A SPECTRAL MULTIDOMAIN PENALTY METHOD SOLVER FOR ENVIRONMENTAL FLOW PROCESSES

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Jorge Alberto Escobar-Vargas

May 2012

A SPECTRAL MULTIDOMAIN PENALTY METHOD SOLVER FOR ENVIRONMENTAL FLOW PROCESSES

Jorge Alberto Escobar-Vargas, Ph.D.

Cornell University 2012

This work presents the details behind each step in the development of a framework for two-dimensional quadrilateral discontinuous Spectral Multidomain Penalty Method (SMPM) solvers for environmental flow processes: a shallow water equation (SWE) solver and an incompressible Navier-Stokes equations (NSE) (under the Boussinesq approximation) solver, with additional emphasis given to the associated pressure solver. The potential for environmental flow simulations through spectral methods is very strong since these methods are exponentially accurate, non-dissipative and non-dispersive. These characteristics translate into capturing the smallest resolved scales of the flow and the propagation of ocean/lake waves with minimum numerical error. In addition, the element-based capability of the method enables the appropriate resolution of the important scales of the processes being modeled, the localization of specific events, and the treatment of complex boundary conditions and geometries. Finally, the discontinuous character of the method add enhanced stability to the method for highly-nonlinear under-resolved simulations, an intrinsic characteristic of environmental flow simulations.

In the SWE solver, the SMPM is compared with a nodal discontinuous Galerkin method (DGM), where the equations are solved with an explicit SSP-RK34 method. The comparison is done by applying both methods to a suite of six commonly considered geophysical flow test cases; we also include results for a classical continuous Galerkin (i.e.,

spectral element) method for comparison. Both the analysis and numerical experiments show that the SMPM and DGM are essentially identical; both methods can be shown to be equivalent for very special choices of quadrature rules and Riemann solvers in the DGM along with special choices in the type of penalty term in the SMPM.

In the NSE solver time is discretized with a high-order fractional step projection method, where the non-linear advection and forcing terms are advanced explicitly via a stiffly stable scheme. After that, an implicit solution of a Poisson pressure equation (PPE) is solved in order to introduce the incompressibility constraint. In the final fractional time-step linear viscosity forces are also solved implicitly by means of a modified Helmholtz equation. Stability of the numerical scheme for under-resolved simulations at high Reynolds numbers is ensured through use of penalty techniques, spectral filtering, dealiasing, and strong adaptive interfacial averaging. Special attention is given to the solution of the PPE linear system of equations, where the fundamental building blocks of the PPE solver presented here are a Kronecker (tensor) product-based computation of the left null singular value of the non-symmetric SMPM-discretized Laplacian matrix and a custom-designed two-level preconditioner. Both of these tools are essential towards ensuring existence and uniqueness of the solution of the discrete linear system of equations and enabling its efficient iterative calculation. Accuracy, efficiency, and stability of the multidomain model are assessed through the solution of the Taylor vortex, lid-driven cavity flow and double shear layer. The propagation of a non-linear internal wave of depression type is also presented to assess the potential of the solver for the study of environmental stratified flows.

The availability of the quadrilateral SMPM solver allows the numerical investigation of a much broader range of environmental processes, namely those in streamwise,vertical

non periodic domains with both horizontal and vertical localization.

BIOGRAPHICAL SKETCH

I was born in Bogota (Colombia), went to the Colegio Calasanz followed by one year in the mandatory military service in the Colombian army. After that, I began my studies on Civil Engineering at the “Universidad Nacional de Colombia”, where I obtained a bachelor degree. My final project was a numerical simulation of the groundwater dynamics in the Fuquene lagoon valley, which is close to Bogota. Once I finished the project, I was intrigued about the “black box” behind the commercial software that I used, and that interest led me to go to Barcelona (Spain) to study a Master’s degree on Numerical Methods for Calculus and Design in Engineering at the “Universidad Politcnica de Cataluña”, where I began to build a strong background in the details behind numerical simulations. One day, after the classes were finished, the director of the program asked for students interested in doing the master’s thesis in Germany. I raised my hand, and I ended up at the Technical University of Munich doing a numerical analysis of degenerated high-order finite-elements for the p-version of the finite element method. Once I finished that project I tried to find a PhD over there, but I did not find a position, so I returned to Colombia and studied a Master’s degree on hydrosciences at the “Pontificia Universidad Javeriana”, and I worked as a lecturer in a small university and as an engineer at the Colombian Institute of Meteorological and Environmental Studies (IDEAM). At that time I was completely sure of my interest on numerical modeling of environmental flows, and that interest put me in contact with Prof. Peter Diamessis at Cornell. And here I am, writing my little biography after almost 6 years of development of an incompressible Navier-Stokes solver via the spectral multidomain penalty method for high Reynolds number flows. This solver is now being used to simulate environmental stratified flows processes in oceans and lakes.

Before I came to the USA I was a competitive Table Tennis player for 16 years, I repre-

sented my city and my country many times. Thanks to that, I know what discipline and focus can do in my life, as well as the importance of a parallel activity besides the daily routine. I don't play table tennis with the same intensity as before, but I play for the Cornell Table Tennis team in the regional tournaments, and besides that I ride my bike everyday, play squash and afro-cuban percussion, read latin-american literature, and I'm involved with some cultural and social organizations here in Ithaca.

This document is dedicated to the universal conspiracies that placed me in these circumstances, and to life, because it is worth living it.

ACKNOWLEDGEMENTS

- Peter Diamessis → A very uncommon friendly person with strong, and sometimes over-analyzed opinions. A Greek man who taught me the skills that I was looking for. A person that showed me what people can do with discipline, slow-deep thinking, no fear, and no arrogance. A sport minded man who treated me as a new coach treats his first serious player. My boss and friend during the most interesting journey of my life. I truly thank him, and I wish the best for him. Respect !!
- Charles Van Loan → The one who gave me the hand when I was going in free fall to the abyss. I additionally thank him for being part of my PhD committee.
- Prof. Stephen Pope → The most knowledgeable of the professors I have had. His motivation and rigor on fluid mechanics is remarkable. It was very challenging to be prepared for his questions at the hour of the examinations. I thank him for that, and for being part of my PhD committee.
- Derek Warner → A very nice person and sharp researcher with a balancing presence. His parallel computing class was a proof for the endurance needed to go through a computational project. I thank him for being very helpful during that semester and also for being part of my PhD committee.
- Todd Cowen → The one who made this possible (The Conspirator, my retainer).
- Frank Giraldo → The one who propose the SWE exercise. A very sharp person who knows how to do things. It was really nice to work with him.
- Jan Hesthaven → The final resource when nothing was working.
- Guus Jacobs → Remember that you are in an engineering department, so you have to chop some corners and make the code work !!.

- Atle Jensen → My squash partner during my finishing stage. The conversations and matches with him were the key to be grounded and balanced during the end of the PhD. I also thank him for recommend me the novel *The Growth of the Soil*.
- Paula Douglass → My friend who now lives in Boston, and was always with me during the first years of my time at Cornell.
- Marcela Monroy → For her inconditional emotional support during the first years of my PhD.
- Jonathan Kline → The grounding pole, a real guy.
- My parents → Because you don't choose them, and they are ALWAYS there.
- Diana Ojeda → Mi chica. With her I finally understood that life can be seen in many different ways, and all of them nice. Respect !
- Others → David Bindel (CS - Cornell), Luke Olson (UIUC), Paul Fischer (Argonne National Lab.), Jose Maria Gonzalez Ondina (CEE - Cornell).
- Cornell University → The sponsor No 1
- NSF - Ocean Sciences → The sponsor No 2

TABLE OF CONTENTS

Biographical Sketch . . . iii
Dedication . . . v
Acknowledgements . . . vi
Table of Contents . . . viii
List of Tables . . . xi
List of Figures . . . xii

1 Introduction 1
1.1 Environmental flows . . . 1
1.1.1 Large-scale environmental flows . . . 2
1.1.2 Small-scale environmental stratified flow processes . . . 5
1.2 High-order methods for incompressible high Re flows . . . 8
1.2.1 Fourier and Galerkin methods . . . 8
1.2.2 Spectral Multidomain Method . . . 11
1.3 Extending SMPM to two-dimensional doubly non-periodic domains . . . 12
1.4 A Poisson pressure equation solver . . . 13
1.5 Thesis structure . . . 15

2 The Spectral Multidomain Penalty Method 17
2.1 Spectral Multidomain Method . . . 17
2.2 Penalty formulation . . . 19

3 Shallow water equation solver 21
3.1 Governing Equations . . . 21
3.1.1 Conservative form of the SWE . . . 21
3.1.2 Linearized SWE . . . 22
3.1.3 Quasilinear form of the SWE . . . 23
3.2 Numerical Methods . . . 24
3.2.1 Spectral Multidomain Penalty Method (SMPM) . . . 24
3.2.2 Discontinuous Galerkin Method (DGM) . . . 30
3.2.3 Temporal Discretization . . . 36
3.3 Test cases: Description and Results . . . 36
3.3.1 Linear Problems . . . 39
3.3.2 Nonlinear Problems . . . 44
3.4 Discussion . . . 49
3.4.1 Mass and Energy Conservation . . . 49
3.4.2 Effect of time step on convergence and conservation properties . . . 51
3.4.3 Effect of Filtering . . . 53
3.4.4 Computational Efficiency and implementation . . . 55

4 Incompressible Navier Stokes solver **59**
4.1 Incompressible Stratified flow model . . . 59
4.1.1 Governing equations and Boundary Conditions . . . 59
4.2 Numerical Method . . . 61
4.2.1 Temporal discretization . . . 61
4.2.2 Spatial discretization . . . 63
4.2.3 Additional Stabilization Measures . . . 67
4.2.4 Discrete Pressure Poisson equation . . . 73
4.3 Numerical results . . . 74
4.3.1 Taylor vortex . . . 75
4.3.2 Lid-Driven Cavity flow . . . 76
4.3.3 Double shear layer . . . 79
4.3.4 Propagation of a solitary non-linear internal wave of depression type . . . 82

5 Poisson Solver **89**
5.1 The pressure Poisson equation . . . 89
5.1.1 Compatibility condition . . . 89
5.1.2 Non-uniqueness of the pressure Poisson equation's solution . . . 90
5.2 The penalty-based discrete pressure Poisson equation . . . 91
5.2.1 Penalty formulation at subdomain interfaces . . . 91
5.2.2 Penalty treatment at physical boundaries . . . 92
5.3 Properties of the discrete pressure Poisson equation . . . 93
5.3.1 The discrete Poisson pressure equation . . . 93
5.3.2 Singular value distribution of the Poisson matrix . . . 94
5.3.3 Compatibility condition revisited . . . 95
5.3.4 Consistency of the linear system of equations . . . 96
5.4 Null singular vector removal . . . 97
5.4.1 Commonly used strategies . . . 98
5.4.2 Strategies for the SMPM-discretized Poisson equation . . . 101
5.5 Computation of the left null singular vector . . . 105
5.5.1 Doubly-periodic domain . . . 106
5.5.2 2D non-periodic domain . . . 108
5.5.3 Computation of the core vectors . . . 111
5.5.4 Validation of the procedure . . . 113
5.6 Preconditioners . . . 115
5.6.1 Diagonal Jacobi preconditioner . . . 115
5.6.2 Block Jacobi (non-overlapping additive Schwarz) . . . 116
5.6.3 Two-Level preconditioner . . . 117
5.7 Numerical results . . . 118
5.7.1 Taylor vortex . . . 118
5.7.2 Lid-driven cavity flow . . . 119
5.8 Discussion . . . 121

6 Conclusions and future work **124**
6.1 Shallow Water equations 124
6.2 Incompressible Navier Stokes solver 125
6.3 Poisson Pressure equation solver 127
6.4 Future work 130

Bibliography **133**

LIST OF TABLES

I Coefficients for the third order - four stage SSP-RK(34) method 36

I Possible choices of the core vectors $\mathbf{u_x}$ and $\mathbf{u_z}$, used in the computation of $\mathbf{u_0}$, depending on the location of the subdomain under consideration, as shown in Fig. 5.9 110

II Possible choices of the core vectors $\mathbf{u_x}$ and $\mathbf{u_z}$ used in the computation of $\mathbf{u_0}$ for different choices of boundary conditions, recalled that $\mathbf{u_B} = \mathbf{u_L}$ and $\mathbf{u_T} = \mathbf{u_R}$ 113

LIST OF FIGURES

1.1 Schematic of the different discretization techniques used to solve numerically partial differential equations 8

2.1 Schematic of a subdomain subject to penalty treatment. I_1 denotes a point along an internal interface of the subdomain and B represents a point on a physical boundary. I_2 denotes the corresponding interfacial point of the subdomain neighboring point I_1 19

