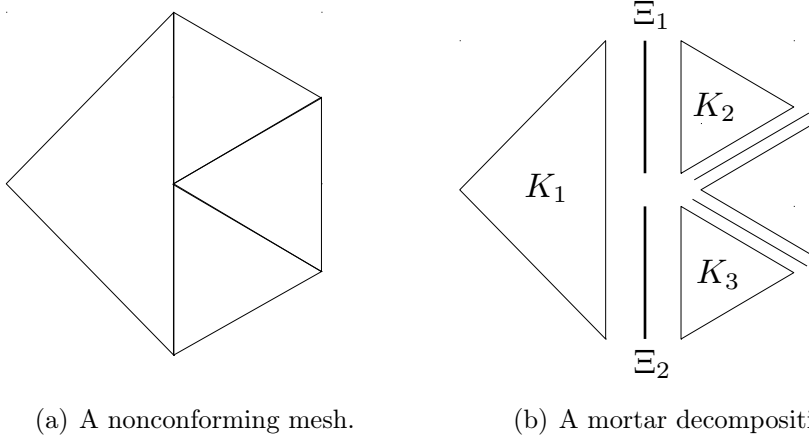


Figure 5.1: Examples of (a) a linear nonconforming mesh and (b) a mortar decomposition. The mortar elements conform to the volume elements K_1 and K_2 .

To that end, we define two projection operators: $P^{(K,f) \rightarrow \Xi}$ from a face to a mortar and $P^{\Xi \rightarrow (K,f)}$ from a mortar to a face. We denote approximated solutions by

$$u^f = \sum_{j=1}^{N_{fp}} \ell_j(\xi)(u|_{(K,f)})_j \quad (5.1)$$

on a face f of an element K , and

$$u^\Xi = \sum_{j=1}^{N_{fp}} \hat{\ell}_j(z) u_j^\Xi \quad (5.2)$$

on a mortar element Ξ , where $\ell \in V_h(K, f)$ and $\hat{\ell} \in \Lambda_h(\Xi)$ are the Lagrange basis functions defined on the face and the mortar; $u|_{(K,f)}$ is the trace of u on the face f and the element K ; $\xi \in [-1, 1]$ is the parameterization of the face coordinate \mathbf{x}^f and $z \in [-1, 1]$ is that of the mortar coordinate \mathbf{x}^Ξ .

Since both the mortar Ξ and the face (K, f) share the intersection, we can relate ξ and z in terms of the offset o^Ξ and the stretch s^Ξ by $\xi = o^\Xi + s^\Xi z$. In case of a linear element as shown in Figure 5.1, the offset and the stretch are given as

$$o^\Xi = \frac{x_1^\Xi + x_2^\Xi - x_1^f - x_2^f + y_1^\Xi + y_2^\Xi - y_1^f - y_2^f}{x_2^f - x_1^f + y_2^f - y_1^f}, \quad (5.3a)$$

$$s^\Xi = \frac{x_2^\Xi - x_1^\Xi + y_2^\Xi - y_1^\Xi}{x_2^f - x_1^f + y_2^f - y_1^f}. \quad (5.3b)$$

When a mortar Ξ is on a circle ² (centered at (x_o, y_o) with a radius r) as shown in Figure 5.2, we can write $\mathbf{x}^f \in [\mathbf{x}_1^f, \mathbf{x}_2^f] = [\mathbf{x}_1^f(\theta_1^f), \mathbf{x}_2^f(\theta_2^f)]$ and $\mathbf{x}^\Xi \in [\mathbf{x}_1^\Xi, \mathbf{x}_2^\Xi] = [\mathbf{x}_1^\Xi(\theta_1^\Xi), \mathbf{x}_2^\Xi(\theta_2^\Xi)]$ in terms of θ only ³. Then we parameterize the angle $\theta \in [\theta_1^f, \theta_2^f]$ with $\xi \in [-1, 1]$ and $\theta \in [\theta_1^\Xi, \theta_2^\Xi]$ with $z \in [-1, 1]$, which results in the offset and the stretch as

$$o^\Xi = \frac{\theta_2^\Xi + \theta_1^\Xi - \theta_2^f - \theta_1^f}{\theta_2^f - \theta_1^f}, \quad (5.4a)$$

$$s^\Xi = \frac{\theta_2^\Xi - \theta_1^\Xi}{\theta_2^f - \theta_1^f}. \quad (5.4b)$$

² To deal with a curved element, we use Golden-Hall blending [90].

³ The physical coordinate on the circle is a function of an angle θ with a given radius $r \in \mathbb{R}$, i.e., $x = x_o + r \cos \theta$ and $y = y_o + r \sin \theta$.

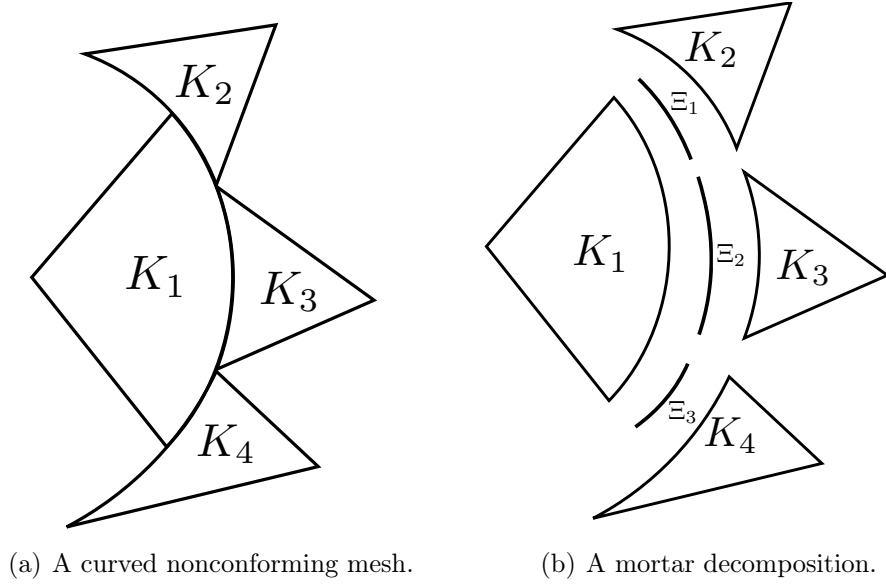


Figure 5.2: Examples of (a) a curved nonconforming mesh and (b) a mortar decomposition. The mortar element m_i is shared by two volume elements K_1 and K_{i+1} for $i = 1, 2$ and 3 .

5.1.1 Projection operator from a face to a mortar

We define the projection operator $P^{(K,f) \rightarrow \Xi} : V_h(K) \ni u|_{K,f} \rightarrow u^\Xi \in \Lambda_h(\Xi)$ such that

$$\langle u^f - u^\Xi, \hat{v} \rangle_{\Xi \cap \partial K} = 0, \quad (5.5)$$

holds for all $\hat{v} \in \Lambda_h(\Xi)$. By expanding u^f and u^Ξ with (5.1) and (5.2), and taking $\hat{v} = \hat{\ell}_i \in \Lambda_h(\Xi)$, we have the discrete projection operator as

$$u_i^\Xi = \underbrace{\hat{\mathbf{M}}_{ij}^{-1} \hat{\mathbf{B}}_{jk}^\Xi}_{\mathbf{P}_{ik}^{(K,f) \rightarrow \Xi}} u_k^f. \quad (5.6)$$

Here, $\hat{\mathbf{M}} = \left\langle \hat{\ell}_i(z), \hat{\ell}_j(z) J_s^\Xi \right\rangle_{[-1,1]}$ and $\hat{\mathbf{B}}^\Xi = \left\langle \hat{\ell}_i(z), \ell_j(o^\Xi + s^\Xi z) J_s^\Xi \right\rangle_{[-1,1]}$.

5.1.2 Projection operator from a mortar to a face

Once the numerical fluxes are computed on the mortar element, they need to be projected back to the adjacent faces. We interpret the surface integral on an element face (in local solver) as the sum of the individual contributions from its *adjacent* mortar elements, $\Sigma = \{\Xi | \Xi \cap (K, f) \neq \emptyset\}$, i.e.,

$$\left\langle \mathbf{n} \cdot \hat{\mathbf{F}}(\mathbf{x}), v \right\rangle_{(K,f)} = \sum_{\Xi_i \in \Sigma} \left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_i}(\mathbf{x}), v \right\rangle_{\Xi_i}, \quad (5.7)$$

for all $v \in V_h(K)$. By approximating the numerical fluxes on the face and the mortar, i.e., $\mathbf{n} \cdot \hat{\mathbf{F}}(\mathbf{x}) = \ell_j(\mathbf{x}) \left(\mathbf{n} \cdot \hat{\mathbf{F}} \right)_j$ and $\mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_i}(\mathbf{x}) = \hat{\ell}_j(\mathbf{x}) \left(\mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_i} \right)_j$, we can define the projection operator from the mortar to the face $P^{\Xi_i \rightarrow (K,f)} : \Lambda_h(\Xi_i) \ni u^{\Xi_i} \rightarrow u|_{K,f} \in V_h(K)$ requiring that

$$\left\langle u^f - \sum_{\Xi_i \in \Sigma} u^{\Xi_i}, v^f \right\rangle_{(K,f)} = 0, \quad (5.8)$$

holds for all $v^f \in V_h|_f$.

By expanding u^f and u^{Ξ} with (5.1) and (5.2), and taking $v^f = \ell_i \in V_h(K)|_f$, we have the discrete projection operator as

$$u_i^f = \sum_{\Xi_i \in \Sigma} \underbrace{\mathbf{M}_{ij}^{-1} \mathbf{B}^{(i)}}_{\mathbf{P}_{ij}^{\Xi_i \rightarrow (K,f)}} u_j^{\Xi_i}, \quad (5.9)$$

where $\mathbf{M}_{ij} = \langle \ell_i(\xi), \ell_j(\xi) J_s \rangle_{(-1,1)}$, $\mathbf{B}^{(i)} = \left\langle \ell_i(o^{\Xi_i} + s^{\Xi_i} z), \hat{\ell}_j(z) J_s^{\Xi_i} \right\rangle_{(-1,1)} = \left(\hat{\mathbf{B}}^{(i)} \right)^T$, J_s is the geometric Jacobian of the element face (K, f) and $J_s^{\Xi_i} = J_s s^{\Xi_i}$.

We construct the projection operator satisfying (5.7), which means that we can replace the surface integral shown in a weak form with the sum of mortar integrals, i.e.,

$$\left\langle \mathbf{n} \cdot \hat{\mathbf{F}}(\mathbf{x}), v \right\rangle_{\partial K} = \sum_{\Xi} \left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi}(\mathbf{x}), v \right\rangle_{\Xi \cap \partial K}, \quad (5.10)$$

For example, the surface integral

$$\left\langle \mathbf{n} \cdot \hat{\mathbf{F}}(\mathbf{x}), v \right\rangle_{\partial K_1}$$

can be replaced with

$$\left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_1}(\mathbf{x}), v \right\rangle_{\Xi_1} + \left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_2}(\mathbf{x}), v \right\rangle_{\Xi_2} + \left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_3}(\mathbf{x}), v \right\rangle_{\Xi_3},$$

for a conforming element in Figure 5.3(a), and

$$\sum_{i=1}^4 \left\langle \mathbf{n} \cdot \hat{\mathbf{F}}^{\Xi_i}(\mathbf{x}), v \right\rangle_{\Xi_i},$$

for a nonconforming element in Figure 5.3(b). From this perspective, we do not need to project the numerical flux computed on a mortar back to the adjacent faces.

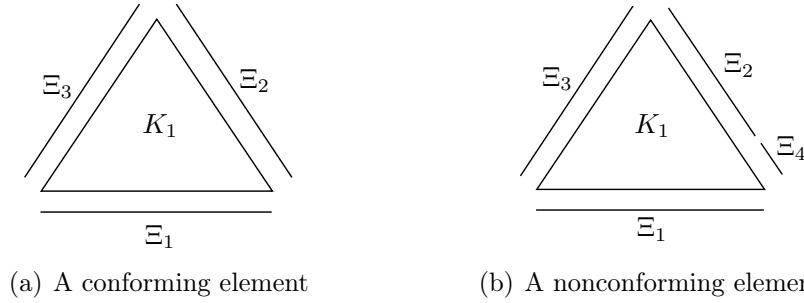


Figure 5.3: Examples of (a) a conforming element and (b) a nonconforming element.

5.1.3 Watertight mesh

When a polynomial approximation does not fully resolve a curved face, a geometrically induced aliasing error occurs. Suppose the curved face is shared by two conforming elements, each of which approximates the curved face with the same polynomial order N . In this case, there are no gaps between two adjacent faces, because each of the elements commits the same aliasing error. When this happens, we call the mesh is *watertight*. When one of the element approximates the face with a

different polynomial order M , there are gaps between two faces. Similarly, when the curved face is shared by nonconforming elements in Figure 5.2, there may be gaps. However, when we refine a curved element so that its child elements match the curved face of the parent element in Figure 5.4, we can construct a watertight mesh.

In Figure 5.4, we observe that the curved face of K_1 is attached to those of K_3 and K_4 . The curved face of K_1 is approximated by a single polynomial, whereas its counterparts (the curves of Ξ_1 and Ξ_2) are approximated by two piecewise polynomials. This means that the single polynomial space is a subspace of the direct sum of the piecewise polynomial spaces. Therefore the geometric information of the curved face of K_1 can be conveyed to the counterparts without losing information because it is an injective map.

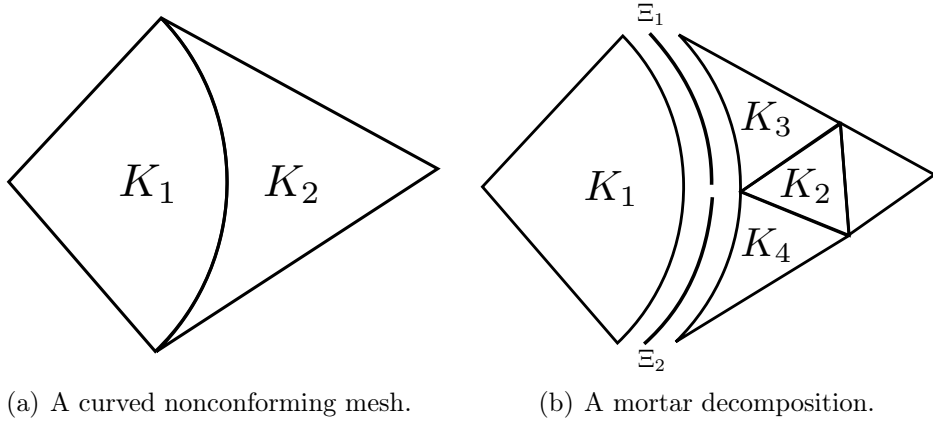


Figure 5.4: Examples of (a) a curved conforming mesh and (b) a local refinement. The parent element K_2 in (a) is subdivided into four child elements in (b).

For example, assume that the curved face of K_1 is an arc of a circle with the center (x_o, y_o) . When we approximate the arc with N order polynomial, we pick $(N + 1)$ points in $\xi \in [-1, 1]$ and compute $(x_i, y_i) = (x_o + r \cos \theta(\xi_i), y_o + r \sin \theta(\xi_i))$ for $i = 1, \dots, N + 1$. Then we construct N order interpolant functions for x and y ,

i.e.,

$$\mathbb{I}^N(x(\xi)) := \sum_{j=1}^{N+1} \ell_j(\xi) x_j, \quad \mathbb{I}^N(y(\xi)) := \sum_{j=1}^{N+1} \ell_j(\xi) y_j.$$

Next, to ensure watertight mesh, we evaluate the interpolant functions $\mathbb{I}^N(x)$ and $\mathbb{I}^N(y)$ at the points $\xi^\Xi \in [\xi_a, \xi_b]$ on the mortar m_k for $k = 1, 2$, i.e.,

$$\tilde{x}_i = \mathbb{I}^N(x(\xi_i^\Xi)), \quad \tilde{y}_i = \mathbb{I}^N(y(\xi_i^\Xi))$$

with $\xi_i^\Xi = o^\Xi + s^\Xi z_i$ for $i = 1, \dots, N+1$. Finally, we build N order interpolant functions on the mortar m_k for x and y , i.e.,

$$\mathbb{I}^{N,\Xi}(x(z)) := \sum_{j=1}^{N+1} \hat{\ell}_j(z) \tilde{x}_j, \quad \mathbb{I}^{N,\Xi}(y(z)) := \sum_{j=1}^{N+1} \hat{\ell}_j(z) \tilde{y}_j.$$

We notice that the union of the interpolant functions $\mathbb{I}^{N,\Xi}(\mathbf{x})$ is equivalent to $\mathbb{I}^N(\mathbf{x})$.

5.2 Arbitrary Lagrangian-Eulerian

In an arbitrary Lagrangian-Eulerian (ALE) framework, the mesh can move with a different velocity from the flow. Since the mesh movement is independent of the flow, this method is useful for moving or deforming a body. The basic idea is to construct a map between the deformed body in the physical domain and the corresponding reference domain, and then transform the original equation in terms of the reference domain.

We consider an abstract PDE in the conservative form

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F}(u, \nabla u) = 0, \tag{5.11}$$

where u is a scalar quantity and \mathbf{F} is a column vector. To obtain the ALE formulation, we integrate (5.11) over a time-varying domain $\vartheta(t)$,

$$\int_{\vartheta(t)} \frac{\partial u}{\partial t} d\vartheta(t) + \int_{\partial\vartheta(t)} \mathbf{F} \cdot \mathbf{n} da = 0. \tag{5.12}$$

On the other hand, applying the Reynolds transport theorem for the total rate of change following $\vartheta(t)$ and combining with the conservation law (5.11), we have

$$\left. \frac{d}{dt} \right|_{\vartheta(t)} \int_{\vartheta(t)} u d\vartheta(t) + \int_{\partial\vartheta(t)} (\mathbf{F} - u\mathbf{u}_G) \cdot \mathbf{n} da = 0, \quad (5.13)$$

where \mathbf{u}_G is the velocity of $\partial\vartheta(t)$.

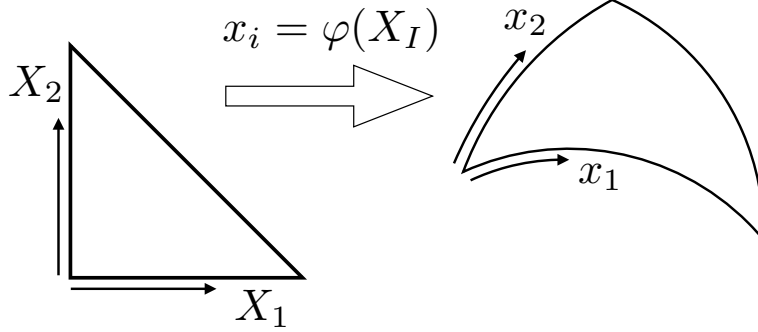


Figure 5.5: Mapping between physical to reference domain.