3.1 Vertical interface 25
3.2 Horizontal interface 26
3.3 Analysis of the standing wave (5×5 subdomains) at $t = 0.5$ seconds for a varying number of GLL points. a) L_2 normalized relative error in the free surface elevation h. b) L_2 normalized relative error in u velocity. c) Relative error in mass. d) Relative error in energy. 40
3.4 Kelvin wave results for 20×10 subdomains at $t = 5$. Panels (a) through (d) show the same quantities with Fig. 3.3. 42
3.5 Free surface elevation computed by all three methods for the linear Stommel problem for 5×5 subdomains and $N = 12$ at $t = 400$ days . . 43
3.6 Linear Stommel problem results for 5×5 subdomains at $t = 320$ days. a) Convergence plot for the L_2 normalized relative error in the free surface elevation h. b) Relative error in mass conservation. 44
3.7 Qualitative comparison of the Non-linear Rossby wave results with $24\times$ 8 subdomains, $N = 12$, and at time $t = 40$. 46
3.8 Non-linear Rossby wave results for 24×8 subdomains at $t = 40$. a) Relative error in mass. b) Relative error in energy. 46
3.9 Nonlinear Stommel problem for 5×5 subdomains and $N = 12$ at $t =$ 400 days 47
3.10 Relative error in mass as a function of polynomial order for the Nonlinear Stommel problem. 5×5 subdomains at $t = 360$ days. 48
3.11 Nonlinear Riemann problem for 5×5 subdomains at $t = 0.2$. Panels (a) and (b) are the same as Fig. 3.8 49
3.12 Comparison of conservation properties of the DGM for the Riemann problem. Results for 5×5 at $t = 0.2$. (a) Mass conservation. (b) Energy conservation. 51
3.13 Convergence plots for the standing wave problem when different Δt sizes are used. 52
3.14 Conservation of mass for the standing wave problem when different Δt sizes are used. 52
3.15 Conservation of energy for the standing wave problem when different Δt sizes are used. 53

3.16 Cross section of the Dam-break problem for 5×5 subdomains, and $N = 20$ at $t = 0.1$. (a) SMPM without dissipation. (b) SMPM with filtering (Filter order $p = 10$). (c) SMPM with dissipative term. (d) DGM without dissipative term ($|\lambda|(\mathbf{q}^{\mathbf{R}} - \mathbf{q}^{\mathbf{L}})$. (e) DGM with filtering (Filter order $p = 10$), and without dissipation term. (f) DGM full Rusanov flux. 54
3.17 Cross section of the filtered Dam-break problem for 5×5 subdomains, and $N = 20$ at $t = 0.1$ (Filter order $p = 10$) for SEM. 55
3.18 CPU time for the Riemann problem. 5×5 subdomains with different polynomial orders at $t = 0.2$. (a) All methods with Courant Number = 0.4 and (b) DGM and SMPM with Courant Number = 0.4 and SEM with Courant Number = 0.8. 56
3.19 Eigenvalue distribution of the 1D discrete linear advection operator ($u\frac{d}{dx}$) for all three methods, with an advective velocity of $u = 1$. In all cases $x \in [-1, 1]$, 5 subdomains, $N = 4$. 57

4.1 Taylor Vortex problem. Left: Structure of the streamlines of the flow. Right: L_∞-error norm of the horizontal velocity as a function of polynomial degree $N$and level of $h-$refinement (3×3, 5×5, and 10×10 subdomains) for $Re = 100$. 76
4.2 Streamlines on the Lid-driven cavity flow at $t = 20$ sec for 10×10 subdomains and $Re = 1000$. a) $n = 9$. b) $n = 12$. c) $n = 15$. 77
4.3 Streamlines on the Lid-driven cavity flow at $t = 20$ sec for a 10×10 grid with non-uniform subdomains, $n = 14$, filter order $p = 11$, and $Re = 1000$. 78
4.4 Vorticity contours for the dealiased solution for the double shear layer test case at $t = 2s$ for 30×30 subdomains, $n = 14$, $Re = 10^4$. Left: Filter order $p = 12$. Right: Filter order $p = 10$ 80
4.5 $p-$refinement convergence test for the double shear layer. Vorticity contours at $t = 2s$ for 30×30 subdomains and $Re = 10^4$. a) $n = 6$, b) $n = 9$, c) $n = 14$, d) $n = 19$. 81
4.6 Snapshots of density contours of propagating internal solitary wave: a) initial condition at $t = 0[s]$; b) at $t = 120[s]$ for $Re_w = 1.6 \times 10^5$; and c) at $t = 120[s]$ for $Re_w = 4 \times 10^4$. The wave propagates to the left, and the domain is periodic in $x-$direction. Ten equally-spaced contour levels in the range $[1005, 1035][kg/m^3]$ are shown. 85
4.7 Time histories of wave properties obtained from numerical simulations: a) wave amplitude normalized by its initial value; b) wavelength normalized by its initial value; c) wave travel normalized by the initial wavelength; and d) wave phase difference from the inviscid limit. . . . 86

5.1 Left panel: SMPM Gauss-Lobatto-Legendre grid on the left (3×3 subdomains with $N = 8$). Central panel: Structure of the corresponding Poisson matrix $\mathbf{A}$. Right panel: Structure of the contribution of each subdomain into the global matrix $\mathbf{A}$. 94

5.2 Singular value distribution of the Poisson matrix **A**: Case of 3×3 subdomains and $N = 8$ 95
5.3 Singular value distribution of the matrix $\mathbf{U_R A V_R^T}$, where **A** is the matrix of Fig. 5.2, obtained with the reduced system technique via Householder matrices. Unlike Fig. 5.2, the null singular value σ_0 is now absent. 103
5.4 Singular value distribution for the augmented system corresponding to the matrix **A** of Fig. 5.2. 104
5.5 Left null singular vector $\mathbf{u_0}$ for an example of a doubly periodic domain with 3×3 subdomains, and $N = 4$. The vertical dashed lines separate the contributions of individual subdomains, i.e. the subvector **sv** shown in 5.6. 106
5.6 Exploded view of the subvector **sv** for an example of a doubly periodic domain with 3×3 subdomains, and $N = 4$. 107
5.7 Structure of the vector $\mathbf{u_I}$ for the case of $N = 4$. 108
5.8 Structure of the of the left null singular vector $\mathbf{u_0}$ for an example non-periodic domain with 3×3 subdomains, and $N = 4$ 109
5.9 Schematic of a general non-periodic domain. It shows the possible locations of subdomain *i* subject to different patching/boundary conditions. 110
5.10 One dimensional base configuration for the generation of the left null singular vector $\mathbf{u_0}$ (case of $N + 1 = 5$). 111
5.11 Left null singular vector structure of the one dimensional discrete SMPM Poisson matrix. $\mathbf{u_L}$, $\mathbf{u_I}$, and $\mathbf{u_R}$ for the case of $N + 1 = 5$ 112
5.12 L_2 error norm (as compared to the corresponding MATLAB estimate) in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a doubly-periodic domain. 114
5.13 L_2 error norm in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a 2D non-periodic domain. 114
5.14 Poisson solver performance for different preconditioners for the Taylor vortex problem. Left panel shows the average number of pressure iterations per timestep as a function of total number of degrees of freedom (DOF). Right panel shows the average CPU time per timestep as a function of total number of degrees of freedom (DOF). In the legend, DJ, BJ and TL correspond to diagonal Jacobi, block-Jacobi and two-level preconditioners, respectively. 119
5.15 Poisson solver performance for the Block Jacobi (BJ) and Two-Level (TL) preconditioners applied to the Lid-driven Cavity flow. Left Panel: pressure iterations as a function of number of subdomains and polynomial degree *N* (White surface: BJ preconditioner. Grey surface: TL preconditioner). Right panel: CPU time per time step as a function of degrees of freedom (DOF) on the right. Also shown are least-squares power law best-fits. 121

CHAPTER 1

INTRODUCTION

1.1 Environmental flows

This work is motivated by the need of an improved physical understanding of environmental flows, namely the different flow processes in the natural environment, whose range of scales vary, and can contain scales, from the $O(1mm)$ to $O(10km)$. That means, these processes are much smaller that the ones present in a planetary scale, and much bigger than the ones present in a engineering/human scale. Examples of these processes are tsunamis, propagation of internal waves in the ocean and lakes, flow through aquatic vegetation, near shore hydrodynamics, lake dynamics, etc. Depending on the most representatives scales and processes of these flows, they can be divided on large scale and small scale environmental flows. Typically, these flow processes are not captured by larger-scale operational models (e.g. weather/ocean forecast), because they are below their resolution limits. Understanding the basic underlying physics of these processes allows one to lump them a lot more reliably in the above large-scale models.

One way to understand this type of processes is through the use of a mathematical model (governing equations), which is generally solved numerically with a computer code. Different numerical techniques have been developed to solve these equations or sets of equations. This work is focused on the development and implementation of a technique called Spectral Multidomain Penalty Method (SMPM), for the numerical solution of two different sets of equations: the Shallow Water Equations (SWE), and the incompressible Navier-Stokes (NS) equations under the Boussinesq approximation.

1.1.1 Large-scale environmental flows

Geophysical and large-scale environmental flows (i.e., flows of horizontal longitude of O(1 km) or greater) exhibit a complex structure and dynamics over a broad range of scales that render their numerical simulation a formidable task for state-of-the-art computational methods and resources. Through a complex interplay between the earth's rotation, ambient stratification and the constraining effects of lateral and vertical boundaries, flow processes in geophysical fluids commonly exhibit a characteristic horizontal lengthscale that can be a few orders of magnitude larger than its vertical counterpart [43]. Hydrostatic wave motions occur from the basin/planetary scale roughly down to the mesoscale. As the wave scales decreases, non-linear effects become significant in the form of internal/surface bores [90, 118]. At wavelengths of $O(1km)$, the waves also become strongly non-hydrostatic [63], localized turbulence occurs at the smaller scales and the dissipative effect of viscosity is ultimately felt at the smallest scales (O(1 mm)) of the flow field (see section 1.1.2 for more details).

As a result, the numerical methods used in the investigation of geophysical flows need to exhibit a number of preferred features. These include: a) front/wave propagation that is effectively non-dissipative and non-dispersive, b) minimum artificial dissipation at the smallest resolved scales to enable as broad a scale separation as possible, c) efficient resolution of localized flow features and complex geometries and d) optimal use of computational resources. High-order accurate element-based schemes [24, 68] are particularly appealing in addressing such needs. These schemes combine the exponential convergence and weak artificial dissipation and dispersion of standard single-domain spectral methods [12] with the spatial adaptivity of classical finite element/volume techniques [73, 120]. Furthermore, the domain decomposition philosophy inherent in these techniques renders them highly amenable for efficient parallelization [40].

On account of the inevitable impossibility of capturing the full range of scales intrinsic to a highly nonlinear, and steep, front/wave or any resulting localized turbulent event, and the minimal feedback obtained from the unresolved scales, geophysical flow simulations are inherently under-resolved. Under-resolved high-order simulations are prone towards, often catastrophic, numerical instability as Gibbs oscillations are compounded by aliasing driven by the nonlinear terms in the governing equations [53]. In high-order element-based simulations, these numerical instabilities are most pronounced at the element interfaces when strong continuity of the solution is enforced across neighboring elements [25] as is typically done in continuous Galerkin methods.

In discontinuous high-order element-based methods, neighboring subdomains carry separate values of the solution at a fixed spatial location thereby relaxing the constraint of strong continuity of the solution and significantly mitigating the above concerns of numerical instability. The two prevalent categories of such methods are spectral multidomain methods (with and without a penalty scheme) [80, 81, 83, 66, 64, 65, 28] and discontinuous Galerkin methods (DGM) [47, 46, 48, 51, 68, 49, 86, 103]. The formulation used in our work follows the Spectral Multidomain Penalty Method (SMPM) presented by Hesthaven [66] and expanded upon by Don [28] but implemented, to our knowledge for the first time, to the shallow water equations. In the SMPM, the strong interfacial patching conditions are replaced with a linear combination of the governing equation and the patching condition, the latter multiplied by an appropriately chosen penalty coefficient. On the other hand, DGM are based on a Galerkin weighted residual formulation where the integration is performed at the level of an individual element. Since adjacent elements are not continuously coupled, as is the case with finite and spectral elements, interfacial flux integrals do not vanish and are represented in the form of

an appropriately chosen numerical flux that preserves consistency and numerical stability.

SMPM have been successfully applied on the simulation of mainly hyperbolic equations that go from Euler equations [82], until compressible Navier-Stokes equations [83, 109]. DGM have been effectively used in the simulation of the shallow water equations (SWE) both on the sphere and on planar but fully unstructured domains [47, 46, 51, 86, 49] and for compressible atmospheric models [48, 103].

However, the literature exploring the similarities and differences of the SMPM and DGM is limited to the recent work by Gottlieb and Jung [55] who considered the modal form of SMPM and DGM, both in Galerkin (integral) formulation. Focusing on one-dimensional conservation laws, that particular study established the equivalence between the two techniques for a specific value of the penalty coefficient and emphasized the additional flexibility of the penalty scheme in varying the value of this coefficient in space and time and splitting the advective flux at the subdomain interfaces, which provided for greater stability in regions of strong inhomogeneity of subdomain thickness. The trade-offs of accuracy vs. stability as a function of the penalty coefficient value were also examined as was the potential of the coefficient truncation method [72] in suppressing rapid error growth when using high-order polynomials in the penalty method. Finally, the impact of inconsistent evaluation of integrals (exact versus numerical quadrature) in the left and right-hand sides of the modal Galerkin formulation of the penalty method was also considered in the framework of linear and nonlinear problems. Note that both the coefficient truncation method and the issues with integral evaluation are restricted to the modal Galerkin form of the SMPM.

No investigations are known so far that compare the collocation-based SMPM and the nodal Galerkin formulation of the DGM, the most commonly used formulations of the two methods which this work focuses on. Furthermore, we are unaware of any comparison of the two methods in the framework of a system of multi-dimensional equations, particularly in a geophysical context. Such a comparison is one of the objective of the present work. The platform for this comparison are the SWEs for a variety of reasons: a) the relative facility of their spatial and temporal discretization with respect to more complex partial differential equations, such as the Navier-Stokes equations, b) their capability for non-dissipative propagation of highly non-linear waves, which renders them an ideal experimentation tool for testing numerical schemes for nonlinear advection, the primary source of the aliasing-driven instabilities mentioned above and c) their role as a predictive tool of ocean wave phenomena for the purpose of coastal engineering applications [30] and tsunami propagation [3]. We specifically aim to compare the two methods in terms of formulation (with a focus on subdomain communication), accuracy, conservation properties, numerical stability and computational cost in the framework of specific linear and non-linear test-cases.

1.1.2 Small-scale environmental stratified flow processes

In the stably stratified portion of the water column of the ocean and lakes, flow processes operating over spatial scales of a kilometer or less are largely characterized by the complex interplay between internal gravity waves (IGW) and highly localized turbulence [119]. IGWs are a type of wave motion unique to stably stratified fluids and have wavelengths between 100m and 1km [115]. In the absence of instabilities in their interior and interactions with bottom/lateral boundaries, IGWs can transport energy non-dissipatively over large horizontal distances of $O(100km)$. A particular class of IGWs

are internal solitary waves (ISWs) large-amplitude, long internal solitary waves are horizontally propagating waves guided by the top and bottom surfaces of the ocean or lakes [63, 8]. Neglecting again internal instabilities and boundary interactions, ISWs not only propagate non-dissipatively over large distances; their propagation is non-dispersive, as manifested by the ability of the waves to maintain a very steep waveform, owing to an intrinsic balance between nonlinearity and physical dispersion.

Localized turbulent events contain a broad range of scales, extending from a largest scale of 1m to 10m to a smallest one of 1mm. This localized turbulence occurs through instabilities within an IGW and interactions of IGWs with topography, currents or a variable background stratification profile [119]. In the absence of IGWs, turbulence is also driven through instabilities in currents or interactions of currents with topography [119].

From the above discussion, it becomes apparent that the scale separation between IGWs and the larger-scales of the localized instabilities and turbulence occuring within or under the waves can be as high as three orders of magnitude. Moreover, a turbulent events itself contains a broad range of scales, quantified by an appropriately defined Reynolds number, $Re = uL/\nu$, where u and L are velocity and length scales, respectively, characteristic of the larger-scales of the turbulence, and ν is the kinematic viscosity of the fluid. Within this range of scales, extending from the larger energy-containing scales down to the dissipation range, exist motions of intermediate scale subject to nearly inviscid dynamics [117]. Finally, localized environmental turbulence is strongly non-hydrostatic, i.e. contains significant vertical accelerations. The degree of hydrostaticity in IGWs varies from case to case, with ISWs being the most extreme example, where non-hydrostatic effects provides the necessary physical dispersion that allows the waves to propagate unchanged over long distances. In the simulation of a strongly

non-hydrostatic fluid flow, the vertical pressure gradient term cannot be neglected and invoking the hydrostatic assumption, commonly used in larger-scale geophysical modeling [61] can lead to highly erroneous physical results.

As with large scale environmental flows (see section 1.1.1), higher-order accuracy element-based numerical methods [12, 24, 68] are a highly promising tool for the simulation of small-scale environmental flow processes. Their high (spectral) accuracy and minimal numerical dissipation, defining features of a global spectral discretization technique (e.g. Fourier or Chebyshev) render the smallest resolved scales of motion in a turbulent event free of any artificial damping. Moreover, the minimal numerical dispersion of these methods, enables wave propagation over long distances free of any spurious dispersive effects. The flexibility in local flow resolution, inherent in the element approach, allows for an efficient capturing of localized instabilities and turbulence.

Nevertheless, the large scale separation between waves and the instabilities/turbulence embedded within them and the broad range of localized turbulent bursts themselves require a prohibitively large number of degrees of freedom to be represented on even the most state-of-the-art available computational resources. As a result, any simulations at Reynolds numbers that are environmentall relevant, will be inevitably under-resolved. Scales where viscous damping is dominant will not be resolved. As a result, for a high-order accuracy scheme devoid of any numerical dissipation, aliasing effects, driven by the nonlinear term in the governing equations, will typically lead to catastrophic numerical instabilities [25]. In addition, when arbitrary, i.e. non-periodic, boundary conditions are desired, the treatment of non-hydrostatic effects, linked to iterative solution of an elleiptic equation for the pressure is a non-trivial process which is compounded by the ill-conditioning inherent in higher-order interpolating polynomials [24, 67, 33].

1.2 High-order methods for incompressible high Re flows

We now turn to a short review of the historical evolution of high-order element-based techniques and, whenever relevant, their application to the simulation of environmental flows. Before going into the details of these techniques, and in order to show them in a general context, Fig. 1.1 presents a schematic of the different discretization techniques for partial diferential equations, where the high-order techniques are shown in addition to the widely used low-order techniques.

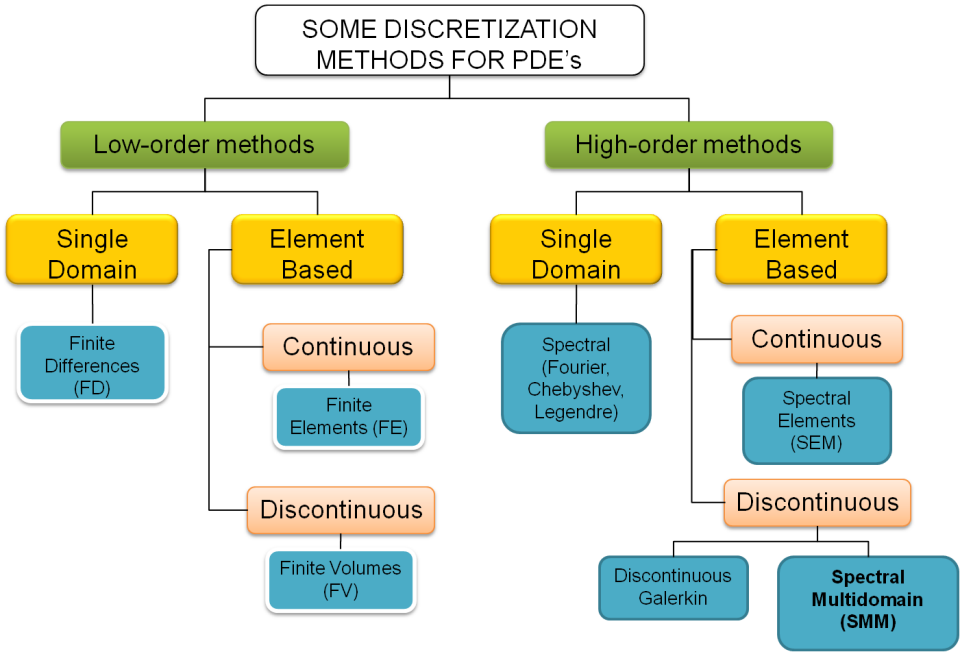


Figure 1.1: Schematic of the different discretization techniques used to solve numerically partial differential equations

1.2.1 Fourier and Galerkin methods

The application of high-order methods to Computational Fluid Dynamics (CFD) has been an active research topic since the early 70's, when computational power evolved

enough to handle a sufficiently large number of operations in an accurate manner. In 1972, Orszag and Patterson [96] were the first to present a numerical solution for the three-dimensional Navier-Stokes equations for homogeneous isotropic turbulence using a Galerkin approximation based on a globally defined Fourier-series representation of the flow field. The first well known element-based Galerkin approach, using locally-defined basis functions, was developed by Patera [99] who introduced the Spectral Element Method (SEM) and applied it to laminar flow in a channel expansion. Alumni of the Patera group developed two well-known SEM codes: the Nek5000 solver based on nodal SEM [41] and the Nektar solver based on the modal SEM approach [75, 76]. These solvers have evolved significantly from their original formulations and currently serve as major reference points for high-order element-based simulators of incompressible flows.

Fourier and SEM discretizations have been applied to a wide range of incompressible flow phenomena of fundamental, engineering and environmental relevance. [24, 75, 14, 15]. Traditionally the method of choice to simulate homogeneous isotropic turbulence [96], Fourier methods have also been used to simulate homogeneous anisotropic turbulence with the anisotropy caused by either background shear [70] or stable stratification [62, 13]. Fourier methods relying on sine/cosine transforms have also been used to simulate localized environmental stratified turbulent flows in domains subject to symmetric Dirichlet/Dirichlet or Neumann/Neumann boundary conditions at the top and bottom of the domain [56, 126]. In the case of environmental flows with more complex boundary conditions and boundary geometries, the SEM has successfully been used in the simulation of turbulent bottom density currents over currogated beds and the investigation of mixing in a lock exchange [97, 98]. Finally, note that in a geophysical/environmental context, the SEM has efficiently been used to discretize the shallow water equations

[88, 48].

The discontinuous Galerkin method (DGM) method was originally developed by Reed in 1973 [102]. It is a discontinuous variant of the SEM, where elements/subdomains do not share nodes with their neighbors along their interfaces forcing the solution to be discontinuous at these locations ; DGM enforces inter-element continuity only weakly. A "numerical flux" [68] must be specified along with the governing equation at the subdomain interfaces to enable communication between elements. Equivalently, boundary conditions are also also enforced weakly, i.e. in combination with the governing equations [84]. The discontinuous character in the form of weak inter-element continuity and boundary condition enforcement, allows for enhanced stability in the treatment of localized discontinuities and/or under-resolved, strongly nonlinear, flow simulations provided the numerical flux is appropriately specified [77, 68]. The DGM has been extensively used in the numerical solution of hyperbolic equations, such as the Maxwell, acoustic and shallow water equations [68, 34] (see also section 1.1.1). More recently, significant advancement has been made towards the effective use of DGM in the solution of elliptical PDE's (see Arnold [5] for a unified theory of DGM for elliptic problems). Application of DGM to the incompressible Navier Stokes equations has been rather limited [89, 110, 93]. DGM-based investigations of the shallow water equations [47] or the Boussinesq equations [30], to the authors' best knowledge, there exists no published work on the application of this method to incompressible (stratified or unstratified) environmental flow problems.

1.2.2 Spectral Multidomain Method

The Spectral Multidomain Method (SMM) is an extension of the single-domain collocation approach (Chebyshev or Legendre) to multiple domains and is also known as the "patching method" [15]. Connectivity across subdomains is enabled through a patching condition which is imposed at the interfacial points. This condition may be imposed strongly or weakly (i.e. combined with the governing equation), leading to a continuous or discontinuous solution (see below). As a result, although different in formulation and implementation, in its continuous or discontinuous form, the SMM is similar to either the SEM or DGM, respectively, in terms of accuracy and conservation properties.

The origins of SMM are from the late 70's and early 80's, when Metivet and Morchoisne [91] used it as an element-based approach for viscous flow calculations. In the mid 80's, Kopriva [80] was the first to present a detailed analysis of the two-dimensional multidomain approach. Later on, in the mid 90's, Hesthaven [66, 64, 65] introduced a multidomain penalty formulation, known as the spectral multidomain penalty method (SMPM). In the SMPM, the governing equation is penalized, in the sense that it is collocated at the physical boundaries/subdomain interfaces with the boundary/patching conditions, respectively. The range of allowable values of penalty parameters used in this approach is computed by requiring conservation of energy of the discretized equation (advectio, diffusion or advection diffusion) [66, 65].

In the framework of hyperbolic equations, the SMPM has been used in the numerical solution of the Euler equations in gas dynamics [82] and compressible viscous flows [83, 109]. A detailed comparison between the DGM, SMPM, and DGM in the context of the inviscid shallow water equations is presented in Chapter 3 and in [34]. In terms of environmental flow applications, Diamessis et al. [25] developed a SMPM solver

for the incompressible Navier-Stokes equations which was used to study a particular number of high-Reynolds number stratified flows such as wakes [27], bottom boundary layer instabilities under ISWs [26] and the propagation of internal wave packets [1]. The above solver can support a computational domain that is non-periodic in the vertical and periodic in the horizontal directions, with a Legendre-based SMPM discretization in the former and a Fourier discretization in the latter. Localized resolution and arbitrary boundary conditions are thus only possible in the vertical, thus limiting the range of environmental flow processes this solver can explored. Moreover, on account of incompressibility and the consideration of impermeable top/bottom boundaries, no Poisson equation with Neumann boundary conditions was solved for the pressure.