We now cast (5.13) into a fixed reference domain $\vartheta(0)$. To this end, let us denote by $\mathbf{x} = \varphi(\mathbf{X})$ the map from $\vartheta(0)$ to $\vartheta(t)$. Using the relation ⁴ $\mathbf{n} da = J\mathcal{G}^{-T}\mathbf{N}dA$, where $\mathcal{G} = \frac{\partial\varphi}{\partial\mathbf{X}}$ the deformation gradient and J is the determinant of the deformation gradient, we have

$$\int_{\vartheta(0)} \frac{\partial(Ju)}{\partial t} d\vartheta(0) + \int_{\partial\vartheta(0)} J(\mathcal{G}^{-1}(\mathbf{F} - u\mathbf{u}_G)) \cdot \mathbf{N} dA = 0. \quad (5.14)$$

By applying the divergence theorem, we have the transformed PDE in the reference domain,

$$\frac{\partial u_X}{\partial t} + \nabla_{\mathbf{X}} \cdot \mathbf{F}_{\mathbf{X}} = 0, \text{ in } \vartheta(0), \quad (5.15)$$

⁴ \mathbf{n} and da are a normal vector and a differential area on physical domain, whereas \mathbf{N} and dA are the counterparts on the reference domain. A differential volume $d\vartheta(t)$ in the physical domain is related to that $\vartheta(0)$ in the reference domain as $d\vartheta(t) = Jd\vartheta(0)$. Since the volume is the dot product of the differential area and the differential length, we can write $d\mathbf{x} \cdot \mathbf{n} da = Jd\mathbf{X} \cdot \mathbf{N} dA$. By using $d\mathbf{x} = \mathcal{G}d\mathbf{X}$, we have the relation of $\mathbf{n} da = J\mathcal{G}^{-T}\mathbf{N} dA$.

where $u_X = Ju$ and $\mathbf{F}_X = J\mathcal{G}^{-1}(\mathbf{F} - u\mathbf{u}_G)$ contravariant flux⁵.

When \mathbf{F} is a function of ∇u , we can cast (5.11) into the first-order form by introducing an auxiliary variable \mathbf{Q} ,

$$\mathbf{Q} - \nabla u = 0, \quad (5.16a)$$

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F}(u, \mathbf{Q}) = 0. \quad (5.16b)$$

The auxiliary variable \mathbf{Q} can be rewritten in the reference domain $\vartheta(0)$ as

$$\mathbf{Q} = \nabla u = \frac{\partial u}{\partial \mathbf{X}} \frac{\partial \mathbf{X}}{\partial \mathbf{x}} = J^{-1} \mathcal{G}^{-T} (\nabla_{\mathbf{x}} u_X - J^{-1} (\nabla_{\mathbf{x}} J) u_X). \quad (5.17)$$

This leads to the governing equation in the reference domain $\vartheta(0)$,

$$\mathbf{Q}_X - \nabla_{\mathbf{x}} u_X + J^{-1} (\nabla_{\mathbf{x}} J) u_X = 0, \quad (5.18a)$$

$$\frac{\partial u_X}{\partial t} + \nabla_{\mathbf{x}} \cdot \mathbf{F}_X(u_X, \mathbf{Q}_X) = 0 \quad (5.18b)$$

with $\mathbf{Q}_X := J\mathcal{G}^T \mathbf{Q}$.

5.2.1 Weak formulation

Testing (5.18) with $v \in V_h(K(0))$ and $\mathbf{P} \in \mathbf{V}_h(K(0))$, we obtain

$$\begin{aligned} (\mathbf{Q}_X, \mathbf{P})_{K(0)} + (u_X, \nabla_{\mathbf{x}} \cdot \mathbf{P})_{K(0)} - \langle \hat{u}_X \mathbf{N}, \mathbf{P} \rangle_{\partial K(0)} \\ + (J^{-1} (\nabla_{\mathbf{x}} J) u_X, \mathbf{P})_{K(0)} = 0, \end{aligned} \quad (5.19a)$$

$$\left(\frac{\partial u_X}{\partial t}, v \right)_{K(0)} - (\mathbf{F}_X, \nabla_{\mathbf{x}} v)_{K(0)} + \langle \hat{\mathbf{F}}_X \cdot \mathbf{N}, v \rangle_{\partial K(0)} = 0, \quad (5.19b)$$

for all $K(0) \in \Omega_h(0)$ ⁶, where $\hat{\mathbf{F}}_X(u_X, \mathbf{Q}_X, \hat{u})$ is a numerical flux (which is defined later).

⁵ Note that each row of \mathcal{G}^{-1} is a contravariant basis vector. Thus, the i th component of the contravariant flux can be denoted by $(\mathbf{F}_X)^i = J \nabla_{\mathbf{x}}^i \cdot (\mathbf{F} - u\mathbf{u}_G)$.

⁶ $K(0) := K(t=0)$, $\Omega_h(0) := \Omega_h(t=0)$ and $\Xi(0) := \Xi(t=0)$.

For the HDG formulation, since we treat a trace \hat{u} as an unknown value, to close the system, we enforce the continuity of the numerical flux across each interior edge, i.e.,

$$\left\langle \left[\mathbf{N} \cdot \hat{\mathbf{F}}_{\mathbf{X}}(u_X, \mathbf{Q}_{\mathbf{X}}, \hat{u}) \right], \hat{v} \right\rangle_{\Xi(0)} = 0, \quad (5.20)$$

for all $\hat{v} \in \Lambda_h(\Xi(0))$.

Note that we impose a boundary condition through a numerical flux or a trace variable. For practical implementation, it is convenient to rewrite the numerical flux in (5.19) and (5.20) in terms of physical fluxes and boundary conditions [74]. This leads to

$$\left\langle \left[\overline{\mathbf{F}(u, \mathbf{Q}, \hat{u}) - u \mathbf{u}_G \cdot \mathbf{n}} \right], \hat{v} \right\rangle_{\Xi(t)} = 0 \quad (5.21)$$

for (5.20), and

$$\begin{aligned} (\mathbf{Q}_{\mathbf{X}}, \mathbf{P})_{K(0)} + (u_X, \nabla_{\mathbf{X}} \cdot \mathbf{P})_{K(0)} - \langle \hat{u} \mathcal{G}^T \mathbf{n}, \mathbf{P} \rangle_{\partial K(t)} \\ + (J^{-1} (\nabla_{\mathbf{X}} J) u_X, \mathbf{P})_{K(0)} = 0, \end{aligned} \quad (5.22a)$$

$$\begin{aligned} \left(\frac{\partial u_X}{\partial t}, v \right)_{K(0)} - (\mathbf{F}_{\mathbf{X}}, \nabla_{\mathbf{X}} v)_{K(0)} \\ + \left\langle \overline{\mathbf{F}(u, \mathbf{Q}, \hat{u}) - u \mathbf{u}_G \cdot \mathbf{n}}, v \right\rangle_{\partial K(t)} = 0, \end{aligned} \quad (5.22b)$$

for (5.19) with some manipulations ⁷.

Summing (5.22) and (5.21) over all elements and all edges, we obtain the HDG system: find the approximate solution $(\mathbf{Q}_{\mathbf{X}}, u_X, \hat{u}) \in \mathbf{V}_h(\Omega_h(0)) \times V_h(\Omega_h(0)) \times$

⁷ Here we implicitly assume that $J\mathcal{G}^{-1}$ is continuous across element boundaries. $\hat{\mathbf{F}}_{\mathbf{X}} \cdot \mathbf{N} dA = \left(J\mathcal{G}^{-1} \overline{\mathbf{F}} - u \mathbf{u}_G \right)^T \mathbf{N} dA = \overline{\mathbf{F}} - u \mathbf{u}_G^T (J\mathcal{G}^{-T} \mathbf{N} dA) = \overline{\mathbf{F}} - u \mathbf{u}_G^T \mathbf{n} da$. We also use $\hat{u}_X \mathbf{N} dA = \hat{u} J \mathbf{N} dA = \hat{u} \mathcal{G}^T \mathbf{n} da$.

$\Lambda_h(\mathcal{E}_h(t))$ such that

$$\begin{aligned} & (\mathbf{Q}_{\mathbf{x}}, \mathbf{P})_{\Omega_h(0)} + (u_X, \nabla_{\mathbf{x}} \cdot \mathbf{P})_{\Omega_h(0)} - \langle \hat{u} \mathcal{G}^T \mathbf{n}, \mathbf{P} \rangle_{\partial\Omega_h(t) \setminus \Gamma_D} \\ & - \langle u_D \mathcal{G}^T \mathbf{n}, \mathbf{P} \rangle_{\Gamma_D} + (J^{-1}(\nabla_{\mathbf{x}} J) u_X, \mathbf{P})_{\Omega_h(0)} = 0, \end{aligned} \quad (5.23a)$$

$$\begin{aligned} & \left(\frac{\partial u_X}{\partial t}, v \right)_{\Omega_h(0)} - (\mathbf{F}_{\mathbf{x}}, \nabla_{\mathbf{x}} v)_{\Omega_h(0)} + \left\langle \widehat{\mathbf{F} - u \mathbf{u}_G \cdot \mathbf{n}}, v \right\rangle_{\partial\Omega_h(t) \setminus \Gamma_D} \\ & + \left\langle \widehat{\mathbf{F}(u, \mathbf{Q}, u_D) - u_D \mathbf{u}_G \cdot \mathbf{n}}, v \right\rangle_{\Gamma_D} = 0, \end{aligned} \quad (5.23b)$$

$$\left\langle \left[\widehat{\mathbf{F} - u \mathbf{u}_G \cdot \mathbf{n}} \right], \hat{v} \right\rangle_{\mathcal{E}_h^o(t)} + \left\langle \hat{b}, \hat{v} \right\rangle_{\mathcal{E}_h^\partial(t)} = 0, \quad (5.23c)$$

for all $(\mathbf{P}, v, \hat{v}) \in \mathbf{V}_h(\Omega_h(0)) \times V_h(\Omega_h(0)) \times \Lambda_h(\mathcal{E}_h(t))$, where $\hat{b} := \hat{u} - u_D$.

5.2.2 Linear convection equation

For a linear convection equation, we take $\mathbf{F} = \beta u$, which leads to the following ALE formulation,

$$\begin{aligned} & \left(\frac{\partial u_X}{\partial t}, v \right)_{\Omega_h(0)} - (J \mathcal{G}^{-1}(\beta u - u \mathbf{u}_G), \nabla_{\mathbf{x}} v)_{\Omega_h(0)} \\ & + \langle \mathbf{n} \cdot (\beta u - u \mathbf{u}_G) + \tau(u - \hat{u}), v \rangle_{\partial\Omega_h(t) \setminus \Gamma_D} \\ & + \langle \mathbf{n} \cdot (\beta u - u \mathbf{u}_G) + \tau(u - u_D), v \rangle_{\Gamma_D} = 0, \end{aligned} \quad (5.24a)$$

$$\langle [\mathbf{n} \cdot (\beta u - u \mathbf{u}_G) + \tau(u - \hat{u})], \hat{v} \rangle_{\mathcal{E}_h^o(t)} + \langle \hat{b}, \hat{v} \rangle_{\mathcal{E}_h^\partial(t)} = 0, \quad (5.24b)$$

where $\tau := |(\beta - \mathbf{u}_G) \cdot \mathbf{n}|$ is a stabilization parameter.

5.2.3 Linear convection-diffusion equation

For a linear convection-diffusion equation, we take $\mathbf{F} = \beta u - \kappa \nabla u$. The ALE formulation reads ⁸

$$\begin{aligned} (\mathbf{Q}_{\mathbf{x}}, \mathbf{P})_{\Omega_h(0)} + (u_X, \nabla_{\mathbf{x}} \cdot \mathbf{P})_{\Omega_h(0)} - \langle \hat{u} \mathcal{G}^T \mathbf{n}, \mathbf{P} \rangle_{\partial \Omega_h(t) \setminus \Gamma_D} \\ - \langle u_D \mathcal{G}^T \mathbf{n}, \mathbf{P} \rangle_{\Gamma_D} + (J^{-1} (\nabla_{\mathbf{x}} J) u_X, \mathbf{P})_{\Omega_h(0)} = 0, \end{aligned} \quad (5.25a)$$

$$\begin{aligned} \left(\frac{\partial u_X}{\partial t}, v \right)_{\Omega_h(0)} - (J \mathcal{G}^{-1} (\beta u - \mathbf{u}_G u - \kappa \mathbf{Q}), \nabla_{\mathbf{x}} v)_{\Omega_h(0)} \\ + \langle \mathbf{n} \cdot (\beta u - u \mathbf{u}_G) - \mathbf{n} \cdot \kappa \mathbf{Q} + \tau (u - \hat{u}), v \rangle_{\partial \Omega_h(t) \setminus \Gamma_D} \\ + \langle \mathbf{n} \cdot (\beta u - u \mathbf{u}_G) - \mathbf{n} \cdot \kappa \mathbf{Q} + \tau (u - u_D), v \rangle_{\Gamma_D} = 0, \end{aligned} \quad (5.25b)$$

$$\langle [\mathbf{n} \cdot (\beta u - u \mathbf{u}_G) - \mathbf{n} \cdot \kappa \mathbf{Q} + \tau (u - \hat{u})], \hat{v} \rangle_{\mathcal{E}_h^o(t)} + \langle \hat{b}, \hat{v} \rangle_{\mathcal{E}_h^{\partial}(t)} = 0, \quad (5.25c)$$

where τ is a stabilization parameter, here we take $\tau := |\mathbf{n} \cdot (\beta - \mathbf{u}_G)| + \kappa - \mathbf{n} \cdot (\beta - \mathbf{u}_G)$.

5.3 Sliding mesh interface

Now we are in a position to combine the two ideas, the nonconforming mesh treatment in Section 5.1 and the ALE method in Section 5.2, for simulating rotating flows. To that end, we first decompose the domain into the stationary region Ω_s and the rotating region Ω_r , as shown in Figure 5.3. Second, we describe a motion in the rotating mesh and the stationary mesh by the ALE formulation (5.15) with

$$\begin{cases} J = 1, \mathcal{G} = \mathcal{I}, \mathbf{u}_G = 0 & \text{for } \Omega_s, \\ J = 1, \mathcal{G} = \mathcal{R}, \mathbf{u}_G = \boldsymbol{\Omega} \times (\mathbf{x} - \mathbf{x}_c) & \text{for } \Omega_r, \end{cases} \quad (5.26)$$

where \mathbf{x}_c is the center of the circular rotating mesh; \mathcal{I} is the identity matrix; $\mathcal{R} = \begin{pmatrix} \cos(\Omega t) & -\sin(\Omega t) \\ \sin(\Omega t) & \cos(\Omega t) \end{pmatrix}$ is the rotation matrix; $\boldsymbol{\Omega} = (0, 0, \Omega)$ is z-axis rotation vector;

⁸ The contravariant flux is defined as $\mathbf{F}_{\mathbf{x}} = J \mathcal{G}^{-1} (\beta u - \mathbf{u}_G u) - J \mathcal{G}^{-1} \kappa \mathbf{Q}$. When we substitute \mathbf{Q} with $J^{-1} \mathcal{G}^{-T} \mathbf{Q}_{\mathbf{x}}$, we can define the inviscid and viscous contributions of the contravariant flux [74], $\mathbf{F}_{\mathbf{x}} = \mathbf{F}_{\mathbf{x}}^I - \mathbf{F}_{\mathbf{x}}^V$ with $\mathbf{F}_{\mathbf{x}}^I := J \mathcal{G}^{-1} (\beta u - \mathbf{u}_G u)$ and $\mathbf{F}_{\mathbf{x}}^V = \mathcal{G}^{-1} \kappa \mathcal{G}^{-T} \mathbf{Q}_{\mathbf{x}} = \kappa_X \mathbf{Q}_{\mathbf{x}}$.

(the reader can find the details in Appendix B regarding the connection between the rotating frame and the ALE formulation.) Third, sliding mesh interfaces (dynamically varying mortars) are constructed between the stationary and the rotating meshes at a given time of instance. Finally, we apply the nonconforming treatment to the ALE formulation (5.15) and substitute the deformation gradient, Jacobian and grid velocity with (5.26). The ALE formulation for rotating flows reads: find the approximate solution $(\mathbf{Q}, u, \hat{u}) \in \mathbf{V}_h(\Omega_h(t)) \times V_h(\Omega_h(t)) \times \Lambda_h(\mathcal{E}_h(t))$ such that

$$\begin{aligned} (\mathcal{R}^T \mathbf{Q}, \mathbf{P})_{\Omega_h(0)} + (u, \nabla_{\mathbf{x}} \cdot \mathbf{P})_{\Omega_h(0)} - \langle \hat{u} \mathcal{R}^T \mathbf{n}, \mathbf{P} \rangle_{\mathcal{E}_h^\sigma(t)} \\ - \langle u_D \mathcal{R}^T \mathbf{n}, \mathbf{P} \rangle_{\Gamma_D} = 0, \end{aligned} \quad (5.27a)$$

$$\begin{aligned} \left(\frac{\partial u}{\partial t}, v \right)_{\Omega_h(0)} - (\mathcal{R}^{-1} (\mathbf{F} - u \mathbf{u}_G), \nabla_{\mathbf{x}} v)_{\Omega_h(0)} \\ + \langle \widehat{\mathbf{F} - u \mathbf{u}_G} \cdot \mathbf{n}, v \rangle_{\mathcal{E}_h^\sigma(t)} \\ + \langle \widehat{\mathbf{F}(u, \mathbf{Q}, u_D) - u_D \mathbf{u}_G} \cdot \mathbf{n}, v \rangle_{\Gamma_D} = 0, \end{aligned} \quad (5.27b)$$

$$\langle \widehat{\mathbf{F} - u \mathbf{u}_G} \cdot \mathbf{n}, \hat{v} \rangle_{\mathcal{E}_h^\sigma(t)} + \langle \hat{b}, \hat{v} \rangle_{\mathcal{E}_h^\partial(t)} = 0, \quad (5.27c)$$

for all $(\mathbf{P}, v, \hat{v}) \in \mathbf{V}_h(\Omega_h(0)) \times V_h(\Omega_h(0)) \times \Lambda_h(\mathcal{E}_h(t))$, where $\hat{b} := \hat{u} - u_D$. Note that we have replaced the surface integrals with the mortar integrals as discussed in Section 5.1.

5.4 Numerical results

We conduct several numerical examples to demonstrate the performance of the curved nonconforming treatment, the ALE formulation, and the sliding mesh interfaces.

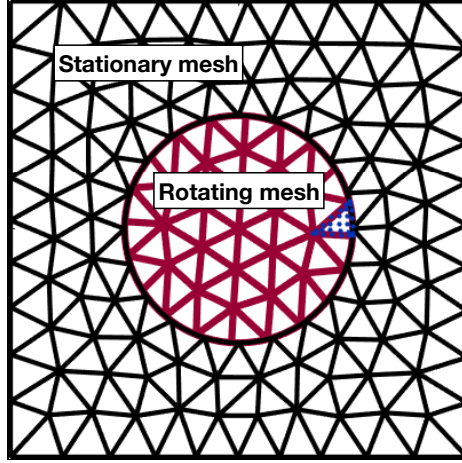


Figure 5.6: A domain is decomposed into the stationary and the rotating regions. Each region is discretized with triangle elements. Isoparametric curved elements are used. Sliding mesh interfaces are constructed between stationary and rotating meshes.