1.3 Extending SMPM to two-dimensional doubly non-periodic domains

The work presented in this thesis is motivated by the need to investigate, in a numerically stable and spectrally accurate manner, a broader range of environmental stratified flow processes at high Reynolds numbers, particularly those with localization and non-periodic boundary conditions in one of the horizontal directions. To this end, a Spectral Multidomain Penalty Method (SMPM) solver of the incompressible Navier-Stokes equations under the Boussinesq approximation has been developed. The solver is based on two-dimensional discontinuous quadrilateral subdomains with Gauss-Lobatto-Legendre (GLL) collocation points. We restrict ourselves to a two-dimensional quadrilateral subdomain discretization, as a third, periodic, direction may be readily introduced by using a Fourier discretization. In the numerical method, the penalty scheme is combined with dealiasing by padding [14], spectral filtering [7, 25], interfacial averaging

and a a high-order temporal discretization [74]. A fundamental difference of the solver presented here, with respect to the singly non-periodic solver created by Diamessis et al. [25], is the efficient iterative solution of a pressure Poisson equation. Details of this iterative solution procedure may be found in Chapter 5 and in [33]. The accuracy and stability of the new quadrilateral SMPM solver are successfully assessed against standard benchmarks, such as the Taylor vortex, modified lid-driven cavity and double shear layer. From an environmental fluid mechanics standpoint, the propagation of an ISW, which is an exact solution to the incompressible Euler equations, is investigated in a two-layer continuously stratified free-slip channel, showing negligible numerical dissipation and dispersion.

1.4 A Poisson pressure equation solver

The time-discretization, originally proposed by Karniadakis and co-workers [74] (hereafter referred to as KIO, and described in Chapter 4), used in the above SMPM model requires the solution of a Poisson equation for the pressure with Neumann boundary conditions. Moreover, on account of the broad range of scales in environmental stratified flow processes, any associated simulation will involve a very large number of degrees of freedom (DOF) and the numerical solution of the linear system of equations corresponding to the pressure Poisson equation (PPE) can only be performed iteratively.

The matrix resulting from the SMPM discretization of the Poisson-Neumann problem is ill-conditioned for two reasons: a) the inherent ill-conditioning of higher-order interpolating polynomials and b) the ill-posedness of the corresponding analytical equation, whose solution can only be determined up to an additive constant. Both of these factors pose significant challenges to the iterative solution of the PPE. Moreover, existence of

a solution requires the satisfaction, at the spatially analytical level, of an integral compatibility condition between boundary conditions and right hand side of the PPE [101]. In the KIO scheme, the compatibility condition is inherently satisfied at the spatially continuous level [74]. However, under-resolution and the presence of the penalty terms can cause a violation of the compatibility condition (see reference [54] and §5.3.3 of this document), thereby posing an additional major challenge to the iterative solution of the PPE.

The above challenges in the iterative solution of the linear system associated with the PPE, or the Stokes equation resulting from alternative time discretizations of the incompressible NSE [24], have been efficiently addressed through the development of appropriate preconditioning techniques [38, 42, 112]. All these techniques are designed for the symmetric matrices resulting directly from the Galerkin formulation of SEM. Extensive background on the numerical solution of symmetric linear systems of equations can already be found in the numerical linear algebra literature.

However, the matrix resulting from the SMPM discretization of the PPE is non-symmetric on account of the use of a collocation discretization [84]. When examining the numerical linear algebra literature, one observes a paucity of tools for preconditioning, matrix singularity treatment and solvability condition enforcement (the matrix-level equivalent of the compatibility condition) for linear systems with non-symmetric matrices.

Motivated by the above observations and the need to study environmental flow processes of increasing complexity, the last chapter of this work presents strategies developed for the efficient iterative solution of the SMPM-discretized PPE with Neumann boundary

conditions resulting from application of the KIO splitting scheme to the incompressible NSE. The fundamental building block of these strategies is a fast computation of the left null singular vector of the global Poisson matrix. Consistency of the associated linear system of equations, paramount to the robust performance of the iterative GMRES solver, can only be ensured if this left null singular vector is available. In addition, a method for removing the null singular value of the Poisson matrix is outlined, which also relies of the availability of the the left null singular vector. This method is contrasted, in terms of accuracy and robustness within the GMRES framework, to other more commonly used techniques designed to ensure a unique solution to the Poisson-Neumann problem. A custom-designed two-level preconditioner is also presented and its superiority is demonstrated with respect to diagonal Jacobi and block-Jacobi preconditioners. Finally, the efficiency of the Poisson solver, as buttressed by all the above strategies, is assessed through its application to the solution of two commonly considered benchmark problems.

1.5 Thesis structure

This dissertation is a compilation of three research papers written during the development of a incompressible Navier Stokes solver via SMPM. These papers were joined together in a document with the following structure: In Chapter 2 the basic definition of the Spectral Multidomain Method is given, as well as the form the penalty treatment is imposed on an arbitrary governing equation. In Chapter 3, the first paper is presented [34] where a comparison of the numerical solution of the Shallow Water Equations for discontinuous high-order methods is analyzed in terms of the SMPM, and DGM. Chapter 4 presents a SMPM solver for high Reynolds number stratified incompressible flows, which is the second paper [32]. In Chapter 5, the details of the most demanding com-

ponent of the NS solver, the numerical solution of the pressure Poisson equation, is presented [33]. Finally, the concluding remarks and future work are presented.

CHAPTER 2

THE SPECTRAL MULTIDOMAIN PENALTY METHOD

2.1 Spectral Multidomain Method

This method is based on a collocation approach over multiple two dimensional quadrilateral subdomains (elements). On each subdomain, any function $u(x, z, t)$ is approximated with a tensor product of its nodal (i.e. Lagrange) basis functions over a two dimensional Gauss-Lobatto-Legendre (GLL) grid. Specifically, $N + 1$ collocation points are used on the grid in each direction such that [65]:

$$u(x, z, t) \approx \sum_{i=0}^{N} \sum_{j=0}^{N} u(x_i, z_j, t) l_i(x) l_j(z), \tag{2.1}$$

where $l_i(x), l_j(z)$ are the $i-$th and $j-$th Lagrange interpolating polynomials evaluated at each one of the GLL points, at the horizontal (x) and vertical (z) direction, within the quadrilateral element. Extension of Eq. (2.1) to three dimensions is immediate, but it is not addressed in this work since its scope is the simulation of two-dimensional flows.

In the collocation approach, on a one-dimensional element, the m-th discrete derivative of a function u is approximated by means of spectral differentiation matrices $\mathbf{D}_N^m$ [23] as

$$\frac{\partial^m u(x_i, t)}{\partial x^m} = \frac{\partial^m u(x_i, t)}{\partial \xi^m} \left(\frac{\partial \xi}{\partial x}\right)^m \approx \left(\frac{\partial \xi}{\partial x}\right)^m \sum_{k=0}^{N} d_{ik}^m u(x_k, t) = J_x^m \mathbf{D_N^m u}, \tag{2.2}$$

Since GLL points are defined over the canonical interval $\xi \in [-1, 1]$, the term $J_x = d\xi/dx$ in Eq. (2.2) represents the mapping/Jacobian from ξ to the global coordinate system represented by x and z. The entries, d_{ij}^m, of the Legendre spectral differentiation matrix, $\mathbf{D}_N^m$, are computed using the algorithm outlined in Costa and Don [23]. The properties of these matrices are discussed in detail in [6].

The extension from the one-dimensional single subdomain set-up (Eq. (2.2)) to a two-dimensional multidomain framework is straightforward if one takes advantage of the tensor (Kronecker) product structure of the multidomain collocation approach on structured grids. Consequently [33],

- One-dimensional multidomain:

$$\frac{d^m \mathbf{u}}{dx^m} = (\mathbf{L}_{nx} \otimes \mathbf{D}_N^m)\,\mathbf{u} \tag{2.3}$$

- Two-dimensional single domain:

$$\frac{\partial^m \mathbf{u}}{\partial x^m} = J_x^m\,(\mathbf{I}_N \otimes \mathbf{D}_N^m)\,\mathbf{u} \tag{2.4}$$

$$\frac{\partial^m \mathbf{u}}{\partial z^m} = J_z^m\,(\mathbf{D}_N^m \otimes \mathbf{I}_N)\,\mathbf{u} \tag{2.5}$$

- Two-dimensional multidomain:

$$\frac{\partial^m \mathbf{u}}{\partial x^m} = (\mathbf{I}_{nz} \otimes \mathbf{L}_{nx} \otimes \mathbf{I}_N \otimes \mathbf{D}_N^m)\,\mathbf{u} \tag{2.6}$$

$$\frac{\partial^m \mathbf{u}}{\partial z^m} = (\mathbf{L}_{nz} \otimes \mathbf{I}_{nx} \otimes \mathbf{D}_N^m \otimes \mathbf{I}_N)\,\mathbf{u} \tag{2.7}$$

In Eqs. (2.6) and (2.7), $\mathbf{I}_i$ are identity matrices of dimension i, nx, nz represent the number of subdomains in the horizontal and vertical directions respectively, and $\mathbf{L}_i$ are diagonal matrices, whose entries are the Jacobians of each subdomain in the x and z directions. As a result, $\mathbf{I}_N \otimes \mathbf{D}_N^m$ and $\mathbf{D}_N^m \otimes \mathbf{I}_N$ accounts for the horizontal and vertical derivatives within each subdomain, respectively. Additionally, $\mathbf{I}_{nx}$ aggregates the contribution of these derivatives across all subdomains in the $x-$direction. Finally, $\mathbf{I}_{nz}$ lumps together the equivalent contributions in the $z-$direction.

Eqs. (2.6) and (2.7) allow the explicit evaluation of a discrete derivative in either direction on the computational domain. When a spectral differentiation matrix is used in

an implicit solve in the spectral multidomain framework, as is done with the $\mathbf{D_N}^2$ matrix for the Poisson equation(Eq. (4.12)) or the modified Helmholtz equation to solve for the viscosity term (Eq. (4.10)), the above expressions should be augmented with the necessary penalty terms to account for the communication between subdomains.

Approximating derivatives in the form of Eqs.(2.6) and (2.7) poses additional challenges in solving iteratively the Poisson and Helmholtz equations mentioned above since the differentiation matrices are inherently non-symmetric. In Galerkin type methods, such as SEM and DGM, this is not the case since most of the global matrices are symmetric [84].

2.2 Penalty formulation

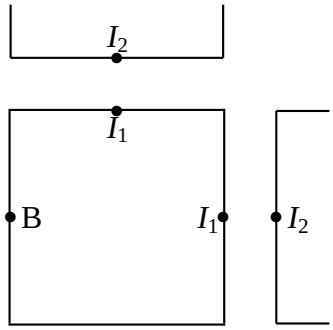


Figure 2.1: Schematic of a subdomain subject to penalty treatment. I_1 denotes a point along an internal interface of the subdomain and B represents a point on a physical boundary. I_2 denotes the corresponding interfacial point of the subdomain neighboring point I_1

For the sake of illustration the general form of the penalty formulation will be introduced with the Poisson equation as example. In this formulation, for any collocation point I_1, I_2 located along any subdomain interface or physical boundary (see Fig. 2.1),

the Poisson equation is recast as

$$\nabla^2 p + \tau \cdot [condition] = f, \tag{2.8}$$

where τ is a penalty coefficient, and the *condition* term in (2.8) represents the patching or boundary condition at the subdomain or physical boundary, respectively. Depending on the type of governing equation, this term takes different forms, as can be seen in Chapter 3 for the case of the Shallow Water Equations, and in Chapter 4 and 5 for the advection, diffusion, and Poisson equations respectively. The weak enforcement of the patching or boundary condition provides for enhanced stability of the numerical scheme. Moreover, in this work, we restrict our presentation to rectangular subdomains, although the extension to arbitrary quadrilaterals is straightforward [65].

For comparison purposes, it can be said that whereas in the SMPM the patching terms are imposed directly at the subdomain interfaces due to its collocation based formulation, in the DGM they are incorporated in the form of numerical fluxes at the boundary integrals that appear during the weighted residual approach treatment done to the governing equations. For details on the DGM formulation of the numerical fluxes, the reader is referred to [68, 84], and in the context of the inviscid shallow water equations to Chapter 3.

CHAPTER 3

SHALLOW WATER EQUATION SOLVER [1]

3.1 Governing Equations

The inviscid shallow water equations (SWE) govern the behavior of a fluid with a horizontal extent much larger than its depth, and are derived by applying the hydrostatic approximation to the incompressible Navier-Stokes equations [85]. The primitive variable formulation of the SWE is given by

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} - Z(u,v) = -g\frac{\partial h}{\partial x} \tag{3.1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + Z(u,v) = -g\frac{\partial h}{\partial y} \tag{3.2}$$

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left[(H+h)u\right] + \frac{\partial}{\partial y}\left[(H+h)v\right] = 0 \tag{3.3}$$

where u, v are the horizontal velocities, H is the mean depth, h is the displacement of the free surface, $Z(u,v)$ is the external forcing and g is the gravitational constant.