5.4.1 Steady-state advection with a discontinuous solution

This example tests the nonconforming mesh treatment for linear and curved elements. A steady-state advection equation reads

$$\nabla \cdot (\beta u) = s, \text{ in } \Omega, \quad (5.28a)$$

$$u = g_D, \text{ on } \Gamma_I, \quad (5.28b)$$

where u is the scalar quantity, β is the fluid velocity and g_D is the function defined on inflow boundary Γ_I .

The HDG discretization of (5.28a) gives the following local solver and global solver: find the approximate solution $(u, \hat{u}) \in V_h(\Omega_h) \times \Lambda_h(\mathcal{E}_h)$ such that

$$-(\beta u, \nabla v)_{\Omega_h} + \left\langle \widehat{F}_n, v \right\rangle_{\mathcal{E}_h^o \cup \mathcal{E}_h^p} = (s, v)_{\Omega_h}, \quad (5.29a)$$

$$\left\langle \left[\widehat{F}_n \right], \hat{v} \right\rangle_{\mathcal{E}_h^o} + \left\langle \hat{b}, \hat{v} \right\rangle_{\mathcal{E}_h^p} = 0, \quad (5.29b)$$

for all $(v, \hat{v}) \in V_h(\Omega_h) \times \Lambda_h(\mathcal{E}_h)$, where

$$\hat{b} := \begin{cases} \hat{u} - g_D & \text{for } \Gamma_I, \\ \hat{u} - u & \text{otherwise,} \end{cases} \quad (5.30)$$

$\widehat{F}_n = \mathbf{n} \cdot \hat{\mathbf{F}} := \mathbf{n} \cdot \boldsymbol{\beta} u + \tau (u - \hat{u})$ the upwind based HDG flux [30], and $\tau = |\boldsymbol{\beta} \cdot \mathbf{n}|$.

We consider the case with $s = 0$ and $\boldsymbol{\beta} = (1 + \sin(\pi y/2), 2)$. The domain is $\Omega \in [0, 2] \times [0, 1]$. The inflow boundary condition is applied at the left and the bottom walls by

$$g_D = \begin{cases} 1, & x = 0, 0 \leq y \leq 2, \\ \sin^6(\pi x), & 0 \leq x \leq 1, y = 0, \\ 0, & 1 \leq x \leq 2, y = 0, \end{cases}$$

and the outflow boundary condition is given at the right and the top walls. The exact solution is obtained by using the method of characteristic ⁹.

$$u = \begin{cases} 1, & x_0 \leq 0, \\ \sin^6(\pi x_0), & x_0 > 0 \end{cases}$$

with $x_0 = \frac{y}{2} - \frac{1}{\pi} \cos(\frac{\pi y}{2}) + \frac{1}{\pi}$. Figure 5.7 shows the numerical solution on a curved nonconforming mesh. Here, the circle in the mesh is represented by isoparametric elements with $k = 3$ order polynomials. To capture the sharp gradient in Figure 5.7, we adaptively refine the mesh three times.

5.4.2 Constant solution

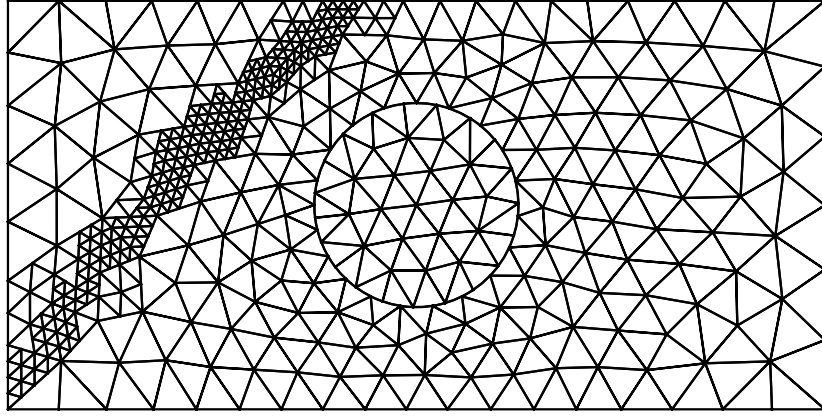
To examine the free-stream preservation property on a nonconforming mesh, we consider a constant solution $u^e = 1$ with the uniform flow of $(\beta_x, \beta_y) = (1, 0)$ in the two-dimensional domain $\Omega = (-0.5, 0.5)^2$. The DG formulation for a linear convection reads

$$\left(\frac{\partial u}{\partial t}, v \right)_{\Omega_h} - (\boldsymbol{\beta} u, \nabla v)_{\Omega_h} + \langle F_n^*, v \rangle_{\mathcal{E}_h^o \cup \mathcal{E}_h^\partial} = 0, \forall v \in V_h(\Omega_h) \quad (5.31)$$

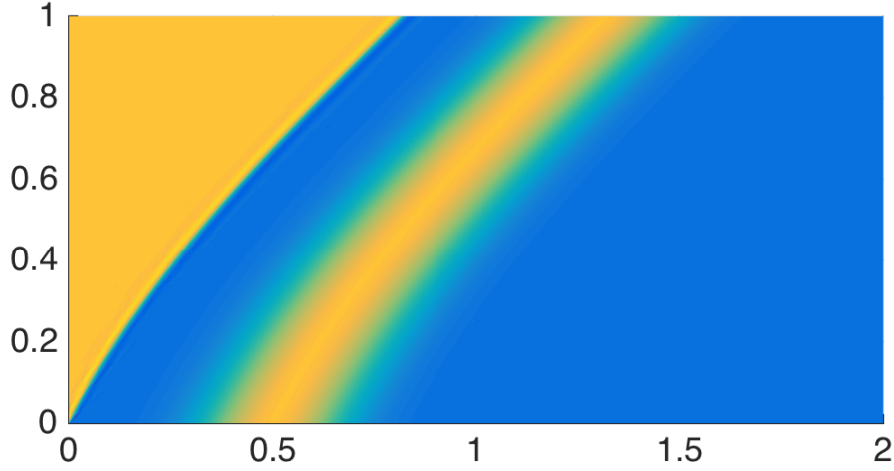
⁹ The solution u on a characteristic does not change, i.e.,

$$du = \frac{\partial u}{\partial x} dx + \frac{\partial u}{\partial y} dy = \beta_x dx + \beta_y dy = 0.$$

Using the separation of variable, we come up with $x_0 = \frac{y}{2} - \frac{1}{\pi} \cos(\frac{\pi y}{2}) + \frac{1}{\pi}$.



(a) A curved nonconforming mesh



(b) A numerical solution field

Figure 5.7: Two-dimensional steady-state transport equation with a discontinuous solution: (a) a curved nonconforming mesh and (b) the solution field, u .

where F_n^* is Lax-Friedrich numerical flux. The domain is discretized with $N_e = \{67, 268, 1072\}$ triangular elements, where the curved nonconforming mesh are constructed to be *watertight* in Figure 5.11(c). As shown in Figure 5.8 (a), the free-stream is preserved up to rounding errors for the polynomial degree of the solution $k = \{3, 4, 5, 6\}$ over a curved nonconforming mesh. We also notice that the error level increases as the degree of the polynomial increases. This is because the time

discretization error becomes dominant over the spatial discretization error for high-order polynomials. We clearly see the trend in Figure 5.8 (b), where the error level decreases as a timestep size decreases as expected.

From this example, we confirm that the watertight mesh and our nonconforming treatment satisfy the freestream preservation condition. However, there remains a challenge for a sliding mesh. Even if we start with a watertight mesh, once the mesh rotates, then we would encounter a polynomial gap between curved nonconforming elements on the sliding interface. This needs to be addressed in the future.

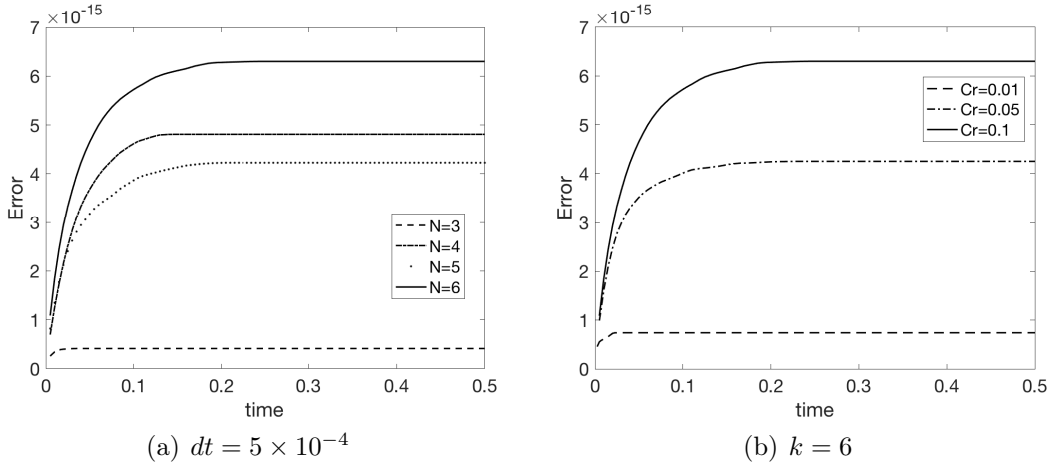


Figure 5.8: Constant solution: error histories are shown for (a) $k = \{3, 4, 5, 6\}$ polynomial order with fixed dt and for (b) $Cr = \{0.01, 0.05, 0.1\}$ with fixed $k = 6$. The freestream is preserved up to rounding error over a curved nonconforming mesh. L^2 errors are measured during $t \in [0, 0.5]$, where the 4th order RK method is used.

5.4.3 Cosine bell

We are interested in a cosine bell,

$$u = 1 + \left(\frac{0.15}{4\pi} \right)^2 (f(4\pi r_c) - f(\pi)), \quad (5.32)$$

translating with a constant velocity $\beta = (1, 0)$ for the DG advection equation. Here, $r_c = \|\mathbf{x}\|_2$ and $f(r) = 2 \cos(r) + 2r \sin(r) + \frac{1}{8} \cos(2r) + \frac{r}{4} \sin(2r) + \frac{4}{3} r^2$. The domain

$\Omega \in [0.5]^2$ is discretized with the curved nonconforming mesh in Figure 5.11(c). A periodic boundary condition is applied to all sides.

Figure 5.9 shows the history of the normalized energy and the energy loss for $t \in [0, 10]$. We observe that the numerical solution is stable and its energy loss is at the level of $\mathcal{O}(10^{-10})$. For time integration, SSPRK3 method is employed.

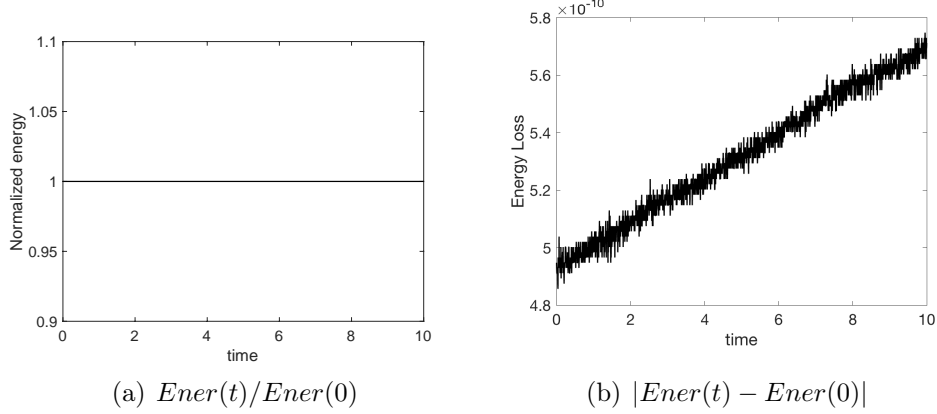


Figure 5.9: Cosine bell test: (a) energy $\frac{u^2}{2}$ history shows the scheme is stable. (b) energy loss is observed at the level of $\mathcal{O}(10^{-10})$. SSPRK3 is used for time integration from $t \in [0, 10]$.

5.4.4 Rotating Gaussian

We consider a Gaussian bell for convection and convection-diffusion equations in the (hybridized) DG spatial discretization. (Hybridized) Lax-Friedrich numerical flux is used for simulations.

The Gaussian bell is rotated with the velocity of $(u, v) = (-4y, 4x)$ in the two-dimensional domain $\Omega = (-0.5, 0.5)^2$, which is discretized with N_e triangular elements. An exact boundary condition is imposed at all boundaries. The exact solution is given by

$$q^e = \exp \left(-\frac{1}{2\sigma^2} ((\tilde{x} - x_c)^2 + (\tilde{y} - y_c)^2) \right), \quad (5.33a)$$

where $\sigma = 0.1$ is the standard deviation, $(x_c, y_c) = (-0.3, 0)$ is the center, $\tilde{x} = \cos(4t)x + \sin(4t)y$ and $\tilde{y} = -\sin(4t)x + \cos(4t)y$.

For a convection equation, Figure 5.10 quantitatively shows a rotating Gaussian bell at $t = 0$, $t = 0.02$, $t = 0.04$ and $t = 0.06$ using the 6th order polynomial degree and 4th order Runge-Kutta methods. The Gaussian bell is initially located at $\mathbf{x}_c = (-0.3, 0)$. As time goes on, the Gaussian bell rotates without changing its shape.

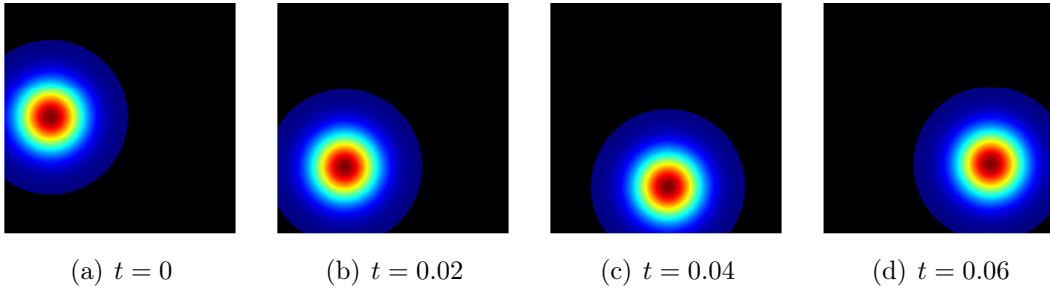


Figure 5.10: Snapshots for the rotating Gaussian bell at $t = \{0, 0.02, 0.04, 0.06\}$. The simulation was conducted with the 6th order polynomial degrees and 4th order RK method of $dt = 10^{-3}$.

5.4.4.1 Nonconforming mesh

We conduct a convergence study on linear conforming, linear nonconforming and curved nonconforming meshes in Figure 5.11 with DG methods. In particular, the curved nonconforming mesh is *watertight*, which is generated by the procedure in Subsection 5.1.3.

Both Figure 5.12 and Table 5.1 show h -convergence results in the $L^2(\Omega_h)$ -norm using a sequence of nested meshes with $N_e = \{32, 128, 512\}$ for a linear conforming mesh, $N_e = \{164, 656, 2624\}$ for a linear nonconforming mesh and $N_e = \{67, 268, 1072\}$ for a curved nonconforming mesh, respectively (here, we compute $h := (N_e)^{\frac{1}{2}}$). We observe that the DG solutions converge to the exact solution with

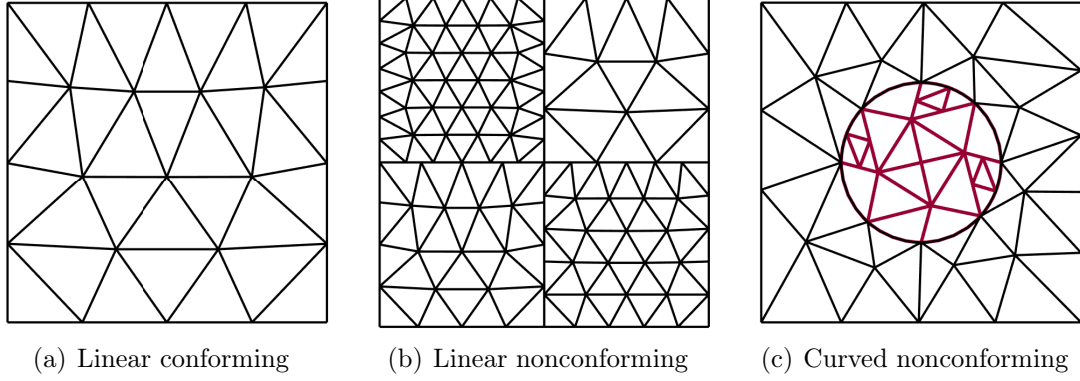


Figure 5.11: Coarse grids with (a) linear conforming, (b) linear nonconforming and (c) curved nonconforming meshes.

the convergence rate of $k + 1$ for linear conforming, linear nonconforming and curved nonconforming meshes. The results support that our nonconforming treatment maintains the high-order accuracy of the DG methods when linear or curved nonconforming meshes are involved.

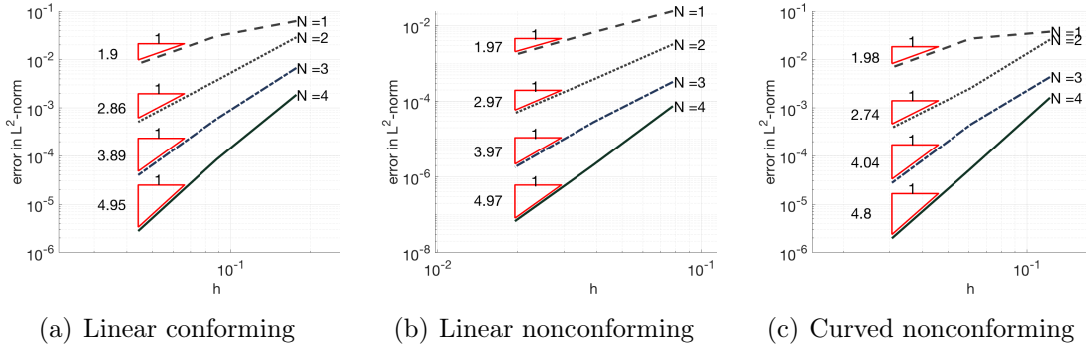


Figure 5.12: Convergence results with (a) linear conforming, (b) linear nonconforming and (c) curved nonconforming meshes. L^2 errors are measured after 100 time integrations, where the 4th order RK method is used with $dt = 10^{-6}$.

5.4.4.2 Moving mesh

We consider a Gaussian bell on a two-dimensional moving mesh. The moving mesh is described by the mapping $\mathbf{x} = \varphi(\mathbf{X})$,

Table 5.1: The rotating Gaussian test: the results show that the DG solutions converge to the exact solution with the optimal order of $k + 1$ for linear conforming, linear nonconforming and curved nonconforming meshes.

k	Linear Conforming			Linear Nonconforming			Curved Nonconforming		
	h	$\ u^e - u\ _{\Omega_h}$ error	order	h	$\ u^e - u\ _{\Omega_h}$ error	order	h	$\ u^e - u\ _{\Omega_h}$ error	order
1	0.177	6.24E-02	—	0.078	2.39E-02	—	0.122	3.76E-02	—
	0.088	3.06E-02	1.03	0.039	6.67E-03	1.84	0.061	2.71E-02	0.47
	0.044	8.18E-03	1.90	0.020	1.70E-03	1.97	0.031	6.86E-03	1.98
2	0.177	2.84E-02	—	0.078	3.20E-03	—	0.122	2.58E-02	—
	0.088	3.73E-03	2.95	0.039	3.83E-04	3.06	0.061	2.57E-03	3.33
	0.044	5.12E-04	2.86	0.020	4.88E-05	2.97	0.031	3.85E-04	2.74
3	0.177	6.70E-03	—	0.078	3.27E-04	—	0.122	4.35E-03	—
	0.088	5.95E-04	3.49	0.039	2.85E-05	3.52	0.061	4.55E-04	3.26
	0.044	4.02E-05	3.89	0.020	1.82E-06	3.97	0.031	2.77E-05	4.04
4	0.177	1.87E-03	—	0.078	7.48E-05	—	0.122	1.63E-03	—
	0.088	8.64E-05	4.43	0.039	2.11E-06	5.15	0.061	5.50E-05	4.89
	0.044	2.79E-06	4.95	0.020	6.73E-08	4.97	0.031	1.97E-06	4.80

$$x_1 = X_1 + A_1 \sin(2\pi/L_1 X_1) \sin(2\pi/L_2 X_2) \sin(2\pi/T_1 t), \quad (5.34a)$$

$$x_2 = X_2 + A_2 \sin(2\pi/L_1 X_1) \sin(2\pi/L_2 X_2) \sin(2\pi/T_2 t). \quad (5.34b)$$

We pick $T_1 = 1$, $T_2 = 0.5$, $L_1 = L_2 = 1$ and $A_1 = A_2 = 0.1$. Figure 5.13 shows the snapshots of the moving mesh at time $t = 0$, $t = 0.1$, $t = 0.2$ and $t = 0.3$.

For a convection-diffusion equation, we use the HDG formulation in (5.25). Exact boundary condition is imposed on all sides. Figure 5.14 and Figure 5.15 show the evolutions of the solution field u for $\kappa = 0.01$ and $\kappa = 0.1$, respectively. We see Gaussian bell is diffused while it rotates due to the diffusion coefficient, κ .

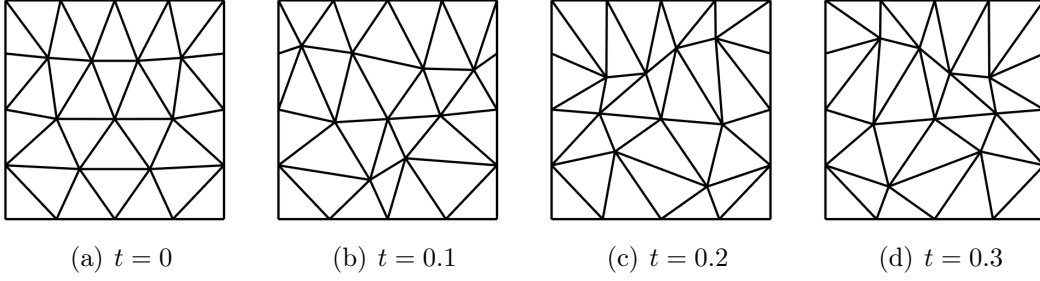


Figure 5.13: Snapshots of the moving mesh at $t = \{0, 0.1, 0.2, 0.3\}$. The mesh moves with a mapping function of (5.34).

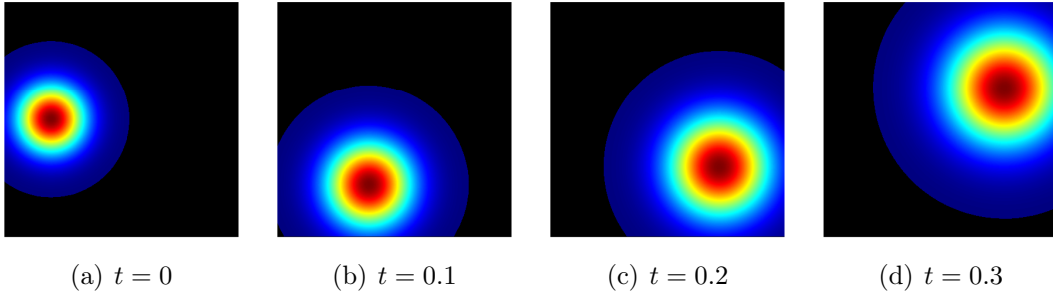


Figure 5.14: Snapshots of the rotating Gaussian bell at $t = \{0, 0.1, 0.2, 0.3\}$ with $\kappa = 0.01$.

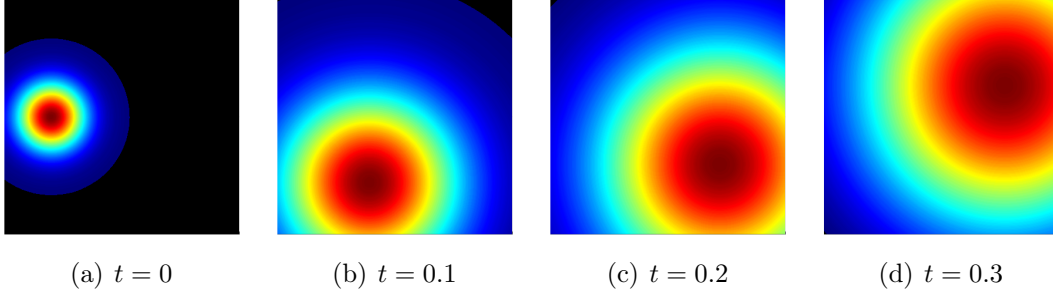


Figure 5.15: Same to Figure 5.14 except $\kappa = 0.1$.

We also perform a h -convergence study of u and \mathbf{Q} in Figure 5.16 using a sequence of nested meshes with $N_e = \{32, 128, 512\}$ elements. We observe $k + 1$ convergence rate for u , but for \mathbf{q} we see k rate for even polynomial and approximately $k + 1/2$ rate for odd polynomial orders.

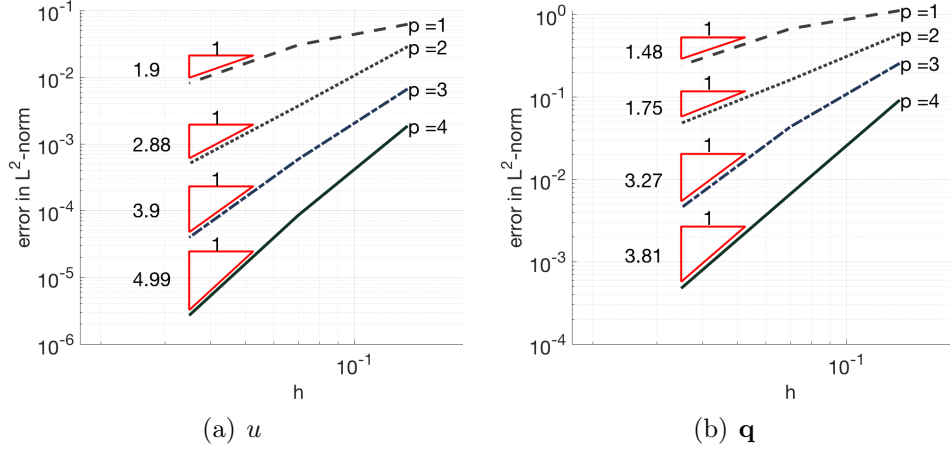


Figure 5.16: Rotating Gaussian with moving mesh: h convergence study for (a) u and (b) q . Backward Euler method is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$ for $\kappa = 0.01$. (Here, p stands for the solution order.)

5.4.4.3 Rotating mesh

We consider a Gaussian bell on a two-dimensional rotating mesh as shown in Figure 5.3. The domain is decomposed into the stationary and the rotating regions. The rotating mapping is prescribed as

$$\mathbf{x} = \mathbf{x}_c + \mathcal{R}(\Omega t)\mathbf{X} \quad (5.35)$$

with $\mathcal{R}(\theta) \begin{pmatrix} \cos(\Omega t) & -\sin(\Omega t) \\ \sin(\Omega t) & \cos(\Omega t) \end{pmatrix}$ the rotation matrix and a constant angular velocity Ω . Figure 5.17 shows the rotating meshes at $t = 0$, $t = 0.1$, $t = 0.2$ and $t = 0.3$ with $\Omega = 2\pi$ and $\mathbf{x}_c = (0, 0)$.

For a convection-diffusion equation, the HDG formulation is used on the rotating mesh in 5.17. We perform a h -convergence study of u and \mathbf{Q} using a sequence of nested meshes with $N_e = \{58, 232, 928\}$ elements. We summarize the results in Figure 5.16. The solution u converges to the exact solution with $k + 1$ rate, whereas \mathbf{q} shows k rate for even polynomial and $k + 1/2$ rate for odd polynomial orders.

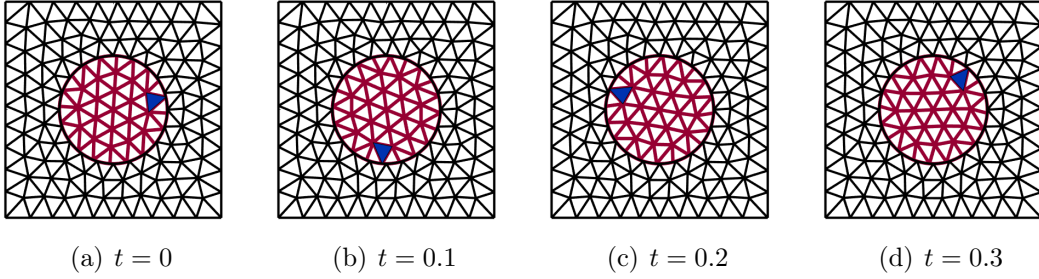


Figure 5.17: Snapshots of the rotating mesh at $t = \{0, 0.1, 0.2, 0.3\}$.

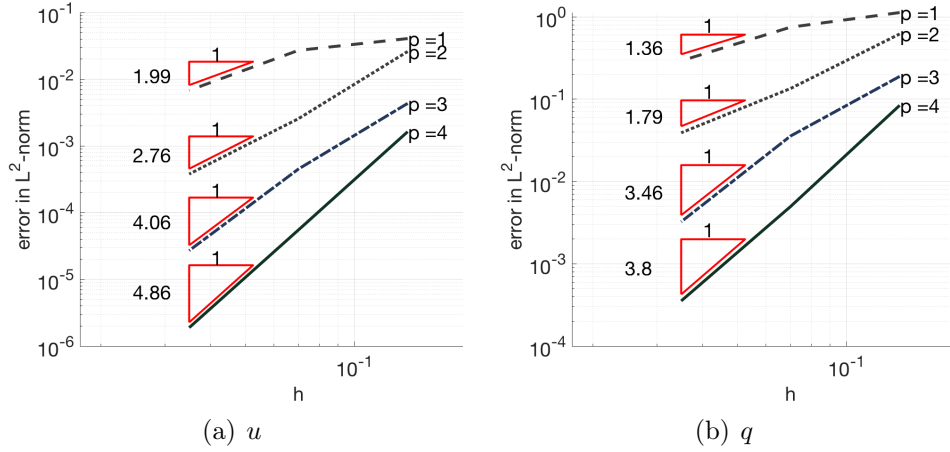


Figure 5.18: Rotating Gaussian with a sliding mesh: h -convergence study for (a) u and (b) q . Backward Euler method is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$ for $\kappa = 0.01$. (Here, p stands for the solution order.)

5.4.5 Isentropic vortex

We consider the isentropic vortex example [234] for Euler equations. The DG formulation is used for the simulation. The domain is $\Omega = [3.5, 5.5] \times [-1, 1]$, which is descirized with triangular elements. Exact boundary condition is imposed at all directions. The center of vortex is located at $\mathbf{x}_c = (4.5, 0)$ at $t = 0$, and translated with the velocity $(0, 1)$ on the rotating mesh in Figure 5.19. We perform the simulation for $t \in [0, 0.18]$ with the $6th$ order polynomial degrees and the $4th$ order RK method of $dt = 10^{-3}$. The density ρ and the y -momentum ρv are plotted in Figure 5.20 and

Figure 5.21. We observe that the fields of the vortex is advected upward without changing its shape.

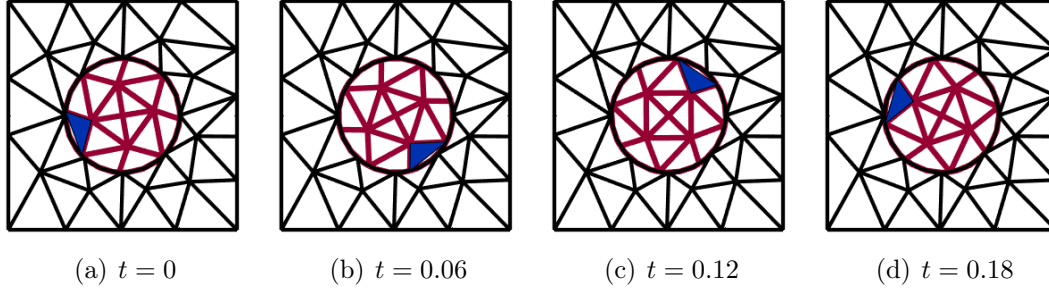


Figure 5.19: Snapshots of the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$. The domain is decomposed into the rotating mesh (red) and the stationary mesh (black). The blue element is highlighted to emphasize the rotation.

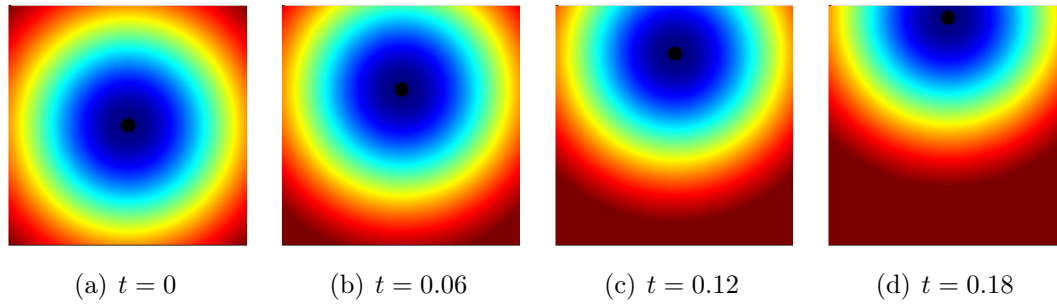


Figure 5.20: Snapshots of ρ for the isentropic vortex on the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$. The simulation was conducted with the 6th order polynomial degrees and the 4th order RK method of $dt = 10^{-3}$.

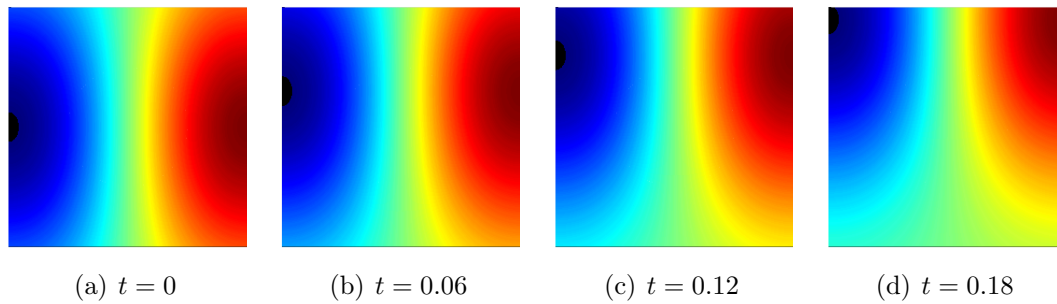


Figure 5.21: Snapshots of ρv for the isentropic vortex on the rotating mesh at $t = \{0, 0.06, 0.12, 0.18\}$.

We perform a h -convergence test in Figure 5.22. We see that all the conservative variables (ρ , ρu , ρv and ρE) converge to the exact solution with a higher rate than the optimal one of $(k + \frac{1}{2})$ for a hyperbolic problem.

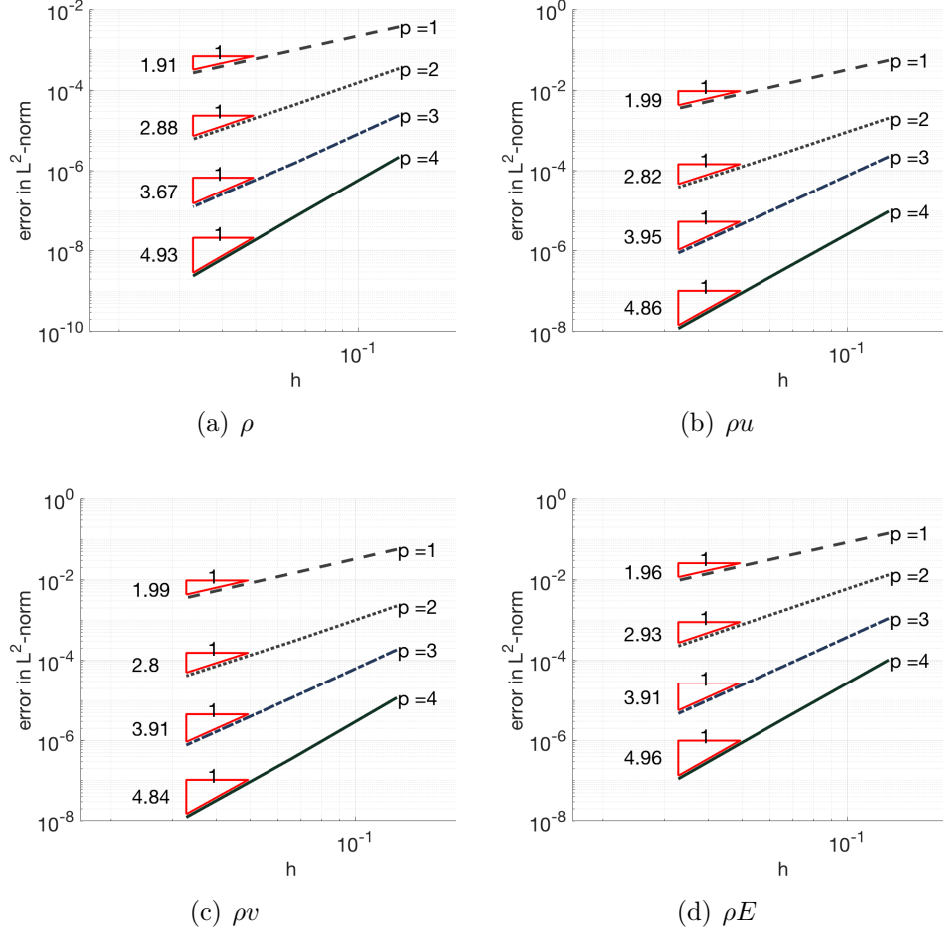


Figure 5.22: Isentropic vortex with a sliding mesh: h -convergence study for (a) ρ , (b) ρu , (c) ρv and (d) ρE . RK4 is used with $dt = 10^{-6}$ and error is measured at $t = 10^{-4}$. (Here, p stands for the solution order.)