3.1.1 Conservative form of the SWE

The inviscid shallow water equations (equations (3.1),(3.2) and (3.3)) can also be written in conservative form:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{q})}{\partial y} = \mathbf{S}(\mathbf{q}), \tag{3.4}$$

[1] The contents of this chapter are published on the article [34] *High-order discontinuous element-based schemes for the inviscid shallow water equations: Spectral multidomain penalty and discontinuous Galerkin methods* written by Jorge Escobar-Vargas, Peter Diamessis and Frank Giraldo

where the conservative variables **q** are

$$\mathbf{q} = \begin{bmatrix} \phi \\ \phi u \\ \phi v \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{3.5}$$

the horizontal and vertical fluxes **F**(**q**) and **G**(**q**) are defined as

$$\mathbf{F}(\mathbf{q}) = \begin{bmatrix} \phi u \\ \phi u^2 + \frac{1}{2}\phi^2 \\ \phi uv \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}, \quad \mathbf{G}(\mathbf{q}) = \begin{bmatrix} \phi v \\ \phi uv \\ \phi v^2 + \frac{1}{2}\phi^2 \end{bmatrix} = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix} \tag{3.6}$$

and the source terms **S**(**q**) are

$$\mathbf{S}(\mathbf{q}) = \begin{bmatrix} 0 \\ f\phi v + \frac{\tau_x}{\rho} - \gamma\phi u \\ -f\phi u + \frac{\tau_y}{\rho} - \gamma\phi v \end{bmatrix}. \tag{3.7}$$

In Eqs. (3.5), (3.6) and (3.7), $\phi = gh$ is the geopotential height, $f = f_0 + \beta(y - y_m)$ is the Coriolis force, τ_x, τ_y are the components of the wind stress, ρ is the fluid density, and γ is a bottom friction constant.

3.1.2 Linearized SWE

Assuming a mean depth much larger than the free surface elevation ($H >> h$), and neglecting the nonlinear terms in (3.4), a linearized version of the conservative SWE is obtained. The modified set of conservation variables is defined as

$$\mathbf{q} = \begin{bmatrix} \phi \\ \Phi u \\ \Phi v \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}, \quad \mathbf{F}(\mathbf{q}) = \begin{bmatrix} \Phi u \\ \Phi\phi \\ 0 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}, \quad \mathbf{G}(\mathbf{q}) = \begin{bmatrix} \Phi v \\ 0 \\ \Phi\phi \end{bmatrix} = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix} \tag{3.8}$$

where $\Phi = gH$ is the mean depth geopotential height.

3.1.3 Quasilinear form of the SWE

Using the chain rule, Eq. (3.4) can be rewritten in the quasi-linear form [120, 4]

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{q})}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{q})}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial y} = \mathbf{S}(\mathbf{q})$$
$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{q}}{\partial x} + \mathbf{B}\frac{\partial \mathbf{q}}{\partial y} = \mathbf{S}(\mathbf{q}) \tag{3.9}$$

where **A** and **B** are the flux Jacobian matrices, that can be decomposed (via an eigendecomposition or characteristic decomposition) as

$$\mathbf{A} = \mathbf{S_A}\mathbf{\Lambda_A}\mathbf{S_A^{-1}} \tag{3.10}$$

$$\mathbf{B} = \mathbf{S_B}\mathbf{\Lambda_B}\mathbf{S_B^{-1}} \tag{3.11}$$

where $\mathbf{\Lambda_A}$ and $\mathbf{\Lambda_B}$ are diagonal matrices containing the eigenvalues of **A** and **B**, and $\mathbf{S_A}$, $\mathbf{S_B}$ are orthogonal matrices whose columns are the respective eigenvectors.

The positive and negative flux vectors ($\mathbf{F}^+$, $\mathbf{F}^-$, $\mathbf{G}^+$, $\mathbf{G}^-$) are defined by

$$\mathbf{F}^+ = \int \mathbf{S_A}\mathbf{\Lambda_A^+}\mathbf{S_A^{-1}}dq \tag{3.12}$$

$$\mathbf{F}^- = \int \mathbf{S_A}\mathbf{\Lambda_A^-}\mathbf{S_A^{-1}}dq \tag{3.13}$$

$$\mathbf{G}^+ = \int \mathbf{S_B}\mathbf{\Lambda_B^+}\mathbf{S_B^{-1}}dq \tag{3.14}$$

$$\mathbf{G}^- = \int \mathbf{S_B}\mathbf{\Lambda_B^-}\mathbf{S_B^{-1}}dq \tag{3.15}$$

where $\mathbf{\Lambda_A^\pm}$ and $\mathbf{\Lambda_B^\pm}$ are the diagonal matrices composed of positive and negative eigenvalues of **A** and **B**, respectively. Based on the above decomposition, the flux vectors have the properties

$$\mathbf{\Lambda_A} = \mathbf{\Lambda_A^+} + \mathbf{\Lambda_A^-} \quad \rightarrow \quad \mathbf{F} = \mathbf{F}^+ + \mathbf{F}^- \tag{3.16}$$

$$\mathbf{\Lambda_B} = \mathbf{\Lambda_B^+} + \mathbf{\Lambda_B^-} \quad \rightarrow \quad \mathbf{G} = \mathbf{G}^+ + \mathbf{G}^-. \tag{3.17}$$

The eigenvalue matrices and flux vectors are the building blocks for the penalty formulation of the SWE via SMPM, and for the definition of the numerical flux of the DGM used in this work [47].

3.2 Numerical Methods

3.2.1 Spectral Multidomain Penalty Method (SMPM)

The SMPM implemented in this work is based on the formulation first introduced by Hesthaven [65] (see Chapter 2) and further refined by Don et al. [28]. Specifically, this SMPM consists of a multidomain collocation approach based on discontinuous non-overlapping rectangular subdomains that are connected by a penalty term that ensures stability of the solution by imposing weak continuity at the subdomain interfaces.

On account of the intrinsic discontinuity of the method and the critical role of interfacial patching, the penalized form of the SWE at a collocation point located along the boundaries of a subdomain requires that (see reference [28] for a similar formulation of the compressible Navier Stokes equations for chemically reacting flow)

$$\begin{aligned} \frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{q})}{\partial y} &= \mathbf{S}(\mathbf{q}) \\ &+ \tau_1 Q(\mathbf{x})[\mathbf{F}^+(\mathbf{q}) - \mathbf{F}^+(\mathbf{q}*)] \\ &+ \tau_2 Q(\mathbf{x})[\mathbf{F}^-(\mathbf{q}) - \mathbf{F}^-(\mathbf{q}*)] \\ &+ \tau_3 Q(\mathbf{x})[\mathbf{G}^+(\mathbf{q}) - \mathbf{G}^+(\mathbf{q}*)] \\ &+ \tau_4 Q(\mathbf{x})[\mathbf{G}^-(\mathbf{q}) - \mathbf{G}^-(\mathbf{q}*)]. \end{aligned} \tag{3.18}$$

In (3.18), τ_i ($i = 1, \cdots, 4$) are the penalty coefficients, $Q(\mathbf{x})$ act effectively as Dirac delta functions that are non-zero only at the interfaces of the subdomain, where the penalty terms are active, and $\mathbf{F}^{\pm}(\mathbf{q}), \mathbf{G}^{\pm}(\mathbf{q}), \mathbf{F}^{\pm}(\mathbf{q}*)$, and $\mathbf{G}^{\pm}(\mathbf{q}*)$ represent the positive and negative fluxes at the grid points on the particular interfaces of the subdomain (with $*$ indicating the corresponding point on the neighboring interface) on the subdomain under consideration. In a general sense, the penalty coefficients can be viewed as weighting factors for the positive and negative fluxes across the interfaces.

In what follows, the penalized form of the SWE will be presented for the case of structured quadrilateral grids with rectangular subdomains, where the treatment for vertical interfaces is determined by the horizontal fluxes $\partial\mathbf{F}/\partial x$, and for the horizontal interfaces by the vertical fluxes $\partial\mathbf{G}/\partial y$. Embedded in the penalty coefficients τ_i ($i = 1, \cdots, 4$) are mapping factors to enable consistency in units between the different terms in Eq. (3.18).

Vertical interfaces Figure 3.1 presents a schematic of the vertical interface between subdomains I and II, where L or R represent any collocation point at the left and right edges of the interface.

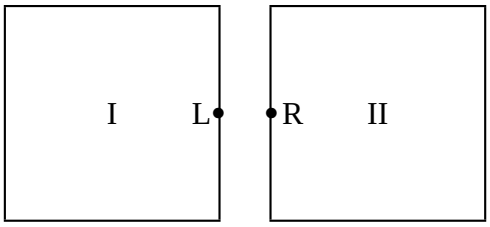


Figure 3.1: Vertical interface

Based on (3.18), the penalized form of the SWE for a point located at the left edge of

the interface is

$$\begin{aligned}\frac{\partial \mathbf{q}^L}{\partial t}+\frac{\partial \mathbf{F}^L}{\partial x}+\frac{\partial \mathbf{G}^L}{\partial y} &= \mathbf{S}(\mathbf{q})^L \\ &+ \tau_1 Q_L[(\mathbf{F}^+)^L-(\mathbf{F}^+)^R] \\ &+ \tau_2 Q_L[(\mathbf{F}^-)^L-(\mathbf{F}^-)^R]. \end{aligned} \tag{3.19}$$

Similarly, for a point along the right edge of the interface the penalized form is

$$\begin{aligned}\frac{\partial \mathbf{q}^R}{\partial t}+\frac{\partial \mathbf{F}^R}{\partial x}+\frac{\partial \mathbf{G}^R}{\partial y} &= \mathbf{S}(\mathbf{q})^R \\ &+ \tau_5 Q_R[(\mathbf{F}^+)^R-(\mathbf{F}^+)^L] \\ &+ \tau_6 Q_R[(\mathbf{F}^-)^R-(\mathbf{F}^-)^L]. \end{aligned} \tag{3.20}$$

In Eq. (3.20) τ_5, τ_6 are the corresponding penalty coefficients for the right edge of the interface.

Horizontal interfaces Figure 3.2 presents a schematic of a horizontal interface between subdomains I and III. In this case, B and T represent the collocation points along the bottom and top edges of the interface. The penalized equations for a point located at

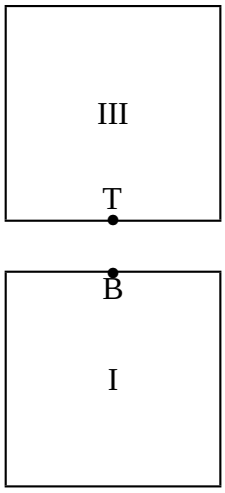


Figure 3.2: Horizontal interface

the bottom edge of the horizontal interface are

$$\begin{aligned}\frac{\partial \mathbf{q}^B}{\partial t}+\frac{\partial \mathbf{F}^B}{\partial x}+\frac{\partial \mathbf{G}^B}{\partial y} &= \mathbf{S}(\mathbf{q})^B \\ &+ \tau_3 Q_B[(\mathbf{G}^+)^B-(\mathbf{G}^+)^T] \\ &+ \tau_4 Q_B[(\mathbf{G}^-)^B-(\mathbf{G}^-)^T] \end{aligned} \tag{3.21}$$

whereas for a point located on the top side are

$$\begin{aligned}\frac{\partial \mathbf{q}^T}{\partial t}+\frac{\partial \mathbf{F}^T}{\partial x}+\frac{\partial \mathbf{G}^T}{\partial y} &= \mathbf{S}(\mathbf{q})^T \\ &+ \tau_7 Q_T[(\mathbf{G}^+)^T-(\mathbf{G}^+)^B] \\ &+ \tau_8 Q_T[(\mathbf{G}^-)^T-(\mathbf{G}^-)^B]. \end{aligned} \tag{3.22}$$

In Eq. (3.22) τ_7, τ_8 are the corresponding penalty coefficients for the top edge of the interface.

The approach of Don et al. [28, 71] for a one-dimensional conservation law can be readily extended to the penalized equations (3.19)-(3.22) to show that the penalty scheme formally conserves mass. Moreover, the energy of the system can been shown to be bounded by its initial value [28, 71] if

$$\begin{aligned} 2\omega^L\tau_1 \le 1, \qquad & 2\omega^L\tau_2 \ge 1 \\ 2\omega^B\tau_3 \le 1, \qquad & 2\omega^B\tau_4 \ge 1 \\ 2\omega^R\tau_5 \le -1, \qquad & 2\omega^R\tau_6 \ge -1 \\ 2\omega^T\tau_7 \le -1, \qquad & 2\omega^T\tau_8 \ge -1 \\ \omega^L\tau_1-\omega^R\tau_5 = 1, \qquad & \omega^L\tau_2-\omega^R\tau_6 = 1 \\ \omega^B\tau_3-\omega^T\tau_7 = 1, \qquad & \omega^B\tau_4-\omega^T\tau_8 = 1 \end{aligned}$$

where ω^L, ω^B, ω^R and ω^T are the GLL quadrature weights assigned to points along the left, bottom, right and top interfaces, respectively. For a uniform order of polynomial

approximation, N, in each subdomain a single value of $\omega = 2/N(N+1)$ can be used instead.