5.5 Discussions

In this chapter, we have developed the sliding-mesh interface in the (hybridized) DG spatial discretization for simulating rotating flows. The sliding-mesh

interface is composed of the nonconforming treatment and the ALE formulation. For an accurate and stable nonconforming treatment, we have employed mortar spaces in the (hybridized) DG spatial discretization. We first develop the projection operator from a face to a mortar. Then, we project state variables on the face to the mortar spaces, and compute the numerical flux. We interpret the surface integrals in the local solver as the sum of its surrounding mortar integrals. With this standpoint, we do not need to project back the numerical flux from a mortar to its adjacent faces. Meanwhile, as for the DG convection equation, we confirm that the watertight mesh and the nonconforming treatment satisfies the freestream preservation condition. We also numerically observe our nonconforming treatment for curved elements is stable.

Next, we extend the nonconforming treatment to rotating mesh. Since we construct a mortar interface such that only two adjacent elements are allowed, mortar spaces need to be updated at every time instance. Once we identify all mortar spaces, then we use the fixed nonconforming treatment to communicate between the stationary and the rotating meshes. Numerical results demonstrate that our (hybridized) DG sliding-mesh method achieves high-order convergence rates for advection, convection-diffusion, and Euler equations.

Chapter 6

Conclusions and Future Work

In this work, we have developed high-order (hybridized) DG methods for several equations including a linear degenerate elliptic equation, shallow water equations, Euler equations, and Navier-Stokes equations. At first, we propose the HDG method for a linear degenerate elliptic equation, where the scaled system [7] is utilized to handle the degeneracy. We show that the proposed HDG method is well-posed by using an energy approach. We derive *a priori* error estimates for the method on simplicial meshes in both two- and three-dimensions, and demonstrate that the convergence rates are optimal for both the scaled pressure and the scaled velocity for non-degenerate problems and are sub-optimal by half order for degenerate ones via numerical experiments. We also enhance the HDG solutions by post-processing. The super-convergence rates of $(k+2)$ and $(k+\frac{3}{2})$ are observed for both a non-degenerate case and a degenerate case away from the degeneracy. Degenerate problems with low regularity solutions are also studied, and numerical results show that high-order methods are beneficial in terms of accuracy.

To achieve high-order accuracy in time and space, in addition to high-order (hybridized) DG spatial discretization, high-order time integrators are also needed. Since explicit time integration methods sometimes suffer from a timestep size restriction for multiscale problems, we propose the IMEX HDG-DG framework for efficient flow simulation. We split the governing system into a stiff part describing the fast waves and a nonstiff part associated with the slow modes. Then, we discretize

the former implicitly with the HDG method while the latter explicitly with the DG method. The coupled HDG-DG spatial discretization combined with implicit-explicit (IMEX) time integrators promote high-order solutions in time and space, and allow larger timestep sizes than explicit time integrators. The IMEX HDG-DG scheme only requires one linear solve per time stage and generates smaller and sparser linear systems relative to DG. Numerical results for shallow water equations, Euler equations, Navier-Stokes equations demonstrate that our methods are beneficial for applications where slow modes are accurately treated while fast modes are inaccurately handled.

We found that the IMEX HDG-DG method suppresses the high-frequency mode in the system to maintain numerical stability, but provides a less accurate solution in the high-frequency mode. Also, the performance of the method highly depends on a linear solver. Therefore, it is difficult to show excellent scalability for parallel computation. To avoid the difficulties of the IMEX HDG-DG scheme, we have developed the EXPONENTIAL DG method. The EXPONENTIAL DG method combines exponential time integrators and the DG spatial discretization. First, we construct linear and nonlinear operators by adding and subtracting a linear flux based on a flux Jacobian to the governing equation at a continuous level. Then the system is discretized with the DG method. Since the exponential integrator does not require any linear solve, it is attractive in parallel computation, but a matrix exponential needs to be evaluated. To compute a matrix exponential accurately and efficiently, we use the adaptive Krylov subspace method [162] and its variants [82, 133]. The parallel implementation of the algorithm involves matrix-vector and vector-vector multiplications. The matrix-vector product is performed in a matrix-free manner and is embarrassingly parallel by the virtue of the DG spatial discretization. The vector-vector product (dot product) requires global communication, but the communication cost is relatively small compared to the computational cost. From the weak and

the strong scaling tests, we observe that the EXPONENTIAL DG method provides promising scalable results for the Euler systems.

Finally, we have proposed the (H) DG sliding-mesh methods for rotating flows. The main ingredients are mortar interfaces and the Arbitrary Lagrangian-Eulerian (ALE) formulation. We decompose a domain into rotating and stationary regions. The ALE formulation represents the motion in rotating and stationary meshes. The remaining job is to handle the curved nonconforming mesh between the rotating and the stationary meshes. Inspired by the HDG method, we use the conservation (transmission) condition of a numerical flux as an interface condition between two regions. To that end, we introduce a mortar interface between the regions, project the state variables from the faces of the adjacent elements to the mortar, and then we compute the numerical flux on the mortar interface. In addition, we interpret the surface integrals in the local solver as the sum of its surrounding mortar integrals. With this point, we do not need to project back the numerical flux from the mortar to its adjacent elements. Numerical examples show that our nonconforming treatment is stable and high-order accurate. By incorporating the nonconforming treatment to the ALE formulation, we develop the (H) DG sliding-mesh interfaces. Numerical experiments show that our method achieves high-order convergence rates for advection, convection-diffusion, and Euler equations.

There are several directions in which the current work can be extended and we suggest a few of them as follows.

- **Extension of the HDG method for a linear degenerate elliptic equation:** We have developed the HDG system for a linear degenerate elliptic equation, which is a part of Darcy-Stokes systems. The authors [6] have developed the full set of dynamical equations, (i.e., (2.1), (2.4), (2.7) and (2.8)) with the

mixed finite element method and demonstrated the benefit of the scaled system. Thus, the next step of our work would be to develop an HDG system for the full model. Another possible direction is to study the case where the degeneracies lie on a set of measure nonzero. Recently, the authors [230] have proposed a shock tracking algorithm, where they adjust a mesh such that shock interfaces align a set of measure zero by solving an optimization problem. We may utilize the method to identify the interfaces between matrix solid and fluid melt.

- **Extension of the IMEX HDG-DG scheme:**

IMEX methods are designed to efficiently handle the stiffness in the system. There are two types of stiffness: scale-separation stiffness and geometry-induced stiffness. In this work, we focus on subcritical flows for shallow water equations and low Mach number flows for Euler equations, which means that scale-separation stiffness is of interest. Thus, another possible direction is to tackle geometry-induced stiffness using the IMEX HDG-DG scheme. The geometry-induced stiffness can be seen when adaptive mesh refinement (AMR) is employed, complex geometry is triangulated, or an anisotropic model is at hand. For example, in a global atmospheric model, the horizontal length scale is larger than the vertical length scale. To efficiently treat the stiffness, horizontal-explicit vertically-implicit (HEVI) methods have been studied [220]. It would be worth to study whether the IMEX HDG-DG schemes well fit this kind of problem. Regarding the performance, preconditioned iterative methods, e.g., multilevel or multigrid methods, also need to be explored because the computational cost highly depends on a linear solver.

- **Improvement of EXPONENTIAL DG:** In this study, we compute a matrix exponential using the adaptive Krylov subspace method. However, the Arnoldi

iteration has communication bottlenecks in parallel computation because the dot product requires global communication. To amend the problem, we can attempt other strategies such as the communication-avoiding Krylov subspace methods [102], block Krylov subspace method [112] and so on. Also because the adaptive Krylov methods rely on the *a posteriori* error estimation, a sharp estimation could help the convergence of the scheme. We also want to mention that the EXPONENTIAL DG scheme is not specific to Euler systems. It will be interesting to apply the scheme to large-scale problems such as climate systems and magnetohydrodynamic (MHD) systems.

- **Applications of (H) DG sliding-mesh interfaces:** In this work, we propose the (H) DG sliding-mesh interfaces for simulating rotating flows. We have numerically shown that our scheme provides stable and high-order accurate solutions, but have not yet demonstrated its performance in real applications such as a wind turbine or a jet engine. Thus it still remains to perform a large-scale simulation with the sliding-mesh interface and show the benefit of the method. Another interesting direction is to pursue a theoretical analysis of the nonconforming mesh treatment. Along the line, constructing watertight mesh with a sliding-mesh interface is an open question.

Appendices

Appendix A

Flux Jacobian of Euler equations

For the completeness, we describe flux Jacobian of Euler equations in two- and three-dimensions. Details can be found in [139].

A.1 Two-dimensional Euler equation

We consider the two-dimensional Euler equations in a component form,

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0, \quad (\text{A.1a})$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial}{\partial x}(\rho u u + p) + \frac{\partial}{\partial y}(\rho v u) = 0, \quad (\text{A.1b})$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial}{\partial x}(\rho u v) + \frac{\partial}{\partial y}(\rho v v + p) = 0, \quad (\text{A.1c})$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x}(\rho u H) + \frac{\partial}{\partial y}(\rho v H) = 0. \quad (\text{A.1d})$$

The flux Jacobian along x -direction is defined as

$$\begin{aligned} A_x(q) &:= \left(\frac{\partial F_x(q)}{\partial q} \right) \\ &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - u^2 & (3-\gamma)u & (1-\gamma)v & (\gamma-1) \\ -uv & v & u & 0 \\ \left(\left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - H\right) u & H + (1-\gamma)u^2 & (1-\gamma)uv & \gamma u \end{pmatrix}. \end{aligned}$$

The flux Jacobian along y -direction is denoted by

$$A_y(q) := \left(\frac{\partial F_y(q)}{\partial q} \right) = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - v^2 & (1-\gamma)u & (3-\gamma)v & (\gamma-1) \\ \left(\left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - H\right) v & (1-\gamma)uv & H + (1-\gamma)v^2 & \gamma v \end{pmatrix}.$$

The normal flux Jacobian reads

$$A(q) := n_x A_x(q) + n_y A_y(q) = \begin{pmatrix} 0 & n_x & n_y & 0 \\ \phi n_x - uu_n & un_x - \tilde{\gamma} un_x + u_n & un_y - \tilde{\gamma} vn_x & -\tilde{\gamma} n_x \\ \phi n_y - vu_n & vn_x - \tilde{\gamma} un_y & vn_y - \tilde{\gamma} vn_y + u_n & -\tilde{\gamma} n_y \\ (\phi - H) u_n & Hn_x - \tilde{\gamma} uu_n & Hn_y - \tilde{\gamma} vu_n & \gamma u_n \end{pmatrix}.$$

where $\phi := \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2$ and $\tilde{\gamma} := \gamma - 1$.

The eigenvalues and eigenvectors of normal flux Jacobian, $A = R\Lambda R^{-1}$ are

$$\Lambda = \begin{pmatrix} u_n - a & 0 & 0 & 0 \\ 0 & u_n & 0 & 0 \\ 0 & 0 & u_n & 0 \\ 0 & 0 & 0 & u_n + a \end{pmatrix},$$

$$R = \begin{pmatrix} 1 & 1 & 0 & 1 \\ u - an_x & u & t_x & u + an_x \\ v - an_y & v & t_y & v + an_y \\ H - au_n & \frac{1}{2} \|\mathbf{u}\|^2 & u_t & H + au_n \end{pmatrix}$$

and

$$R^{-1} = \begin{pmatrix} \frac{1}{2} \left(\frac{1}{2} \left(\frac{\gamma-1}{a^2} \right) \|\mathbf{u}\|^2 + \frac{u_n}{a} \right) & -\frac{1}{2} \left(\left(\frac{\gamma-1}{a^2} \right) u + \frac{n_x}{a} \right) & -\frac{1}{2} \left(\left(\frac{\gamma-1}{a^2} \right) v + \frac{n_y}{a} \right) & \frac{1}{2} \left(\frac{\gamma-1}{a^2} \right) \\ 1 - \frac{1}{2} \left(\frac{\gamma-1}{a^2} \right) \|\mathbf{u}\|^2 & \left(\frac{\gamma-1}{a^2} \right) u & \left(\frac{\gamma-1}{a^2} \right) v & -\left(\frac{\gamma-1}{a^2} \right) \\ -u_t & t_x & t_y & 0 \\ \frac{1}{2} \left(\frac{1}{2} \left(\frac{\gamma-1}{a^2} \right) \|\mathbf{u}\|^2 - \frac{u_n}{a} \right) & -\frac{1}{2} \left(\left(\frac{\gamma-1}{a^2} \right) u - \frac{n_x}{a} \right) & -\frac{1}{2} \left(\left(\frac{\gamma-1}{a^2} \right) v - \frac{n_y}{a} \right) & \frac{1}{2} \left(\frac{\gamma-1}{a^2} \right) \end{pmatrix}.$$

A.2 Three-dimensional Euler equation

Consider the three-dimensional Euler equations in a component form,

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) + \frac{\partial}{\partial z}(\rho w) = 0, \quad (\text{A.2a})$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial}{\partial x}(\rho u u + p) + \frac{\partial}{\partial y}(\rho v u) + \frac{\partial}{\partial z}(\rho w u) = 0, \quad (\text{A.2b})$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial}{\partial x}(\rho u v) + \frac{\partial}{\partial y}(\rho v v + p) + \frac{\partial}{\partial z}(\rho w v) = 0, \quad (\text{A.2c})$$

$$\frac{\partial \rho w}{\partial t} + \frac{\partial}{\partial x}(\rho u w) + \frac{\partial}{\partial y}(\rho v w) + \frac{\partial}{\partial z}(\rho w w + p) = 0, \quad (\text{A.2d})$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x}(\rho u H) + \frac{\partial}{\partial y}(\rho v H) + \frac{\partial}{\partial z}(\rho w H) = 0. \quad (\text{A.2e})$$

The flux Jacobians for Euler equations along x, y, z direction and normal flux Jacobian, $A := n_x A_x(q) + n_y A_y(q) + n_z A_z(q)$, are defined as follows:

$$A_x := \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - u^2 & (3-\gamma)u & (1-\gamma)v & (1-\gamma)w & (\gamma-1) \\ -uv & v & u & 0 & 0 \\ -uw & w & 0 & u & 0 \\ \left(\left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - H\right) u & H + (1-\gamma)u^2 & (1-\gamma)uv & (1-\gamma)uw & \gamma u \end{pmatrix},$$

$$A_y := \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ -vu & v & u & 0 & 0 \\ \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - v^2 & (1-\gamma)u & (3-\gamma)v & (1-\gamma)w & (\gamma-1) \\ -vw & 0 & w & v & 0 \\ \left(\left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - H\right) v & (1-\gamma)vu & H + (1-\gamma)v^2 & (1-\gamma)vw & \gamma v \end{pmatrix},$$

$$A_z := \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ -wu & w & 0 & u & 0 \\ -wv & 0 & w & v & 0 \\ \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - w^2 & (1-\gamma)u & (1-\gamma)v & (3-\gamma)w & (\gamma-1) \\ \left(\left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2 - H\right) w & (1-\gamma)wu & (1-\gamma)wv & H + (1-\gamma)w^2 & \gamma w \end{pmatrix},$$

and

$$A = \begin{pmatrix} 0 & n_x & n_y & n_z & 0 \\ \phi n_x - uu_n & un_x - \tilde{\gamma}un_x + u_n & un_y - \tilde{\gamma}vn_x & un_z - \tilde{\gamma}wn_x & \tilde{\gamma}n_x \\ \phi n_y - vu_n & vn_x - \tilde{\gamma}un_y & vn_y - \tilde{\gamma}vn_y + u_n & vn_z - \tilde{\gamma}wn_y & \tilde{\gamma}n_y \\ \phi n_z - wu_n & wn_x - \tilde{\gamma}un_z & wn_y - \tilde{\gamma}vn_z & wn_z - \tilde{\gamma}wn_z + u_n & \tilde{\gamma}n_z \\ (\phi - H)u_n & Hn_x - \tilde{\gamma}uu_n & Hn_y - \tilde{\gamma}vu_n & Hn_z - \tilde{\gamma}wu_n & \gamma u_n \end{pmatrix}$$

where $\phi := \left(\frac{\gamma-1}{2}\right) \|\mathbf{u}\|^2$ and $\tilde{\gamma} := \gamma - 1$.

The eigenvalues and eigenvectors of $A = R\Lambda R^{-1}$ reads

$$\Lambda = \begin{pmatrix} u_n - a & 0 & 0 & 0 & 0 \\ 0 & u_n & 0 & 0 & 0 \\ 0 & 0 & u_n + a & 0 & 0 \\ 0 & 0 & 0 & u_n & 0 \\ 0 & 0 & 0 & 0 & u_n \end{pmatrix},$$

$$R = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ u - an_x & u & u + an_x & t_x & s_x \\ v - an_y & v & v + an_y & t_y & s_y \\ w - an_z & w & v + an_y & t_z & s_z \\ H - au_n & \frac{1}{2} \|\mathbf{u}\|^2 & H + au_n & u_t & u_s \end{pmatrix}$$

and

$$R^{-1} = \begin{pmatrix} \frac{\tilde{\gamma}}{4} M^2 + \frac{u_n}{2a} & -\frac{\tilde{\gamma}u}{2a^2} - \frac{n_x}{2a} & -\frac{\tilde{\gamma}v}{2a^2} - \frac{n_y}{2a} & -\frac{\tilde{\gamma}w}{2a^2} - \frac{n_z}{2a} & \frac{\tilde{\gamma}}{2a^2} \\ 1 - \frac{\tilde{\gamma}}{2} M^2 & \frac{\tilde{\gamma}u}{a^2} & \frac{\tilde{\gamma}v}{a^2} & \frac{\tilde{\gamma}w}{a^2} & -\frac{\tilde{\gamma}}{a^2} \\ \frac{\tilde{\gamma}}{4} M^2 - \frac{u_n}{2a} & -\frac{\tilde{\gamma}u}{2a^2} + \frac{n_x}{2a} & -\frac{\tilde{\gamma}v}{2a^2} + \frac{n_y}{2a} & -\frac{\tilde{\gamma}w}{2a^2} + \frac{n_z}{2a} & \frac{\tilde{\gamma}}{2a^2} \\ -u_t & t_x & t_y & t_z & 0 \\ -u_s & s_x & s_y & s_z & 0 \end{pmatrix}$$

where $M^2 := \frac{\|\mathbf{u}\|^2}{a^2}$; \mathbf{t} and \mathbf{s} are tangent vectors such that $\mathbf{s} \times \mathbf{t} = \mathbf{n}$, $\mathbf{t} \times \mathbf{n} = \mathbf{s}$ and $\mathbf{n} \times \mathbf{s} = \mathbf{t}$; $u_t = \mathbf{u} \cdot \mathbf{t}$ and $u_s = \mathbf{u} \cdot \mathbf{s}$.

We also define

$$|A| = R|\Lambda|R^{-1} = \sum_{i=1}^5 |\lambda_i| \mathbf{r}_i \mathbf{r}_i^{-T}.$$

We notice that the sum of $\mathbf{r}_4 \mathbf{r}_4^{-T} + \mathbf{r}_5 \mathbf{r}_5^{-T}$ can be denoted as

$$\begin{pmatrix} 0 & \mathbf{0}^T & 0 \\ u_n \mathbf{n} - \mathbf{u} & \mathcal{J} - \mathbf{n} \otimes \mathbf{n} & 0 \\ u_n^2 - \|\mathbf{u}\|^2 & \mathbf{u}^T - u_n \mathbf{n}^T & 0 \end{pmatrix}$$

in terms of a normal vector \mathbf{n} . Thus, $|A|$ can be rewritten as

$$\begin{aligned}
|A| = & |\lambda_1| \begin{pmatrix} \frac{\tilde{\gamma}}{4}M^2 + \frac{u_n}{2a} & -\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} - \frac{\mathbf{n}^T}{2a} & \frac{\tilde{\gamma}}{2a^2} \\ (\mathbf{u} - a\mathbf{n}) \left(\frac{\tilde{\gamma}}{4}M^2 + \frac{u_n}{2a} \right) & (\mathbf{u} - a\mathbf{n}) \otimes \left(-\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} - \frac{\mathbf{n}^T}{2a} \right) & (\mathbf{u} - a\mathbf{n}) \frac{\tilde{\gamma}}{2a^2} \\ (H - au_n) \left(\frac{\tilde{\gamma}}{4}M^2 + \frac{u_n}{2a} \right) & (H - au_n) \otimes \left(-\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} - \frac{\mathbf{n}^T}{2a} \right) & (H - au_n) \frac{\tilde{\gamma}}{2a^2} \end{pmatrix} \\
& + |\lambda_2| \begin{pmatrix} 1 - \frac{\tilde{\gamma}}{2}M^2 & \frac{\tilde{\gamma}}{a^2}\mathbf{u}^T & -\frac{\tilde{\gamma}}{a^2} \\ u_n\mathbf{n} - \frac{\tilde{\gamma}}{2}M^2\mathbf{u} & \frac{\tilde{\gamma}}{a^2}\mathbf{u} \otimes \mathbf{u} + \mathcal{J} - \mathbf{n} \otimes \mathbf{n} & -\frac{\tilde{\gamma}}{a^2}\mathbf{u} \\ u_n^2 - \frac{\|\mathbf{u}\|^2}{2} \left(1 + \frac{\tilde{\gamma}}{2}M^2 \right) & \left(1 + \frac{\tilde{\gamma}}{2}M^2 \right) \mathbf{u}^T - u_n\mathbf{n}^T & -\frac{\tilde{\gamma}}{2}M^2 \end{pmatrix} \\
& + |\lambda_3| \begin{pmatrix} \frac{\tilde{\gamma}}{4}M^2 - \frac{u_n}{2a} & -\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} + \frac{\mathbf{n}^T}{2a} & \frac{\tilde{\gamma}}{2a^2} \\ (\mathbf{u} + a\mathbf{n}) \left(\frac{\tilde{\gamma}}{4}M^2 - \frac{u_n}{2a} \right) & (\mathbf{u} + a\mathbf{n}) \otimes \left(-\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} + \frac{\mathbf{n}^T}{2a} \right) & (\mathbf{u} + a\mathbf{n}) \frac{\tilde{\gamma}}{2a^2} \\ (H + au_n) \left(\frac{\tilde{\gamma}}{4}M^2 - \frac{u_n}{2a} \right) & (H + au_n) \otimes \left(-\frac{\tilde{\gamma}\mathbf{u}^T}{2a^2} + \frac{\mathbf{n}^T}{2a} \right) & (H + au_n) \frac{\tilde{\gamma}}{2a^2} \end{pmatrix}
\end{aligned}$$

with $\lambda_1 = u_n - a$, $\lambda_2 = u_n$ and $\lambda_3 = u_n + a$.

Appendix B

Solid body rotation

This appendix shows how the solid body rotation can be described in the ALE formulation (5.15). Suppose a rotating disk embedded in stationary domain. For simplicity, we focus on a linear transport equation,

$$\frac{\partial u}{\partial t} + \nabla \cdot (\boldsymbol{\beta} u) = 0, \quad \text{in } \vartheta(t), \quad (\text{B.1})$$

where $\boldsymbol{\beta}$ is the velocity and u is the scalar quantity. The velocity $\boldsymbol{\beta}$ in the inertial frame can be decomposed into the velocity $\boldsymbol{\beta}_r$ in the rotating frame and the velocity \mathbf{u}_G of the rotating frame, i.e.,

$$\boldsymbol{\beta} = \boldsymbol{\beta}_r + \underbrace{\boldsymbol{\Omega} \times \mathbf{x}}_{\mathbf{u}_G}.$$

The governing equation in the rotating frame is written as

$$\frac{\partial u}{\partial t} + \nabla_{rot} \cdot (\boldsymbol{\beta}_r u) = 0, \quad \text{in } \vartheta(t)_{rot}, \quad (\text{B.2})$$

where ∇_{rot} is a divergence operator defined in the rotating frame. We can rewrite (B.2) in the inertia frame ¹,

$$\frac{\partial u}{\partial t} + \nabla \cdot (\mathcal{R}^{-1}(\theta) (\boldsymbol{\beta} u - u \mathbf{u}_G)) = 0, \quad \text{in } \vartheta(t) \quad (\text{B.3})$$

with $\theta = \Omega t$.

¹ We can use a chain rule to show $\nabla_{rot} \cdot \boldsymbol{\beta}_r = \nabla \cdot (\mathcal{R}^{-1} \boldsymbol{\beta}_r)$.

Note that (B.3) can be expressed in the ALE formulation (5.15) with a grid velocity ² $\mathbf{u}_G = \mathbf{\Omega} \times \mathbf{x}$, a deformation gradient $\mathcal{G} = \mathcal{R}$ and Jacobian $J = \det(\mathcal{G}) = 1$. Similarly, (B.1) is equivalent to (5.15) with $\mathbf{u}_G = 0$, $\mathcal{G} = \mathcal{J}$ and $J = \det(\mathcal{G}) = 1$.

² For example, in two-dimension, $\mathbf{u}_G = (u_G, v_G) = \Omega(-y + y_o, x - x_o)$ with the center of circle (x_o, y_o) .

Bibliography

- [1] Y ABOUSLEIMAN, AH-D CHENG, L CUI, E DETOURNAY, AND J-C ROEGIERS, *Mandel's problem revisited*, Geotechnique, 46 (1996), pp. 187–195.
- [2] MARTIN AFANASJEW, MICHAEL EIERMANN, OLIVER G ERNST, AND STEFAN GÜTTEL, *Implementation of a restarted Krylov subspace method for the evaluation of matrix functions*, Linear Algebra and its applications, 429 (2008), pp. 2293–2314.
- [3] NASH'AT AHMAD AND JOHN LINDEMAN, *Euler solutions using flux-based wave decomposition*, International Journal for Numerical Methods in Fluids, 54 (2007), pp. 47–72.
- [4] V. AIZINGER AND C. DAWSON, *A discontinuous Galerkin method for two-dimensional flow and transport in shallow water*, Advances in Water Resources, 25 (2002), pp. 67–84.
- [5] AWAD H AL-MOHY AND NICHOLAS J HIGHAM, *A new scaling and squaring algorithm for the matrix exponential*, SIAM Journal on Matrix Analysis and Applications, 31 (2009), pp. 970–989.
- [6] TODD ARBOGAST, MARC A HESSE, AND ABRAHAM L TAICHER, *Mixed methods for two-phase Darcy–Stokes mixtures of partially melted materials with regions of zero porosity*, SIAM Journal on Scientific Computing, 39 (2017), pp. B375–B402.

- [7] TODD ARBOGAST AND ABRAHAM L TAICHER, *A linear degenerate elliptic equation arising from two-phase mixtures*, SIAM Journal on Numerical Analysis, 54 (2016), pp. 3105–3122.
- [8] ———, *A cell-centered finite difference method for a degenerate elliptic equation arising from two-phase mixtures*, Computational Geosciences, 21 (2017), pp. 700–712.
- [9] DOUGLAS N ARNOLD, *An interior penalty finite element method with discontinuous elements*, SIAM journal on numerical analysis, 19 (1982), pp. 742–760.
- [10] DOUGLAS N ARNOLD, FRANCO BREZZI, BERNARDO COCKBURN, AND L DONATELLA MARINI, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM journal on numerical analysis, 39 (2002), pp. 1749–1779.
- [11] WALTER EDWIN ARNOLDI, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quarterly of applied mathematics, 9 (1951), pp. 17–29.
- [12] URI M ASCHER, STEVEN J RUUTH, AND RAYMOND J SPITERI, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Applied Numerical Mathematics, 25 (1997), pp. 151–167.
- [13] ANDY ASCHWANDEN, ED BUELER, CONSTANTINE KHROULEV, AND HEINZ BLATTER, *An enthalpy formulation for glaciers and ice sheets*, Journal of Glaciology, 58 (2012), pp. 441–457.
- [14] FRANCESCO BASSI, ANDREA CRIVELLINI, STEFANO REBAY, AND MARCO SAVINI, *Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k - ω turbulence model equations*, Computers and Fluids, 34 (2005), pp. 507–540.

- [15] ARMIN BECKERT AND HOLGER WENDLAND, *Multivariate interpolation for fluid-structure-interaction problems using radial basis functions*, Aerospace Science and Technology, 5 (2001), pp. 125–134.
- [16] DOUGLAS BENN, DAVID JA EVANS, ET AL., *Glaciers and glaciation*, Routledge, 2014.
- [17] DAVID BERCOVICI AND YANICK RICARD, *Energetics of a two-phase model of lithospheric damage, shear localization and plate-boundary formation*, Geophysical Journal International, 152 (2003), pp. 581–596.
- [18] DAVID BERCOVICI, YANICK RICARD, AND GERALD SCHUBERT, *A two-phase model for compaction and damage: 3. applications to shear localization and plate boundary formation*, Journal of Geophysical Research: Solid Earth, 106 (2001), pp. 8925–8939.
- [19] LUCA BERGAMASCHI, MARCO CALIARI, AND MARCO VIANELLO, *The ReLPM exponential integrator for FE discretizations of advection-diffusion equations*, in International Conference on Computational Science, Springer, 2004, pp. 434–442.
- [20] S BIDWELL, M HASSELL, AND C WESTPHAL, *A weighted least squares finite element method for elliptic problems with degenerate and singular coefficients*, Mathematics of Computation, 82 (2013), pp. 673–688.
- [21] SJ BILLETT AND EF TORO, *AOn WAF-type schemes for multidimensional hyperbolic conservation laws*, Journal of Computational Physics, 130 (1997), pp. 1–24.

- [22] GEORGIJ BISPEN, KOOTTUNGAL REVI ARUN, MÁRIA LUKÁVCOVÁ-MEDVID'OVÁ,
AND SEBASTIAN NOELLE, *IMEX large time step finite volume methods for
low Froude number shallow water flows*, Communications in Computational
Physics, 16 (2014), pp. 307–347.
- [23] GEORGIJ BISPEN, MÁRIA LUKÁVCOVÁ-MEDVID'OVÁ, AND LEONID YELASH,
*Asymptotic preserving IMEX finite volume schemes for low Mach number Eu-
ler equations with gravitation*, Journal of Computational Physics, 335 (2017),
pp. 222–248.
- [24] SÉBASTIEN BLAISE, JONATHAN LAMBRECHTS, AND ERIC DELEERSNIJDER,
*A stabilization for three-dimensional discontinuous Galerkin discretizations ap-
plied to nonhydrostatic atmospheric simulations*, International Journal for Nu-
merical Methods in Fluids, (2015).
- [25] MIKE A BOTCHEV, *A short guide to exponential Krylov subspace time integra-
tion for Maxwell's equations*, Dept. of Applied Mathematics, Univ. of Twente,
(2012).
- [26] —, *A block Krylov subspace time-exact solution method for linear ordinary
differential equation systems*, Numerical linear algebra with applications, 20
(2013), pp. 557–574.
- [27] MIKE A BOTCHEV, DAVIT HARUTYUNYAN, AND JAAP JW VAN DER VEGT,
*The Gautschi time stepping scheme for edge finite element discretizations of the
Maxwell equations*, Journal of computational physics, 216 (2006), pp. 654–686.
- [28] RAY M BOWEN, *Compressible porous media models by use of the theory of
mixtures*, International Journal of Engineering Science, 20 (1982), pp. 697–735.

- [29] SUSANNE BRENNER AND RIDGWAY SCOTT, *The mathematical theory of finite element methods*, vol. 15, Springer Science & Business Media, 2007.
- [30] T. BUI-THANH, *From Godunov to a unified hybridized discontinuous Galerkin framework for partial differential equations*, Journal of Computational Physics, 295 (2015), pp. 114–146.
- [31] TAN BUI-THANH, *From Godunov to a unified hybridized discontinuous Galerkin framework for partial differential equations*, Journal of Computational Physics, 295 (2015), pp. 114–146.
- [32] T. BUI-THANH, *From Rankine-Hugoniot Condition to a Constructive Derivation of HDG Methods*, Lecture Notes in Computational Sciences and Engineering, Springer, 2015, pp. 483–491.
- [33] TAN BUI-THANH, *Construction and analysis of HDG methods for linearized shallow water equations*, SIAM Journal on Scientific Computing, 38 (2016), pp. A3696–A3719.
- [34] ———, *Construction and analysis of HDG methods for linearized shallow water equations*, SIAM Journal on Scientific Computing, 38 (2016), pp. A3696–A3719.
- [35] MARCO CALIARI AND ALEXANDER OSTERMANN, *Implementation of exponential Rosenbrock-type integrators*, Applied Numerical Mathematics, 59 (2009), pp. 568–581.
- [36] MARCO CALIARI, MARCO VIANELLO, AND LUCA BERGAMASCHI, *Interpolating discrete advection–diffusion propagators at Leja sequences*, Journal of Computational and Applied Mathematics, 172 (2004), pp. 79–99.

- [37] ELENA CELLEDONI, DAVID COHEN, AND BRYNJULF OWREN, *Symmetric exponential integrators with an application to the cubic Schrödinger equation*, Foundations of Computational Mathematics, 8 (2008), pp. 303–317.
- [38] ONDVREJ ŠRÁMEK, YANICK RICARD, AND DAVID BERCOVICI, *Simultaneous melting and compaction in deformable two-phase media*, Geophysical Journal International, 168 (2007), pp. 964–982.
- [39] ALEXANDER H-D CHENG AND EMMANUEL DETOURNAY, *A direct boundary element method for plane strain poroelasticity*, International Journal for Numerical and Analytical Methods in Geomechanics, 12 (1988), pp. 551–572.
- [40] S CHIPPADA, CN DAWSON, ML MARTINEZ-CANALES, AND MF WHEELER, *Finite element approximations to the system of shallow water equations, part ii: Discrete-time a priori error estimates*, SIAM journal on numerical analysis, 36 (1998), pp. 226–250.
- [41] RICHARD J CHORLEY AND BARBARA A KENNEDY, *Physical geography: a systems approach*, Prentice Hall, 1971.
- [42] KENNEDY CHRISTOPHER A AND CARPENTER MARK H, *Additive Runge–Kutta schemes for convection–diffusion–reaction equations*, Applied Numerical Mathematics, 44 (2003), pp. 139–181.
- [43] COLM CLANCY AND JANUSZ A PUDYKIEWICZ, *On the use of exponential time integration methods in atmospheric models*, Tellus A: Dynamic Meteorology and Oceanography, 65 (2013), p. 20898.
- [44] B. COCKBURN, B. DONG, J. GUZMAN, M. RESTELLI, AND R. SACCO, *A hybridizable discontinuous Galerkin method for steady state convection-diffusion-reaction problems*, SIAM J. Sci. Comput., 31 (2009), pp. 3827–3846.

- [45] BERNARDO COCKBURN, GUOSHENG FU, AND FRANCISCO SAYAS, *Superconvergence by m -decompositions. part i: General theory for hdg methods for diffusion*, Mathematics of Computation, 86 (2017), pp. 1609–1641.
- [46] B. COCKBURN AND J. GOPALAKRISHNAN, *The derivation of hybridizable discontinuous Galerkin methods for Stokes flow*, SIAM J. Numer. Anal., 47 (2009), pp. 1092–1125.
- [47] BERNARDO COCKBURN, JAY GOPALAKRISHNAN, AND RAYTCHO LAZAROV, *Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems*, SIAM J. Numer. Anal., 47 (2009), pp. 1319–1365.
- [48] BERNARDO COCKBURN, JAY GOPALAKRISHNAN, AND FRANCISCO-JAVIER SAYAS, *A projection-based error analysis of HDG methods*, Mathematics Of Computation, 79 (2010), pp. 1351–1367.
- [49] BERNARDO COCKBURN, GEORGE E KARNIADAKIS, AND CHI-WANG SHU, *The development of discontinuous Galerkin methods*, in Discontinuous Galerkin Methods, Springer, 2000, pp. 3–50.
- [50] B. COCKBURN, F. LI, N. C. NGUYEN, AND J. PERAIRE, *Hybridization and postprocessing techniques for mixed eigenfunctions*, SIAM J. Numer. Anal., 48 (2010), pp. 857–881.
- [51] BERNARDO COCKBURN, WEIFENG QIU, AND KE SHI, *Conditions for superconvergence of HDG methods for second-order elliptic problems*, Mathematics of Computation, 81 (2012), pp. 1327–1353.

- [52] BERNARDO COCKBURN AND CHI-WANG SHU, *The Runge–Kutta discontinuous Galerkin method for conservation laws v: multidimensional systems*, Journal of Computational Physics, 141 (1998), pp. 199–224.
- [53] J CÔTÉ, *A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere*, Quarterly Journal of the Royal Meteorological Society, 114 (1988), pp. 1347–1352.
- [54] STEVEN M COX AND PAUL C MATTHEWS, *Exponential time differencing for stiff systems*, Journal of Computational Physics, 176 (2002), pp. 430–455.
- [55] J. CUI AND W. ZHANG, *An analysis of HDG methods for the Helmholtz equation*, IMA J. Numer. Anal., 34 (2014), pp. 279–295.
- [56] HENRY PHILIBERT GASPARD DARCY, *Les Fontaines publiques de la ville de Dijon. Exposition et application des principes à suivre et des formules à employer dans les questions de distribution d’eau, etc*, Victor Dalmont, 1856.
- [57] TIMOTHY A DAVIS, *Algorithm 832: UMFPACK v4. 3—an unsymmetric-pattern multifrontal method*, ACM Transactions on Mathematical Software (TOMS), 30 (2004), pp. 196–199.
- [58] A DE BOER, MS VAN DER SCHOOT, AND HESTER BIJL, *Mesh deformation based on radial basis function interpolation*, Computers & structures, 85 (2007), pp. 784–795.
- [59] PIERRE DEGOND, S JIN, AND J YUMING, *Mach-number uniform asymptotic-preserving gauge schemes for compressible flows*, Bulletin-Institute of Mathematics Academia Sinica, 2 (2007), p. 851.