Implementation Issues In this work, the averaging method [28, 16] is implemented such that the penalty coefficients for positive and negative fluxes (Eqs. (3.19)-(3.22)) at the sides of the interfaces are taken to be equal. This leads to

$$\begin{aligned} \tau_L = \tau_1 = \tau_2 &= \frac{1}{2\omega}\frac{\partial \xi}{\partial x} \\ &= \frac{1}{\omega \Delta x} \end{aligned} \tag{3.23}$$

$$\begin{aligned} \tau_B = \tau_3 = \tau_4 &= \frac{1}{2\omega}\frac{\partial \eta}{\partial y} \\ &= \frac{1}{\omega \Delta y} \end{aligned} \tag{3.24}$$

$$\begin{aligned} \tau_R = \tau_5 = \tau_6 &= -\frac{1}{2\omega}\frac{\partial \xi}{\partial x} \\ &= -\frac{1}{\omega \Delta x} \end{aligned} \tag{3.25}$$

$$\begin{aligned} \tau_T = \tau_7 = \tau_8 &= -\frac{1}{2\omega}\frac{\partial \eta}{\partial y} \\ &= -\frac{1}{\omega \Delta y} \end{aligned} \tag{3.26}$$

where $\partial \xi/\partial x, \partial \eta/\partial y$ are the mapping factors for the penalty terms acting on vertical and horizontal interfaces respectively (see Eqs. (3.46) and (3.47)). This approach ensures stability of the penalty scheme. Moreover, the positive and negative fluxes of Eqs. (3.16) and (3.17), have been lumped into a single total flux in the penalty term.

The penalized SWE (eqs. (3.19)- (3.22)) may now be recast accordingly for each possible orientation of subdomain interfaces:

- Vertical interfaces

– Left edge of the interface

$$\frac{\partial \mathbf{q}^L}{\partial t} + \frac{\partial \mathbf{F}^L}{\partial x} + \frac{\partial \mathbf{G}^L}{\partial y} = \mathbf{S}(\mathbf{q})^L + \tau_L Q_L[\mathbf{F}^L - \mathbf{F}^R] \tag{3.27}$$

– Right edge of the interface

$$\frac{\partial \mathbf{q}^R}{\partial t} + \frac{\partial \mathbf{F}^R}{\partial x} + \frac{\partial \mathbf{G}^R}{\partial y} = \mathbf{S}(\mathbf{q})^R + \tau_R Q_R[\mathbf{F}^R - \mathbf{F}^L] \tag{3.28}$$

- Horizontal interfaces

– Bottom edge of the interface

$$\frac{\partial \mathbf{q}^B}{\partial t} + \frac{\partial \mathbf{F}^B}{\partial x} + \frac{\partial \mathbf{G}^B}{\partial y} = \mathbf{S}(\mathbf{q})^B + \tau_B Q_B[\mathbf{G}^B - \mathbf{G}^T] \tag{3.29}$$

– Top edge of the interface

$$\frac{\partial \mathbf{q}^T}{\partial t} + \frac{\partial \mathbf{F}^T}{\partial x} + \frac{\partial \mathbf{G}^T}{\partial y} = \mathbf{S}(\mathbf{q})^T + \tau_T Q_T[\mathbf{G}^T - \mathbf{G}^B] \tag{3.30}$$

Note that, in this scheme, unlike Hesthaven [65] no special formulation is used for the corners, which are simply treated as points that belong to two edges of the same subdomain orthogonal to each other. This simplified approach is found to be more stable than the theoretically derived one. In addition, the formulation of the penalty term is the same form used by Hesthaven [64, 65], Don et al. [28] and Diamessis et al. [25]. Variations of this formulation are possible and a particular one, involving the incorporation of dissipative Rusanov flux-like term, is examined in more detail in section 3.4.3.

Compact Representation of the SMPM A compact form of representing Eqs. (3.27) - (3.30) is

$$\frac{\partial \mathbf{q}^e}{\partial t} + \frac{\partial \mathbf{F}^e}{\partial x} + \frac{\partial \mathbf{G}^e}{\partial y} = \mathbf{S}(\mathbf{q})^e + \sum_{l=1}^{4} \widehat{\tau}_e Q_e \boldsymbol{n}^{(e,l)} \cdot [\mathbf{F}^e - \mathbf{F}^l] \tag{3.31}$$

where $\boldsymbol{n}^{(e,l)}$ is the outward pointing unit vector in the direction from control volume e to l,

$$\widehat{\tau} \equiv |\tau| = \frac{1}{\omega \Delta s}$$

with $\Delta s = (\Delta x, \Delta y)$ depending on the orientation of the subdomain interfaces.

3.2.2 Discontinuous Galerkin Method (DGM)

The discontinuous Galerkin (DG) discretization of SWE (3.4) is as follows: we begin with the governing equations in continuous flux-form

$$\frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{F}(\boldsymbol{q}) = S(\boldsymbol{q}). \tag{3.32}$$

Next we introduce a basis function expansion

$$\boldsymbol{q}_N(\boldsymbol{x}) = \sum_{i=1}^{(N+1)^2} \psi_i(\boldsymbol{x}) \boldsymbol{q}_i \tag{3.33}$$

where ψ represents the basis functions of order N and $\boldsymbol{q}_i$ are the solution variables at specially chosen interpolation points; in this work they are chosen to be the Gauss-Legendre-Lobatto (GLL) points in order to make the comparison with the SMPM more relevant and because we have used these points in previous DG formulations (e.g., [47, 48]). Using Eq. (3.33) we can now construct approximations for the remainder of the spatial terms in Eq. (3.32). For example, we can now represent the flux tensor as

$$\boldsymbol{F}_N = \boldsymbol{F}(\boldsymbol{q}_N) \tag{3.34}$$

and the source function as

$$\boldsymbol{S}_N = S(\boldsymbol{q}_N). \tag{3.35}$$

Upon defining these expansions, we can then substitute them into Eq. (3.32), multiply the equations by a test function, and integrate to obtain the element-wise integral

problem: find $\boldsymbol{q}_N \in \mathcal{S}(\Omega_e) \forall \psi \in \mathcal{S}(\Omega_e)$ on each element Ω_e such that

$$\int_{\Omega_e} \psi_i \left(\frac{\partial \boldsymbol{q}_N}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{F}_N \right) d\Omega_e = \int_{\Omega_e} \psi_i S_N \, d\Omega_e \tag{3.36}$$

where $\mathcal{S}$ is the finite-dimensional space

$$\mathcal{S} = \left\{ \psi \in \mathcal{L}^2(\Omega) : \psi|_{\Omega_e} \in P_N(\Omega_e) \forall \Omega_e \right\},$$

P_N is the polynomial space of order N defined on Ω_e and the union of these N_e elements defines the global domain, i.e., $\Omega = \bigcup_{e=1}^{N_e} \Omega_e$. Next, we integrate the divergence term by parts to get

$$\begin{aligned} \int_{\Omega_e} \psi_i \frac{\partial \boldsymbol{q}_N^{(e)}}{\partial t} \, d\Omega_e + \sum_{l=1}^{4} \int_{\Gamma_e} \psi_i \boldsymbol{n}^{(e,l)} \cdot \boldsymbol{F}_N^{(e)} \, d\Gamma_e \quad &- \quad \int_{\Omega_e} \boldsymbol{\nabla} \psi_i \cdot \boldsymbol{F}_N^{(e)} \, d\Omega_e \\ &= \quad \int_{\Omega_e} \psi_i S_N^{(e)} \, d\Omega_e \end{aligned} \tag{3.37}$$

where $\boldsymbol{n}^{(e,l)}$ is the outward normal vector going from element e to element l that defines a specific edge of the (in this specific case) quadrilateral control volume. Now, since the solutions are discontinuous across element boundaries then it becomes critical (in order to construct a consistent and stable numerical approximation to the governing continuous equations) to choose the flux tensor carefully. To resolve this inconsistency, a numerical flux is introduced that we denote by $F^{(*,l)}$. The simplest choice is the mean value between the two elements claiming the same interface

$$\boldsymbol{F}_N^{(*,l)} = \frac{1}{2} \left[\boldsymbol{F}_N^{(e)} + \boldsymbol{F}_N^{(l)} \right]$$

where the superscripts e and l represent the element under consideration and the side (interface) neighbor; unfortunately this numerical flux is not the best choice. Another easy but better choice is the local Lax-Friedrichs (or Rusanov) flux defined as

$$\boldsymbol{F}_N^{(*,l)} = \frac{1}{2} \left[\boldsymbol{F}_N^{(e)} + \boldsymbol{F}_N^{(l)} - \delta_{diss} |\lambda_{max}| \boldsymbol{n}^{(e,l)} \left(\boldsymbol{q}_N^{(l)} - \boldsymbol{q}_N^{(e)} \right) \right] \tag{3.38}$$

where λ_{max} is the maximum wave speed of the shallow water equations (the maximum eigenvalue of the Jacobian matrix at the edge l) and we have included the switch δ_{diss}

that controls whether the dissipation term is included. Alternative, more sophisticated formulations for the numerical flux have also reported in the literature [36, 121, 127], which, however, we will not use as they are beyond the scope of this work. With a specific numerical flux defined, the DG formulation becomes

$$\int_{\Omega_e} \psi_i \frac{\partial \boldsymbol{q}_N^{(e))}}{\partial t}\, d\Omega_e + \sum_{l=1}^{4} \int_{\Gamma_e} \psi_i \boldsymbol{n}^{(e,l)} \cdot \boldsymbol{F}_N^{(*,l)}\, d\Gamma_e - \int_{\Omega_e} \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{F}_N^{(e)}\, d\Omega_e = \int_{\Omega_e} \psi_i S_N^{(e)})\, d\Omega_e \quad (3.39)$$

that is in fact the *weak* form DGM. Integrating by parts one more time yields the following mathematically equivalent system

$$\int_{\Omega_e} \psi_i \frac{\partial \boldsymbol{q}_N^{(e)}}{\partial t}\, d\Omega_e + \sum_{l=1}^{4} \int_{\Gamma_e} \psi_i \boldsymbol{n}^{(e,l)} \cdot \left(\boldsymbol{F}_N^{(*,l)} - \boldsymbol{F}_N^{(e)}\right)\, d\Gamma_e + \int_{\Omega_e} \psi_i \boldsymbol{\nabla} \cdot \boldsymbol{F}_N^{(e)}\, d\Omega_e = \int_{\Omega_e} \psi_i S_N^{(e)}\, d\Omega_e \quad (3.40)$$

which is the *strong* form DGM and is the form that we shall use to compare and contrast with the SMPM described in section 3.2.1. Next, let us expand the terms $\boldsymbol{q}_N$ and $\boldsymbol{S}_N$ in order to rewrite Eq. (3.40) in matrix-vector form. Expanding these terms in Eq. (3.40) gives

$$M_{ij}^{(e)} \frac{dq_j^{(e)}}{dt} + \left(\boldsymbol{D}_{ij}^{(e)}\right)^{\mathcal{T}} \boldsymbol{F}_j^{(e)} + \sum_{l=1}^{4} \left(\boldsymbol{M}_{ij}^{(l)}\right)^{\mathcal{T}} \left(\boldsymbol{F}_j^{(*,l)} - \boldsymbol{F}_j^{(e)}\right) = M_{ij}^{(e)} S_j^{(e)} \quad (3.41)$$

where the elemental matrices are defined as follows:

$$M_{ij}^{(e)} = \int_{\Omega_e} \psi_i \psi_j\, d\Omega_e, \quad \boldsymbol{D}_{i,j}^{(e)} = \int_{\Omega_e} \psi_i \boldsymbol{\nabla}\psi_j\, d\Omega_e, \quad \boldsymbol{M}_{ij}^{(l)} = \int_{\Gamma_e} \psi_i \psi_j \boldsymbol{n}^{(e,l)} \quad (3.42)$$

where $^{\mathcal{T}}$ denotes the transpose operator. At this point in the DG formulation, we have to introduce numerical quadrature in order to evaluate the integrals defined in Eq.(3.42) in

the following way

$$
\begin{aligned}
M_{ij}^{(e)} &= \sum_{k=1}^{(Q+1)^2} \omega_k^{(e)} |J_k^{(e)}| \psi_i(\boldsymbol{x}_k) \psi_j(\boldsymbol{x}_k), \\
\boldsymbol{D}_{ij}^{(e)} &= \sum_{k=1}^{(Q+1)^2} \omega_k^{(e)} |J_k^{(e)}| \psi_i(\boldsymbol{x}_k) \boldsymbol{\nabla} \psi_j(\boldsymbol{x}_k), \\
\boldsymbol{M}_{ij}^{(l)} &= \sum_{k=1}^{(Q+1)} \omega_k^{(l)} |J_k^{(l)}| \psi_i(\boldsymbol{x}_k) \psi_j(\boldsymbol{x}_k)
\end{aligned} \tag{3.43}
$$

where Q is the number of quadrature points along each direction of the quadrilateral element, and ω and J are quadrature weights and Jacobians, respectively.