- [60] LESZEK DEMKOWICZ, *Computing with hp-adaptive finite elements: Volume 1 one and two dimensional elliptic and Maxwell problems*, Chapman and Hall/CRC, 2006.
- [61] V DOLEJVSÍ, M FEISTAUER, AND J HOZMAN, *Analysis of semi-implicit DGFEM for nonlinear convection–diffusion problems on nonconforming meshes*, Computer methods in applied mechanics and engineering, 196 (2007), pp. 2813–2827.
- [62] JEAN DONEA, S GIULIANI, AND JEAN-PIERRE HALLEUX, *An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions*, Computer methods in applied mechanics and engineering, 33 (1982), pp. 689–723.
- [63] JIM DOUGLAS AND TODD DUPONT, *Interior penalty procedures for elliptic and parabolic Galerkin methods*, in Computing methods in applied sciences, Springer, 1976, pp. 207–216.
- [64] F DUCROS, F LAPORTE, TH SOULÈRES, V GUINOT, PH MOINAT, AND B CARUELLE, *High-order fluxes for conservative skew-symmetric-like schemes in structured meshes: application to compressible flows*, Journal of Computational Physics, 161 (2000), pp. 114–139.
- [65] M. DUMBSER AND V. CASULLI, *A staggered semi-implicit spectral discontinuous Galerkin scheme for the shallow water equations*, Applied Mathematics and Computation, 219 (2013), pp. 8057–8077.
- [66] H. EGGER AND J. SCHÖBERL, *A hybrid mixed discontinuous Galerkin finite element method for convection-diffusion problems*, IMA Journal of Numerical Analysis, 30 (2010), pp. 1206–1234.

- [67] GORDON ERLEBACHER, M YOUSUFF HUSSAINI, AND C-W SHU, *Interaction of a shock with a longitudinal vortex*, Journal of Fluid Mechanics, 337 (1997), pp. 129–153.
- [68] O ESTRUCH, O LEHMKUHL, R BORRELL, CD PÉREZ SEGARRA, AND A OLIVA, *A parallel radial basis function interpolation method for unstructured dynamic meshes*, Computers & Fluids, 80 (2013), pp. 44–54.
- [69] M. J. BERTIN ET AL, *Pisot and Salem Numbers*, user Verlag, Berlin, 1992.
- [70] M. FEISTAUER, V. DOLEJSI, AND V. KUCERA, *On the discontinuous Galerkin method for the simulation of compressible flow with wide range of Mach numbers*, Computing and Visualization in Science, 10 (2007), pp. 17–27.
- [71] ESTEBAN FERRER, *An interior penalty stabilised incompressible discontinuous Galerkin–fourier solver for implicit large eddy simulations*, Journal of Computational Physics, 348 (2017), pp. 754–775.
- [72] MASSIMILIANO FERRONATO, NICOLA CASTELLETTO, AND GIUSEPPE GAMBOLATI, *A fully coupled 3-d mixed finite element model of Biot consolidation*, Journal of Computational Physics, 229 (2010), pp. 4813–4830.
- [73] LOULA FEZOU, STÉPHANE LANTERI, STÉPHANIE LOHRENGEL, AND SERGE PIPERNO, *Convergence and stability of a discontinuous Galerkin time-domain method for the 3d heterogeneous Maxwell equations on unstructured meshes*, ESAIM: Mathematical Modelling and Numerical Analysis, 39 (2005), pp. 1149–1176.
- [74] KRZYSZTOF J FIDKOWSKI, *A hybridized discontinuous Galerkin method on mapped deforming domains*, Computers & Fluids, 139 (2016), pp. 80–91.

- [75] MICHEL FORTIN AND ROLAND GLOWINSKI, *Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems*, vol. 15, Elsevier, 2000.
- [76] AC FOWLER, *On the transport of moisture in polythermal glaciers*, Geophysical & Astrophysical Fluid Dynamics, 28 (1984), pp. 99–140.
- [77] JOSEPH GALEWSKY, RICHARD K SCOTT, AND LORENZO M POLVANI, *An initial-value problem for testing numerical models of the global shallow-water equations*, Tellus A, 56 (2004), pp. 429–440.
- [78] GREGOR J GASSNER, *A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods*, SIAM Journal on Scientific Computing, 35 (2013), pp. A1233–A1253.
- [79] —, *A kinetic energy preserving nodal discontinuous Galerkin spectral element method*, International Journal for Numerical Methods in Fluids, 76 (2014), pp. 28–50.
- [80] GREGOR J GASSNER, ANDREW R WINTERS, FLORIAN J HINDENLANG, AND DAVID A KOPRIVA, *The BR1 scheme is stable for the compressible Navier–Stokes equations*, Journal of Scientific Computing, (2017), pp. 1–47.
- [81] GREGOR J GASSNER, ANDREW R WINTERS, AND DAVID A KOPRIVA, *Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations*, Journal of Computational Physics, 327 (2016), pp. 39–66.
- [82] STÉPHANE GAUDREAU, GREG RAINWATER, AND MAYYA TOKMAN, *Kiops: A fast adaptive Krylov subspace solver for exponential integrators*, Journal of Computational Physics, 372 (2018), pp. 236–255.

- [83] A YU GELFGAT, PZ BAR-YOSEPH, AND AL YARIN, *Stability of multiple steady states of convection in laterally heated cavities*, Journal of Fluid Mechanics, 388 (1999), pp. 315–334.
- [84] FRANCIS X GIRALDO, JAN S HESTHAVEN, AND TIM WARBURTON, *Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations*, Journal of Computational Physics, 181 (2002), pp. 499–525.
- [85] FRANCIS X GIRALDO, JAMES F KELLY, AND EM CONSTANTINESCU, *Implicit-explicit formulations of a three-dimensional nonhydrostatic unified model of the atmosphere (NUMA)*, SIAM Journal on Scientific Computing, 35 (2013), pp. B1162–B1194.
- [86] FRANCIS X GIRALDO AND MARCO RESTELLI, *A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases*, Journal of Computational Physics, 227 (2008), pp. 3849–3877.
- [87] F. X. GIRALDO AND M. RESTELLI, *High-order semi-implicit time-integrators for a triangular discontinuous Galerkin oceanic shallow water model*, International Journal For Numerical Methods In Fluids, 63 (2010), pp. 1077–1102.
- [88] FRANCIS X GIRALDO, MARCO RESTELLI, AND MATTHIAS LÄUTER, *Semi-implicit formulations of the Navier–Stokes equations: Application to nonhydrostatic atmospheric modeling*, SIAM Journal on Scientific Computing, 32 (2010), pp. 3394–3425.
- [89] F. X. GIRALDO AND T. WARBURTON, *A high-order triangular discontinuous Galerkin oceanic shallow water model*, International Journal For Numerical Methods In Fluids, 56 (2008), pp. 899–925.

- [90] WILLIAM J GORDON AND CHARLES A HALL, *Construction of curvilinear co-ordinate systems and applications to mesh generation*, International Journal for Numerical Methods in Engineering, 7 (1973), pp. 461–477.
- [91] SYLVIE GRAVEL AND ANDREW STANIFORTH, *A mass-conserving semi-lagrangian scheme for the shallow-water equations*, Monthly Weather Review, 122 (1994), pp. 243–248.
- [92] R. GRIESMAIER AND P. MONK, *Error analysis for a hybridizable discontinuous Galerkin method for the Helmholtz equation*, J. Sci. Comput., 49 (2011), pp. 291–310.
- [93] BERNHARD HAURWITZ, *The motion of atmospheric disturbances on the spherical earth*, Journal of Marine Research, 3 (1940), pp. 254–267.
- [94] YINNAN HE, *The euler implicit/explicit scheme for the 2d time-dependent navier-stokes equations with smooth or non-smooth initial data*, Mathematics of Computation, 77 (2008), pp. 2097–2124.
- [95] TOM HERRING, *Geodesy: treatise on geophysics*, Elsevier, 2010.
- [96] JAN S HESTHAVEN AND TIM WARBURTON, *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, Springer Science & Business Media, 2007.
- [97] IJ HEWITT AND AC FOWLER, *Partial melting in an upwelling mantle column*, in Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, The Royal Society, 2008, pp. 2467–2491.
- [98] JOHN G HEYWOOD AND ROLF RANNACHER, *Finite element approximation of the nonstationary Navier–Stokes problem. i. regularity of solutions and second-*

- order error estimates for spatial discretization*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 275–311.
- [99] ———, *Finite-element approximation of the nonstationary Navier–Stokes problem. part iv: Error analysis for second-order time discretization*, SIAM Journal on Numerical Analysis, 27 (1990), pp. 353–384.
- [100] CW HIRT, ANTHONY A AMSDEN, AND JL COOK, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, Journal of computational physics, 14 (1974), pp. 227–253.
- [101] MARLIS HOCHBRUCK AND ALEXANDER OSTERMANN, *Exponential integrators*, Acta Numerica, 19 (2010), pp. 209–286.
- [102] MARK HOEMMEN, *Communication-avoiding Krylov subspace methods*, PhD thesis, UC Berkeley, 2010.
- [103] CLAES JOHNSON AND JUHANI PITKÄRANTA, *An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation*, Mathematics of computation, 46 (1986), pp. 1–26.
- [104] SHINHOO KANG, TAN BUI-THANH, AND TODD ARBOGAST, *A hybridized discontinuous Galerkin method for a linear degenerate elliptic equation arising from two-phase mixtures*, Computer Methods in Applied Mechanics and Engineering, 350 (2019), pp. 315–336.
- [105] SHINHOO KANG, FRANCIS X GIRALDO, AND TAN BUI-THANH, *IMEX HDG-DG: a coupled implicit hybridized discontinuous Galerkin (HDG) and explicit discontinuous Galerkin (DG) approach for shallow water systems*, arXiv preprint arXiv:1711.02751, (2017).

- [106] SAM KANNER, LUMING WANG, AND PER-OLOF PERSSON, *Implicit large-eddy simulation of 2d counter-rotating vertical-axis wind turbines*, in 34th Wind Energy Symposium, 2016, p. 1731.
- [107] GEORG KASER, *Glacier-climate interaction at low latitudes*, Journal of Glaciology, 47 (2001), pp. 195–204.
- [108] CHRISTOPHER A KENNEDY AND ANDREA GRUBER, *Reduced aliasing formulations of the convective terms within the Navier–Stokes equations for a compressible fluid*, Journal of Computational Physics, 227 (2008), pp. 1676–1700.
- [109] ANDREW C KIRBY, MICHAEL BRAZELL, ZHI YANG, RAJIB ROY, BEHZAD REZA AHRABI, DIMITRI MAVRIPLIS, JAY SITARAMAN, AND MICHAEL K STOELLINGER, *Wind farm simulations using an overset hp-adaptive approach with blade-resolved turbine models*, in 23rd AIAA Computational Fluid Dynamics Conference, 2017, p. 3958.
- [110] R. M. KIRBY, S. J. SHERWIN, AND B. COCKBURN, *To CG or to HDG: A comparative study*, J. Sci. Comput., 51 (2012), pp. 183–212.
- [111] DONALD K. KNUTH, *The T_EXbook*, Addison-Wesley, 1984.
- [112] GIJS L KOOIJ, MIKE A BOTCHEV, AND BERNARD J GEURTS, *A block krylov subspace implementation of the time-parallel paraexp method and its extension for nonlinear partial differential equations*, Journal of computational and applied mathematics, 316 (2017), pp. 229–246.
- [113] GIJS L KOOIJ, MIKE A BOTCHEV, AND BERNARD J GEURTS, *An exponential time integrator for the incompressible Navier–Stokes equation*, SIAM journal on scientific computing, 40 (2018), pp. B684–B705.

- [114] DAVID A KOPRIVA, *A conservative staggered-grid Chebyshev multidomain method for compressible flows. ii. a semi-structured method*, Journal of computational physics, 128 (1996), pp. 475–488.
- [115] —, *Metric identities and the discontinuous spectral element method on curvilinear meshes*, Journal of Scientific Computing, 26 (2006), p. 301.
- [116] —, *Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers*, Springer Science & Business Media, 2009.
- [117] DAVID A KOPRIVA AND GREGOR J GASSNER, *An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems*, SIAM Journal on Scientific Computing, 36 (2014), pp. A2076–A2099.
- [118] DAVID A KOPRIVA, ANDREW R WINTERS, MARVIN BOHM, AND GREGOR J GASSNER, *A provably stable discontinuous Galerkin spectral element approximation for moving hexahedral meshes*, Computers & Fluids, 139 (2016), pp. 148–160.
- [119] ANTTI KOSKELA, *Approximating the matrix exponential of an advection-diffusion operator using the incomplete orthogonalization method*, in Numerical Mathematics and Advanced Applications-ENUMATH 2013, Springer, 2015, pp. 345–353.
- [120] LIG KOVASZNAY, *Laminar flow behind a two-dimensional grid*, in Mathematical Proceedings of the Cambridge Philosophical Society, Cambridge University Press, 1948, pp. 58–62.
- [121] LESLIE LAMPORT, *TEX: A document preparation system*, Addison-Wesley, 2nd ed., 1994.

- [122] MATTHIAS LÄUTER, FRANCIS X GIRALDO, DÖRTHE HANDORF, AND KLAUS DETHLOFF, *A discontinuous Galerkin method for the shallow water equations in spherical triangular coordinates*, Journal of Computational Physics, 227 (2008), pp. 10226–10242.
- [123] JEONGHUN J LEE, STEPHEN SHANNON, TAN BUI-THANH, AND JOHN N SHADID, *Construction and analysis of an hdg method for incompressible magnetohydrodynamics*, arXiv preprint arXiv:1702.05124, (2017).
- [124] RAGNAR S LEHMANN, MÁRIA LUKÁVCOVÁ-MEDVID’OVÁ, BORIS JP KAUS, AND ANTON A POPOV, *Comparison of continuous and discontinuous Galerkin approaches for variable-viscosity Stokes flow*, ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 96 (2016), pp. 733–746.
- [125] P. LESAIN AND P. A. RAVIART, *On a finite element method for solving the neutron transport equation*, in Mathematical Aspects of Finite Element Methods in Partial Differential Equations, C. de Boor, ed., Academic Press, 1974, pp. 89–145.
- [126] RANDALL J LEVEQUE, *Finite volume methods for hyperbolic problems*, vol. 31, Cambridge university press, 2002.
- [127] HENGGUANG LI, *A-priori analysis and the finite element method for a class of degenerate elliptic equations*, Mathematics of Computation, 78 (2009), pp. 713–737.
- [128] H. LI AND R. X. LIU, *The discontinuous Galerkin finite element method for the 2D shallow water equations*, Mathematics and Computers in Simulation, 56 (2001), pp. 171–184.

- [129] L. LI, S. LANTERI, AND R. PERRRUSSEL, *A hybridizable discontinuous Galerkin method for solving 3D time harmonic Maxwell's equations*, in Numerical Mathematics and Advanced Applications 2011, Springer, 2013, pp. 119–128.
- [130] SHU-JIE LI AND LILI JU, *Exponential time-marching method for the unsteady Navier-Stokes equations*, in AIAA Scitech 2019 Forum, 2019, p. 0907.
- [131] SHU-JIE LI, LI-SHI LUO, ZHI JIAN WANG, AND LILI JU, *An exponential time-integrator scheme for steady and unsteady inviscid flows*, Journal of Computational Physics, 365 (2018), pp. 206–225.
- [132] RUIJIE LIU, *Discontinuous Galerkin finite element solution for poromechanics*, PhD thesis, The University of Texas at Austin, 2004.
- [133] VU THAI LUAN, JANUSZ A PUDYKIEWICZ, AND DANIEL R REYNOLDS, *Further development of efficient and accurate time integration schemes for meteorological models*, Journal of Computational Physics, 376 (2019), pp. 817–837.
- [134] F MITTELBACH M GOOSENS AND A SAMARIN, *The L^AT_EX Companion*, Addison-Wesley, 1994.
- [135] PIERRE-HENRI MAIRE, RÉMI ABGRALL, JÉRÔME BREIL, AND JEAN OVA-DIA, *A cell-centered Lagrangian scheme for two-dimensional compressible flow problems*, SIAM Journal on Scientific Computing, 29 (2007), pp. 1781–1824.
- [136] JEAN MANDEL, *Consolidation des sols (étude mathématique)*, Geotechnique, 3 (1953), pp. 287–299.
- [137] JUAN MANZANERO, GONZALO RUBIO, ESTEBAN FERRER, EUSEBIO VALERO, AND DAVID A KOPRIVA, *Insights on aliasing driven instabilities for advec-*

- tion equations with application to gauss–lobatto discontinuous galerkin methods*, Journal of Scientific Computing, 75 (2018), pp. 1262–1281.
- [138] S MARRAS, MICHAL A KOPERA, AND FRANCIS X GIRALDO, *Simulation of shallow-water jets with a unified element-based continuous/discontinuous Galerkin model with grid flexibility on the sphere*, Quarterly Journal of the Royal Meteorological Society, 141 (2015), pp. 1727–1739.
- [139] KATATE MASATSUKA, *I do Like CFD, vol. 1*, vol. 1, Lulu. com, 2013.
- [140] CATHY MAVRIPLIS, *Nonconforming discretizations and a posteriori error estimators for adaptive spectral element techniques*, PhD thesis, Massachusetts Institute of Technology, 1989.
- [141] DAN MCKENZIE, *The generation and compaction of partially molten rock*, Journal of Petrology, 25 (1984), pp. 713–765.
- [142] DAN P MCKENZIE, JEAN M ROBERTS, AND NIGEL O WEISS, *Convection in the Earth’s mantle: towards a numerical simulation*, Journal of Fluid Mechanics, 62 (1974), pp. 465–538.
- [143] DOMINIK L MICHELS, VU THAI LUAN, AND MAYYA TOKMAN, *A stiffly accurate integrator for elastodynamic problems*, ACM Transactions on Graphics (TOG), 36 (2017), p. 116.
- [144] DOMINIK L MICHELS, GERRIT A SOBOTTKA, AND ANDREAS G WEBER, *Exponential integrators for stiff elastodynamic problems*, ACM Transactions on Graphics (TOG), 33 (2014), p. 7.