Using GLL points for both interpolation and integration we obtain the following element matrices

$$
\begin{aligned}
M_{ij}^{(e)} &= \omega_i^{(e)} |J_i^{(e)}| \delta_{ij}, \\
\boldsymbol{D}_{ij}^{(e)} &= \omega_i^{(e)} |J_i^{(e)}| \boldsymbol{\nabla} \psi_j(\boldsymbol{x}_i), \\
\boldsymbol{M}_{ij}^{(l)} &= \omega_i^{(l)} |J_i^{(l)}| \delta_{ij}
\end{aligned} \tag{3.44}
$$

where δ denotes the usual Kronecker delta function. Using Eq. (3.44) in Eq. (3.41) and dividing by the mass matrix yields:

$$
\frac{dq_i^{(e)}}{dt} + \left(\boldsymbol{\nabla} \psi_j(\boldsymbol{x}_i) \right)^{\mathcal{T}} \boldsymbol{F}_j^{(e)} = S_i^{(e)} + \sum_{l=1}^{4} \tau_i^{(l)} Q_i^{(l)} \boldsymbol{n}_i^{(e,l)} \cdot \left(\boldsymbol{F}_i^{(e)} - \boldsymbol{F}_i^{(*,l)} \right) \tag{3.45}
$$

where

$$
Q_i^{(l)} = \begin{cases} 1 & \text{if } i \text{ is on the edge } l \\ 0 & \text{otherwise} \end{cases}
$$

and

$$
\tau_i^{(l)} = \frac{\omega_i^{(l)} |J_i^{(l)}|}{\omega_i^{(e)} |J_i^{(e)}|};
$$

note that Eq. (3.45) is quite similar to Eq. (3.31) for the SMPM.

Next, we need to simplify the penalty-like term that we have called τ. To do so requires explicitly stating the value of the Jacobians of both the element and edges. For the sake of simplicity, if we assume that $\xi = \xi(x)$ and $\eta = \eta(y)$, that is, that the computational axes are aligned exactly with the physical axes, then we can write

$$\begin{aligned} \xi &= \frac{2(x - x_0)}{\Delta x} - 1 \\ \eta &= \frac{2(y - y_0)}{\Delta y} - 1 \end{aligned} \tag{3.46}$$

where x_0, y_0 is the left-bottom most point on each element and $\Delta x, \Delta y$ is the length of the element along the x and y directions, respectively.

This mapping yields the following metric terms

$$\begin{aligned} \frac{\partial \xi}{\partial x} &= \frac{2}{\Delta x} \\ \frac{\partial \eta}{\partial y} &= \frac{2}{\Delta y} \end{aligned} \tag{3.47}$$

with the following Jacobians

$$|J^{(e)}| \equiv \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi} = \frac{\Delta x \Delta y}{4}$$

and

$$|J^{(l)}| = \begin{cases} \frac{\Delta y}{2} & \text{along a vertical interface (Left-Right edge)} \\ \\ \frac{\Delta x}{2} & \text{along a horizontal interface (Top-Bottom edge).} \end{cases}$$

From the definition of these metric terms we can see that the penalty-like term simplifies to

$$\tau_i^{(l)} = \begin{cases} \frac{2}{\omega \Delta x} & \text{along a vertical interface (Left-Right edge)} \\ \\ \frac{2}{\omega \Delta y} & \text{along a horizontal interface (Top-Bottom edge)} \end{cases}$$

where $\omega = \omega_0 = \omega_N$ is the value of the quadrature weight at the beginning or end point (they are equal by symmetry). Introducing the DGM numerical flux given in Eq. (3.38)

into Eq. (3.45) yields

$$\frac{dq_i^{(e)}}{dt} + \left(\boldsymbol{\nabla}\psi_j(\boldsymbol{x}_i)\right)^{\mathcal{T}} \boldsymbol{F}_j^{(e)} = S_i^{(e)} + \sum_{l=1}^{4} \widehat{\tau}_i^{(l)} Q_i^{(l)} \boldsymbol{n}_i^{(e,l)} \cdot \left[\boldsymbol{F}_i^{(e)} - \boldsymbol{F}_i^{(l)} - \delta_{diss}|\lambda_{max}|\boldsymbol{n}_i^{(e,l)}\left(\boldsymbol{q}_i^{(l)} - \boldsymbol{q}_i^{(e)}\right)\right] \quad (3.48)$$

where

$$\widehat{\tau} \equiv \frac{\tau}{2} = \frac{1}{\omega \Delta s}$$

and $\Delta s = (\Delta x, \Delta y)$ depending in which direction the interface is oriented. At this point, we have not made too many sacrifices or simplifications in deriving Eq. (3.48). This equation is in fact a valid DGM representation of the shallow water equations with only the very slight assumptions that:

1. The computational coordinates (ξ, η) are aligned with the physical coordinates (x, y).
2. Co-located interpolation and integration points are used. The fact that we have chosen these points to be the GLL points results in inexact integration.
3. The numerical flux used is the simple Rusanov flux.

Taking the special case $\delta_{diss} = 0$, that is, no dissipation in the flux term, yields

$$\frac{dq_i^{(e)}}{dt} + \left(\boldsymbol{\nabla}\psi_j(\boldsymbol{x}_i)\right)^{\mathcal{T}} \boldsymbol{F}_j^{(e)} = S_i^{(e)} + \sum_{l=1}^{4} \widehat{\tau}_i^{(l)} Q_i^{(l)} \boldsymbol{n}_i^{(e,l)} \cdot \left[\boldsymbol{F}_i^{(e)} - \boldsymbol{F}_i^{(l)}\right] \quad (3.49)$$

which is identical to the SMPM representation given in Eq. (3.31). Eq. (3.49) shows that another way of viewing the penalty term is as an extra differencing term (as is evident by the $\frac{1}{\Delta s}$ term in $\widehat{\tau}$ and ΔF in the numerator) that considers the information from the neighboring elements, which is in fact what we mean by the usual term *flux*. In section 3.3 we use Eq. (3.48) with and without the dissipation term to compare the SMPM with the DGM. We now turn our discussion to the time-integrator we use to advance the SMPM and DGM solutions forward in time.

s	α_{ik}				β_{ik}			
	1				1/2			
4	0	1			0	1/2		
	2/3	0	1/3		0	0	1/6	
	0	0	0	1	0	0	0	1/2

Table I: Coefficients for the third order - four stage SSP-RK(34) method

3.2.3 Temporal Discretization

To retain the high-order accuracy of the SMPM and the DGM, a high-order time advancement scheme is needed. The explicit strongly stability preserving Runge-Kutta (SSP-RK) method [22, 113] is implemented for both approaches. Consider the following initial value problem

$$\frac{dq}{dt} = R(q). \tag{3.50}$$

The prediction at the time $n+1$ is based on the existing solution at the time n and the forcing terms $R(q)$. The scheme can be written as [113]

$$q^{(0)} = q^n \tag{3.51}$$

$$q^{(i)} = \sum_{k=0}^{i-1} \left(\alpha_{ik} q^{(k)} + \Delta t \beta_{ik} R(q^{(k)}) \right), \qquad i = 1, 2, \cdots, s \tag{3.52}$$

$$q^{(n+1)} = q^{(s)} \tag{3.53}$$

where s are the number of stages of the SSP-RK approach, α_{ik} and β_{ik} are constant coefficients given in Table I [113], and Δt is the size of the time step at a specific time.

3.3 Test cases: Description and Results

Six test cases are examined to compare the performance of the SMPM and DGM in terms of accuracy, dynamic stability, robustness and conservation properties: three linear

(standing wave, Kelvin wave, and Stommel problem), where accuracy can be evaluated through the availability of analytic solutions, and three non-linear (nonlinear Stommel, equatorial Rossby wave, and Riemann problem) that provide a platform for assessing the dynamic stability and robustness of the methods. In addition, results obtained with the spectral element method (SEM) [44, 45, 50] are included to compare, for each case, the behavior of a continuous method with a discontinuous element-based approach. For the linear cases an additional error analysis based on the normalized L_∞ and L_2 norms of the error is performed.

The normalized L_∞ and L_2 error norms are defined as

$$||h||_{L_\infty} = \frac{\max_{\mathbf{x}\in\Omega}(h_{exact} - h)}{\max_{\mathbf{x}\in\Omega} h_{exact}} \tag{3.54}$$

$$||h||_{L_2} = \sqrt{\frac{\int_\Omega (h_{exact} - h)^2 d\Omega}{\int_\Omega h_{exact}^2 d\Omega}}. \tag{3.55}$$

The mass (M) and energy (E) of the system are measured in the following way

$$M = \int_\Omega \phi d\Omega \tag{3.56}$$

$$E = \int_\Omega \left[\phi(u^2 + v^2) + \phi^2\right] d\Omega. \tag{3.57}$$

The metric for assessing mass and energy conservation is the respective relative error, defined with respect to the corresponding initial values of M and E. It is computed as

$$R_M = \left|\frac{M_t - M_0}{M_0}\right|, \qquad R_E = \left|\frac{E_t - E_0}{E_0}\right| \tag{3.58}$$

where R_M and R_E are the relative errors in mass and energy, and M_0, E_0, M_t, E_t are the corresponding values for mass and energy at the initial and final times of the simulation, respectively. For each test case, it is specified explicitly if mass and energy are lost or generated by the end of simulation.

For all simulations no boundary conditions are applied to the continuity equation. For the momentum equation no-flux (i.e., reflecting) boundary conditions are applied along all four walls of the basins; for the SEM and SMPM methods this is accomplished via strong homogeneous Dirichlet boundary conditions whereas for the DGM they are satisfied in a weak sense.

To compute the Courant number a high-order cell technique is used, where the cells are defined based on the GLL points on each subdomain. A mean velocity and geopotential height is defined at the center of each cell [51]. With these considerations, the Courant number is defined as

$$\text{Courant Number} = \max\left(\frac{\Delta t(U + \sqrt{\phi})}{\Delta s}\right)$$

where Δt is the size of the time step, U is the mean velocity magnitude at the cell, ϕ is the average geopotential height in the cell and $\Delta s = \sqrt{\Delta x^2 + \Delta y^2}$ is the grid spacing. For SMPM and DGM, the maximum Courant number Δt that ensures stability of the numerical simulations 0.5 (Courant Number ≤ 0.5). The equivalent value for SEM is 1. As specified in the relevant sections, two test cases (standing and Kelvin wave) are run with a significantly smaller time step to prevent the time-stepping error from dominating the error associated with the spatial discretization. Nonetheless, as the conservation properties of the SMPM are negatively impacted by a linearly growing loss of mass which is of order machine epsilon at each time step, all other test cases are run with a time step that is 80% the maximum time step associated with Courant number limits indicated above. We refer the reader to sections 3.4.1 and 3.4.2 for further discussion on time-stepping error and the impact of time step on the conservation properties of the spatial discretization methods under consideration. The degree of polynomial approximation is varied from $N = 4$ to 20. The number of subdomains is also varied within a range dependent

on the geometry of each case, and the SSP-RK34 method defined previously is used to advance in time the simulations.

3.3.1 Linear Problems

In this section, we compare the three methods quantitatively using linear test cases that have analytic solutions.

Linear Standing Wave

This case represents the evolution in time of a wave driven only by gravitational effects ($S = 0$) through an initial perturbation of the free surface. From references [51, 69], the analytic solution for this case is given by

$$h(x,y,t) = \cos(\pi x)\cos(\pi y)\cos(\pi t\sqrt{2}) \tag{3.59}$$

$$u(x,y,t) = \frac{1}{\sqrt{2}}\sin(\pi x)\sin(\pi y)\sin(\pi t\sqrt{2}) \tag{3.60}$$

$$v(x,y,t) = \frac{1}{\sqrt{2}}\cos(\pi x)\sin(\pi y)\sin(\pi t\sqrt{2}) \tag{3.61}$$

with $(x,y) \in [0,1]\times[0,1]$.