- [145] IGOR MORET, *On RD-rational Krylov approximations to the core-functions of exponential integrators*, Numerical Linear Algebra with Applications, 14 (2007), pp. 445–457.
- [146] YOHEI MORINISHI, *Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-Mach number flows*, Journal of Computational Physics, 229 (2010), pp. 276–300.
- [147] D. MORO, N. C. NGUYEN, AND J. PERAIRE, *Navier-Stokes solution using hybridizable discontinuous Galerkin methods*, American Institute of Aeronautics and Astronautics, 2011-3407 (2011).
- [148] LIN MU, JUNPING WANG, AND XIU YE, *A new weak Galerkin finite element method for the Helmholtz equation*, IMA Journal of Numerical Analysis, 35 (2014), pp. 1228–1255.
- [149] —, *A new weak Galerkin finite element method for the Helmholtz equation*, IMA Journal of Numerical Analysis, (2014).
- [150] SRIRAMKRISHNAN MURALIKRISHNAN, MINH-BINH TRAN, AND TAN BUI-THANH, *ihdg: An iterative HDG framework for partial differential equations*, arXiv preprint arXiv:1605.03228, (2016).
- [151] —, *iHDG: An iterative HDG framework for partial differential equations*, SIAM Journal on Scientific Computing, 39 (2017), pp. S782–S808.
- [152] RAMACHANDRAN D NAIR, STEPHEN J THOMAS, AND RICHARD D LOFT, *A discontinuous Galerkin global shallow water model*, Monthly Weather Review, 133 (2005), pp. 876–888.

- [153] NC NGUYEN, J PERAIRE, AND B COCKBURN, *A hybridizable discontinuous Galerkin method for Stokes flow*, Computer Methods in Applied Mechanics and Engineering, 199 (2010), pp. 582–597.
- [154] N. C. NGUYEN AND J. PERAIRE, *Hybridizable discontinuous Galerkin methods for partial differential equations in continuum mechanics*, Journal Computational Physics, 231 (2012), pp. 5955–5988.
- [155] N. C. NGUYEN, J. PERAIRE, AND B. COCKBURN, *An implicit high-order hybridizable discontinuous Galerkin method for linear convection-diffusion equations*, Journal Computational Physics, 228 (2009), pp. 3232–3254.
- [156] —, *An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations*, Journal Computational Physics, 228 (2009), pp. 8841–8855.
- [157] —, *A hybridizable discontinuous Galerkin method for Stokes flow*, Comput Method Appl. Mech. Eng., 199 (2010), pp. 582–597.
- [158] —, *High-order implicit hybridizable discontinuous Galerkin method for acoustics and elastodynamics*, Journal Computational Physics, 230 (2011), pp. 3695–3718.
- [159] —, *Hybridizable discontinuous Galerkin method for the time harmonic Maxwell’s equations*, Journal Computational Physics, 230 (2011), pp. 7151–7175.
- [160] NGOC CUONG NGUYEN, JAUME PERAIRE, AND BERNARDO COCKBURN, *An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations*, Journal of Computational Physics, 230 (2011), pp. 1147–1170.

- [161] N. C. NGUYEN, J. PERAIRE, AND B. COCKBURN, *An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier-Stokes equations*, Journal Computational Physics, 230 (2011), pp. 1147–1170.
- [162] JITSE NIESEN AND WILL M WRIGHT, *Algorithm 919: A Krylov subspace algorithm for evaluating the ϕ -functions appearing in exponential integrators*, ACM Transactions on Mathematical Software (TOMS), 38 (2012), p. 22.
- [163] LUDOVIC NOELS AND RAÚL RADOVITZKY, *An explicit discontinuous Galerkin method for non-linear solid dynamics: Formulation, parallel implementation and scalability properties*, International Journal for Numerical Methods in Engineering, 74 (2008), pp. 1393–1420.
- [164] ISSEI OIKAWA, *A hybridized discontinuous Galerkin method with reduced stabilization*, Journal of Scientific Computing, 65 (2015), pp. 327–340.
- [165] LORENZO PARESCHI AND GIOVANNI RUSSO, *Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation*, Journal of Scientific computing, 25 (2005), pp. 129–155.
- [166] PHILLIP JOSEPH PHILLIPS AND MARY F WHEELER, *A coupling of mixed and continuous Galerkin finite element methods for poroelasticity i: the continuous in time case*, Computational Geosciences, 11 (2007), p. 131.
- [167] —, *Overcoming the problem of locking in linear elasticity and poroelasticity: an heuristic approach*, Computational Geosciences, 13 (2009), pp. 5–12.
- [168] D. A. DI PIETRO AND A. ERN, *Mathematical aspects of Discontinuous Galerkin methods*, Springer, 2012.

- [169] SERGIO PIROZZOLI, *Numerical methods for high-speed flows*, Annual review of fluid mechanics, 43 (2011), pp. 163–194.
- [170] STEFANIA RAGNI, *Rational Krylov methods in exponential integrators for European option pricing*, Numerical Linear Algebra with Applications, 21 (2014), pp. 494–512.
- [171] W. H. REED AND T. R. HILL, *Triangular mesh methods for the neutron transport equation*, Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
- [172] TCS RENDALL AND CB ALLEN, *Unified fluid–structure interpolation and mesh motion using radial basis functions*, International Journal for Numerical Methods in Engineering, 74 (2008), pp. 1519–1559.
- [173] —, *Parallel efficient mesh motion using radial basis functions with application to multi-bladed rotors*, International journal for numerical methods in engineering, 81 (2010), pp. 89–105.
- [174] M. RESTELLI AND F. X. GIRALDO, *A conservative discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in non-hydrostatic mesoscale modeling*, SIAM Journal on Scientific Computing, 31 (2009), pp. 2231–2257.
- [175] SANDER RHEBERGEN AND GARTH N WELLS, *A hybridizable discontinuous Galerkin method for the navier–stokes equations with pointwise divergence-free velocity field*, arXiv preprint arXiv:1704.07569, (2017).
- [176] SANDER RHEBERGEN, GARTH N WELLS, RICHARD F KATZ, AND ANDREW J WATHEN, *Analysis of block preconditioners for models of coupled magma/mantle dynamics*, SIAM Journal on Scientific Computing, 36 (2014), pp. A1960–A1977.

- [177] MARIO RICCHIUTO AND ANDREAS BOLLERMANN, *Stabilized residual distribution for shallow water simulations*, Journal of Computational Physics, 228 (2009), pp. 1071–1115.
- [178] A. ROBERT, J. HENDERSON, AND C. TURNBULL, *An implicit time integration scheme for baroclinic models of the atmosphere*, Monthly Weather Review, 100 (1972), pp. 329–335.
- [179] P. L. ROE, *Approximate Riemann solvers, parametric vectors, and difference schemes*, Journal of Computational Physics, 43 (1981), pp. 357–372.
- [180] R. RÖÖM, *Acoustic filtering in nonhydrostatic pressure coordinate dynamics: a variational approach*, Journal of the Atmospheric Sciences, 55 (1998), pp. 654–668.
- [181] VLADIMIR VASIL'EVICH RUSANOV, *The calculation of the interaction of non-stationary shock waves and obstacles*, USSR Computational Mathematics and Mathematical Physics, 1 (1962), pp. 304–320.
- [182] YOUSEF SAAD, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM Journal on Numerical Analysis, 29 (1992), pp. 209–228.
- [183] RICHARD SAUREL AND RÉMI ABGRALL, *A simple method for compressible multifluid flows*, SIAM Journal on Scientific Computing, 21 (1999), pp. 1115–1145.
- [184] FRANK SCHEURICH AND RICHARD E. BROWN, *Modelling the aerodynamics of vertical-axis wind turbines in unsteady wind conditions*, Wind Energy, 16 (2013), pp. 91–107.

- [185] ALAN R SCHIEMENZ, *Advances in the discontinuous Galerkin Method: hybrid schemes and applications to the reactive infiltration instability in an upwelling compacting mantle*, PhD thesis, Brown University, 2009.
- [186] ALAN R SCHIEMENZ, MARC A HESSE, AND JAN S HESTHAVEN, *Modeling magma dynamics with a mixed Fourier collocation-discontinuous Galerkin method*, Communications in Computational Physics, 10 (2011), pp. 433–452.
- [187] JAN C SCHULZE, PETER J SCHMID, AND JÖRN L SESTERHENN, *Exponential time integration using Krylov subspaces*, International journal for numerical methods in fluids, 60 (2009), pp. 591–609.
- [188] DAVID R SCOTT AND DAVID J STEVENSON, *Magma solitons*, Geophysical Research Letters, 11 (1984), pp. 1161–1164.
- [189] —, *Magma ascent by porous flow*, Journal of Geophysical Research: Solid Earth, 91 (1986), pp. 9283–9296.
- [190] ROGER B SIDJE, *Expokit: A software package for computing matrix exponentials*, ACM Transactions on Mathematical Software (TOMS), 24 (1998), pp. 130–156.
- [191] BRARD SKAFLESTAD AND WILL M WRIGHT, *The scaling and modified squaring method for matrix functions related to the exponential*, Applied Numerical Mathematics, 59 (2009), pp. 783–799.
- [192] WILLIAM C SKAMAROCK AND JOSEPH B KLEMP, *Efficiency and accuracy of the klemp-wilhelmson time-splitting technique*, Monthly Weather Review, 122 (1994), pp. 2623–2630.

- [193] NORMAN H SLEEP, *Tapping of melt by veins and dikes*, Journal of Geophysical Research: Solid Earth, 93 (1988), pp. 10255–10272.
- [194] MICHAEL SPIVAK, *The joy of T_EX*, American Mathematical Society, Providence, R.I., 2nd ed., 1990.
- [195] RENE STEIJL AND GEORGE BARAKOS, *Sliding mesh algorithm for CFD analysis of helicopter rotor–fuselage aerodynamics*, International journal for numerical methods in fluids, 58 (2008), pp. 527–549.
- [196] JERRY M STRAKA, ROBERT B WILHELMSON, LOUIS J WICKER, JOHN R ANDERSON, AND KELVIN K DROEGEMEIER, *Numerical solutions of a non-linear density current: A benchmark solution and comparisons*, International Journal for Numerical Methods in Fluids, 17 (1993), pp. 1–22.
- [197] BO STRAND, *Summation by parts for finite difference approximations for d/dx* , Journal of Computational Physics, 110 (1994), pp. 47–67.
- [198] HILLEL TAL-EZER, *On restart and error estimation for Krylov approximation of $w = f(a)v$* , SIAM Journal on Scientific Computing, 29 (2007), pp. 2426–2441.
- [199] SIRUI TAN AND CHI-WANG SHU, *A high order moving boundary treatment for compressible inviscid flows*, Journal of Computational Physics, 230 (2011), pp. 6023–6036.
- [200] SESHU TIRUPATHI, JAN S HESTHAVEN, AND YAN LIANG, *Modeling 3d magma dynamics using a discontinuous Galerkin method*, Communications in Computational Physics, 18 (2015), pp. 230–246.

- [201] SESHU TIRUPATHI, JAN S HESTHAVEN, YAN LIANG, AND MARC PARMEN-
TIER, *Multilevel and local time-stepping discontinuous Galerkin methods for
magma dynamics*, Computational Geosciences, 19 (2015), pp. 965–978.
- [202] MAYYA TOKMAN, *Efficient integration of large stiff systems of ODEs with expo-
nential propagation iterative (EPI) methods*, Journal of Computational Physics,
213 (2006), pp. 748–776.
- [203] M TOKMAN AND PM BELLAN, *Three-dimensional model of the structure and
evolution of coronal mass ejections*, The Astrophysical Journal, 567 (2002),
p. 1202.
- [204] ELEUTERIO F TORO, *Riemann problems and the WAF method for solving
the two-dimensional shallow water equations*, Philosophical Transactions of the
Royal Society of London A: Mathematical, Physical and Engineering Sciences,
338 (1992), pp. 43–68.
- [205] —, *Riemann solvers and numerical methods for fluid dynamics: a practical
introduction*, Springer Science & Business Media, 2013.
- [206] PAUL ULLRICH AND CHRISTIANE JABLONOWSKI, *Operator-split Runge–Kutta–
Rosenbrock methods for nonhydrostatic atmospheric models*, Monthly Weather
Review, 140 (2012), pp. 1257–1284.
- [207] PAUL A ULLRICH, CHRISTIANE JABLONOWSKI, AND BRAM VAN LEER, *High-
order finite-volume methods for the shallow-water equations on the sphere*, Jour-
nal of Computational Physics, 229 (2010), pp. 6104–6134.
- [208] ALF J. VAN DER POORTEN, *Some problems of recurrent interest*, Tech. Re-
port 81-0037, School of Mathematics and Physics, Macquarie University, North
Ryde, Australia 2113, August 1981.

- [209] CORNELIS J VAN DER VEEN, *Fundamentals of glacier dynamics*, CRC Press, 2013.
- [210] HUY D VO AND ROGER B SIDJE, *Approximating the large sparse matrix exponential using incomplete orthogonalization and Krylov subspaces of variable dimension*, Numerical linear algebra with applications, 24 (2017), p. e2090.
- [211] J WALTZ, THOMAS R CANFIELD, NATHANIEL R MORGAN, LD RISINGER, AND JOHN G WOHLBIER, *Manufactured solutions for the three-dimensional Euler equations with relevance to inertial confinement fusion*, Journal of Computational Physics, 267 (2014), pp. 196–209.
- [212] HAIJIN WANG, CHI-WANG SHU, AND QIANG ZHANG, *Stability and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for advection-diffusion problems*, SIAM Journal on Numerical Analysis, 53 (2015), pp. 206–227.
- [213] —, *Stability analysis and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for nonlinear convection-diffusion problems*, Applied Mathematics and Computation, 272 (2016), pp. 237–258.
- [214] HAIJIN WANG, QIANG ZHANG, AND CHI-WANG SHU, *Stability analysis and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for the time-dependent fourth order pdes*, ESAIM: Mathematical Modelling and Numerical Analysis, 51 (2017), pp. 1931–1955.
- [215] HERBERT F WANG, *Theory of linear poroelasticity with applications to geomechanics and hydrogeology*, Princeton University Press, 2017.

- [216] JUNPING WANG AND XIU YE, *A weak Galerkin finite element method for second-order elliptic problems*, Journal of Computational and Applied Mathematics, 241 (2013), pp. 103–115.
- [217] JUNPING WANG AND XIU YE, *A weak Galerkin mixed finite element method for second order elliptic problems*, Math. Comp., 83 (2014), pp. 2101–2126.
- [218] LUMING WANG AND PER-OLOF PERSSON, *A high-order discontinuous Galerkin method with unstructured space–time meshes for two-dimensional compressible flows on domains with large deformations*, Computers & Fluids, 118 (2015), pp. 53–68.
- [219] RF WARMING, RICHARD M BEAM, AND BJ HYETT, *Diagonalization and simultaneous symmetrization of the gas-dynamic matrices*, Mathematics of Computation, 29 (1975), pp. 1037–1045.
- [220] HILARY WELLER, SARAH-JANE LOCK, AND NIGEL WOOD, *Runge–Kutta IMEX schemes for the horizontally explicit/vertically implicit (HEVI) solution of wave equations*, Journal of Computational Physics, 252 (2013), pp. 365–381.
- [221] HOLGER WENDLAND, *Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree*, Advances in computational Mathematics, 4 (1995), pp. 389–396.
- [222] MARY FANETT WHEELER, *An elliptic collocation-finite element method with interior penalties*, SIAM Journal on Numerical Analysis, 15 (1978), pp. 152–161.
- [223] LUCAS C WILCOX, GEORG STADLER, CARSTEN BURSTEDDE, AND OMAR GHATTAS, *A high-order discontinuous Galerkin method for wave propagation through coupled elastic–acoustic media*, Journal of Computational Physics, 229 (2010), pp. 9373–9396.

- [224] TIM WILDEY, SRIRAMKRISHNAN MURALIKRISHNAN, AND TAN BUI-THANH, *Unified geometric multigrid algorithm for hybridized high-order finite element methods*, arXiv preprint arXiv:1811.09909, (2018).
- [225] DAVID L WILLIAMSON, JOHN B DRAKE, JAMES J HACK, RÜDIGER JAKOB, AND PAUL N SWARZTRAUBER, *A standard test set for numerical approximations to the shallow water equations in spherical geometry*, Journal of Computational Physics, 102 (1992), pp. 211–224.
- [226] NIKLAS WINTERMEYER, ANDREW R WINTERS, GREGOR J GASSNER, AND DAVID A KOPRIVA, *An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry*, Journal of Computational Physics, 340 (2017), pp. 200–242.
- [227] ANDREW R WINTERS AND GREGOR J GASSNER, *Affordable, entropy conserving and entropy stable flux functions for the ideal mhd equations*, Journal of Computational Physics, 304 (2016), pp. 72–108.
- [228] MICHAEL WOOPEN, ARAVIND BALAN, GEORG MAY, AND JOCHEN SCHÜTZ, *A comparison of hybridized and standard DG methods for target-based hp-adaptive simulation of compressible flow*, Computers & Fluids, 98 (2014), pp. 3–16.
- [229] YAN XU AND CHI-WANG SHU, *Local discontinuous Galerkin methods for three classes of nonlinear wave equations*, Journal of Computational Mathematics, (2004), pp. 250–274.
- [230] MATTHEW J ZAHR AND PER-OLOF PERSSON, *An optimization based discontinuous Galerkin approach for high-order accurate shock tracking*, in 2018 AIAA Aerospace Sciences Meeting, 2018, p. 0063.

- [231] JONAS ZEIFANG, KLAUS KAISER, ANDREA BECK, JOCHEN SCHÜTZ, AND CLAUS-DIETER MUNZ, *Efficient high-order discontinuous Galerkin computations of low Mach number flows*, Communications in Applied Mathematics and Computational Science, 13 (2018), pp. 243–270.
- [232] QILONG ZHAI, RAN ZHANG, AND XIAOSHEN WANG, *A hybridized weak Galerkin finite element scheme for the Stokes equations*, Science China Mathematics, 58 (2015), pp. 2455–2472.
- [233] BIN ZHANG AND CHUNLEI LIANG, *A simple, efficient, and high-order accurate curved sliding-mesh interface approach to spectral difference method on coupled rotating and stationary domains*, Journal of Computational Physics, 295 (2015), pp. 147–160.
- [234] YC ZHOU AND GW WEI, *High resolution conjugate filters for the simulation of flows*, Journal of Computational Physics, 189 (2003), pp. 159–179.

Vita

Shinhoo Kang was born in South Korea on 1 April 1980, the son of Jongsup Kang and Okhui Kim. He received the Bachelor of Engineering in Electrical Engineering from the Handong Global University in Feb. 2007. He received the Master of Engineering in Spatial Design and Engineering from the Handong Global University in Aug. 2009. He joined the National Institute of Meteorological Research (NIMR) in Feb. 2010 and worked as a research scientist until Dec. 2011. He moved to Korea Institute of Atmospheric Prediction Systems (KIAPS) in Jan. 2012 and worked as a research scientist for developing the dynamical core of the global weather forecasting model in South Korea. He then left KIAPS in May 2014 and started his PhD program in Aerospace Engineering and Engineering Mechanics at the University of Texas at Austin in Aug. 2014.

Permanent address: sh.kang@utexas.edu

This Dissertation was typeset with L^AT_EX[†] by the author.

[†]L^AT_EX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's T_EX Program.