The simulations are run for $t \in [0, 0.5]$. Figure 3.3 shows results for SMPM, DGM and SEM simulations for a fixed number of subdomains and variable order of polynomial approximation N. A time step which is $1/50$th of that associated with a Courant number value of 0.4 is used, to make time-stepping errors sufficiently small. The results are indistinguishable if an even smaller time step is employed. Exponential convergence of the error norms for free surface elevation and horizontal velocity is attained for each method for polynomial degree less or equal than $N = 8$. At higher values of

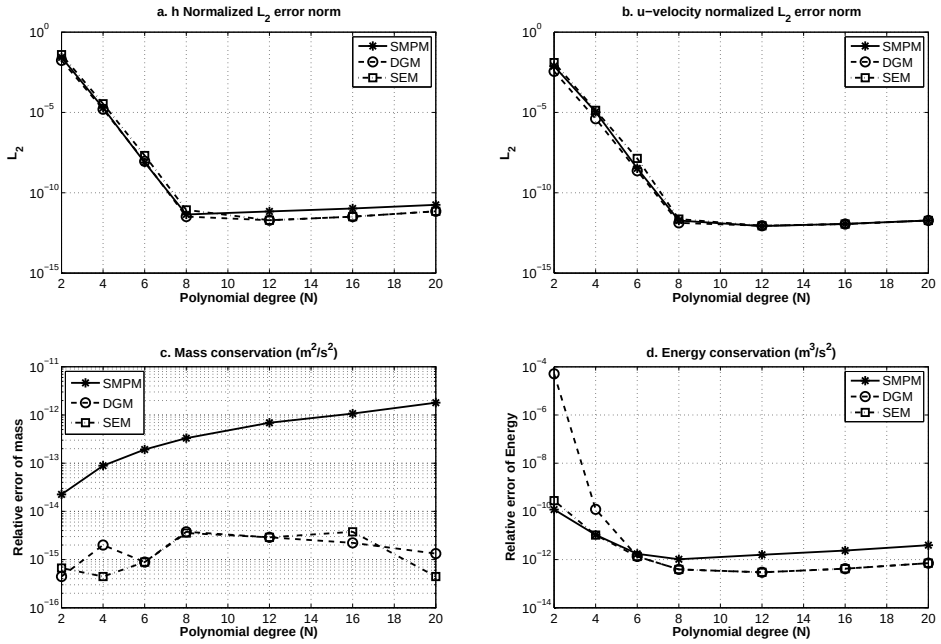


Figure 3.3: Analysis of the standing wave (5×5 subdomains) at $t = 0.5$ seconds for a varying number of GLL points. a) L_2 normalized relative error in the free surface elevation h. b) L_2 normalized relative error in u velocity. c) Relative error in mass. d) Relative error in energy.

N, the convergence rate is finally reduced, reaching a plateau of the order of $O(10^{-12})$. The Galerkin based methods (i.e. DGM, SEM) conserve mass up to machine precision. The SMPM mass cumulatively loses mass over time. All three methods show improved energy conservation with increasing N with the relative error reaching a value of $O(10^{-12})$ at $N = 8$. An interpretation for the performance of the SMPM in terms of mass conservation is offered in section 3.4.1.

Linear Kelvin Wave

The equatorial Kelvin wave is a low amplitude non-dispersive wave trapped in the vicinity of the equator. It is driven by rotational and gravitational effects through an initial

perturbation of the free surface. The analytic solution for this case [51, 35] is

$$h(x,y,t) = 1 + \exp\left(-\frac{y^2}{2}\right)\exp\left(-\frac{(x+5-t)^2}{2}\right) \quad (3.62)$$

$$u(x,y,t) = \exp\left(-\frac{y^2}{2}\right)\exp\left(-\frac{(x+5-t)^2}{2}\right) \quad (3.63)$$

$$v(x,y,t) = 0 \quad (3.64)$$

for $f_0 = 0, \beta = 1$ and $(x,y) \in [-20,20] \times [-10,10]$.

Simulations are run for $t \in [0,5]$. Figure 3.4 shows results for this case for a domain discretized with 20×10 elements and a varying value of N. As with the standing wave, here the time step is $1/50$th that associated with a Courant number value of 0.4. No further reduction in time step was required to make time-stepping errors sufficiently small. The behavior of the error norms is similar to that observed for the linear standing wave: exponential convergence is observed for all the three methods. DGM and SEM conserve mass up to machine precision. On the contrary, SMPM again shows a loss of mass, which, in the end of simulations, is up to one order of magnitude larger than the value computed for DGM and SEM. The trend in relative error of total energy conserved is comparable to that observed for the linear standing wave in Fig. 3.3. Improved energy conservation occurs with increasing N with a relative error value of $O(10^{-13})$ observed for $N = 20$.

Linear Stommel Problem

This problem [114] also known as westward intensification of wind-driven ocean currents, represents the steady balance between rotation, gravity, friction and wind stress in a square ocean basin. A sinusoidal wind stress forces an unperturbed free surface generating a small amplitude wave moving westward due to the Coriolis force that is

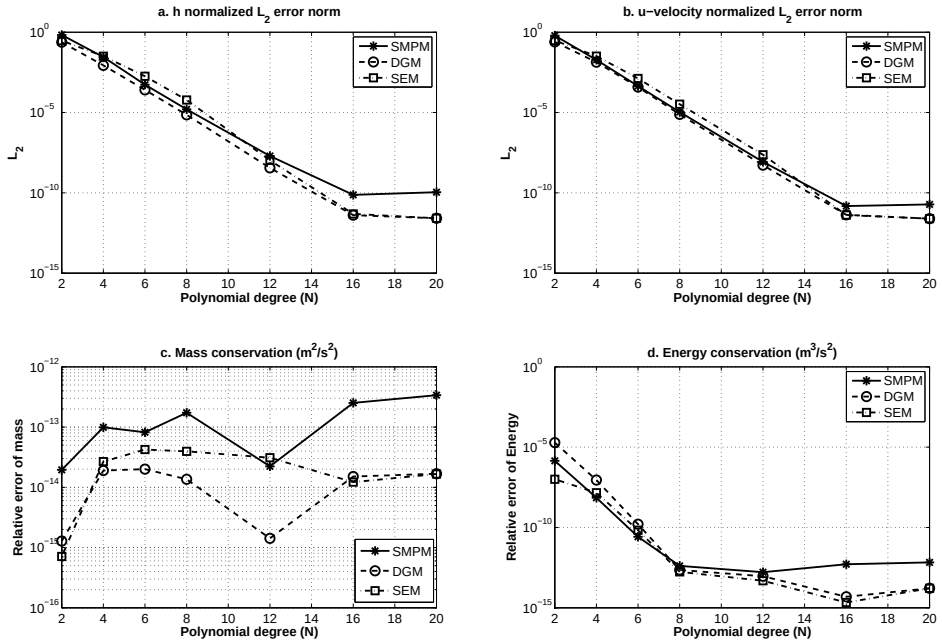


Figure 3.4: Kelvin wave results for 20×10 subdomains at $t = 5$. Panels (a) through (d) show the same quantities with Fig. 3.3.

compensated by bottom friction and gravitational effects and, eventually, reaches steady state. The analytic solution used for this case is [51]

$$\begin{aligned} h(x,y,t) &= \left(\frac{C_1}{\lambda_1}e^{\lambda_1 x} + \frac{C_2}{\lambda_2}e^{\lambda_2 x}\right)\frac{\gamma\pi}{l}\cos\left(\frac{\pi y}{l}\right) \\ &\quad + \frac{\tau\beta}{\gamma}\left(\frac{l}{\pi}\right)^2\cos\left(\frac{\pi y}{l}\right) \\ &\quad + f\sin\left(\frac{\pi y}{l}\right)(C_1 e^{\lambda_1 x} + C_2 e^{\lambda_2 x} + C_3) \qquad (3.65) \\ u(x,y,t) &= -(C_1 e^{\lambda_1 x} + C_2 e^{\lambda_2 x} + C_3)\frac{\pi}{l}\cos\left(\frac{\pi y}{l}\right) \qquad (3.66) \\ v(x,y,t) &= (C_1\lambda_1 e^{\lambda_1 x} + C_2\lambda_2 e^{\lambda_2 x})\sin\left(\frac{\pi y}{l}\right) \qquad (3.67) \end{aligned}$$

where

$$C_1 = C_3 \frac{1 - e^{\lambda_2 l}}{e^{\lambda_2 l} - e^{\lambda_1 l}} \tag{3.68}$$

$$C_2 = -C_3 \frac{1 - e^{\lambda_1 l}}{e^{\lambda_2 l} - e^{\lambda_1 l}} \tag{3.69}$$

$$C_3 = \frac{\tau l}{\pi \gamma} \tag{3.70}$$

For the case presented here, $f_0 = 1 \times 10^{-4}, \beta = 1 \times 10^{-11}, \gamma = 1 \times 10^{-6}, g = 10, \rho = 1000$, $\tau = 0.2$, $H_0 = 1000$, and $(x, y) \in [0, 1 \times 10^6] \times [0, 1 \times 10^6]$. Note that the solution is symmetric with respect to the y axis.

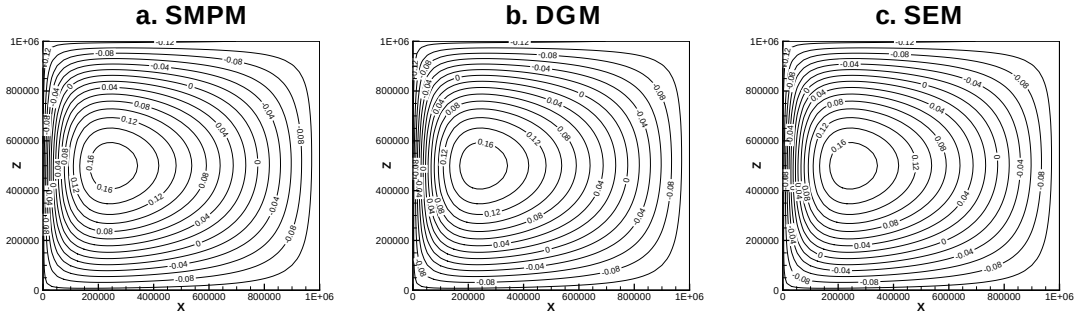


Figure 3.5: Free surface elevation computed by all three methods for the linear Stommel problem for 5×5 subdomains and $N = 12$ at $t = 400$ days

Simulations are run until the solution is close to the steady state (i.e. $t = 320$ days), and the structure of the steady state flow field, displaying the expected symmetry around the horizontal axis at $z = 5 \times 10^5$, is shown in Fig. 3.5 for all three methods. Figure 3.6 shows the error norm convergence curves for the case of a 5×5 mesh for solutions obtained with different values of N. For all three methods, the error in the free surface displacement shows an exponential convergence similar to the previous two linear cases for up to $N = 8$, beyond which the error norms level off to a constant value. This plateau is reached because an exact steady is almost never attained in practice, as simulations are dominated by slowly-decaying, weak-amplitude basin-scale modes, with the decay time

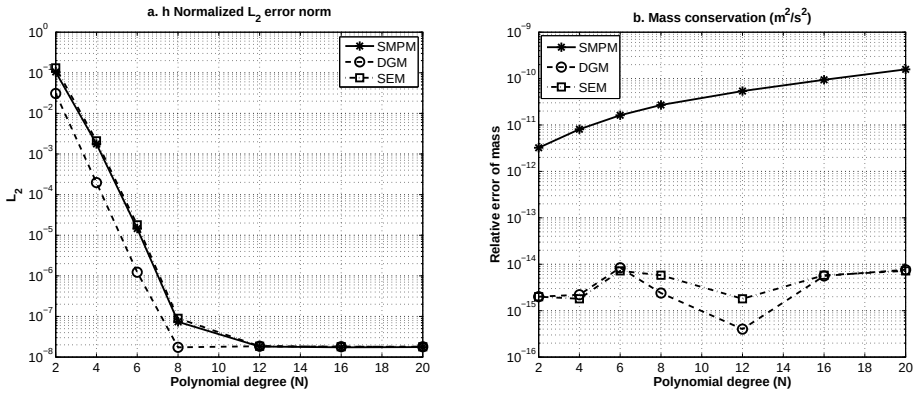


Figure 3.6: Linear Stommel problem results for 5×5 subdomains at $t = 320$ days. a) Convergence plot for the L_2 normalized relative error in the free surface elevation h. b) Relative error in mass conservation.

of the gravest, longest-wavelength, mode reaching 60 years [61]. Mass is conserved up to machine precision by DGM and SEM, whereas SMPM shows a loss of total mass up to three orders of magnitude larger than DGM and SEM.

3.3.2 Nonlinear Problems

In this section, we compare the three methods qualitatively using nonlinear test cases that, unfortunately, do not have analytic solutions. Instead, we use the conservation of mass and energy to compare the methods. All three models formally should conserve mass but are not guaranteed to conserve energy. It is possible to conserve energy (at least up to the time-truncation error) but this requires slight modifications to the discrete operators that we will not pursue in this work.

Nonlinear Rossby Soliton

This case considers an equatorial non-linear Rossby wave of weak amplitude, driven by gravity and rotational forces. It is initialized by a Gaussian-like perturbation in the free surface elevation. An approximate asymptotic solution of the system of Korteweg-DeVries equations resulting from the SWE through application of the method of multiple scales is obtained for this problem in [10]. Although this first order solution does not provide a reference to assess the convergence rate of the numerically computed solution for the SEM, DGM, and SMPM, it is used to compare associated phase speed and solution structure with the corresponding estimates computed by the three numerical methods. For this case $(x, y) \in [-24, 24] \times [-8, 8]$, $g = 1$, and the Coriolis force $f(y) = y$.

Simulations are run for $t \in [0, 40]$. All three methods accurately reproduce the free surface/velocity structure of the soliton and its propagation at a constant phase speed equal to the analytically predicted value. The structure of the free surface elevation field at the end of the simulation, with its characteristic two-lobe structure, as computed by all three methods is shown in Fig. 3.7. Figure 3.8 shows results for mass and energy conservation for 24×8 subdomains, and varying N, which are similar to their counterparts obtained for the linear cases. The SMPM is subject to a decrease in mass when the polynomial order increases. The DGM conserves mass up to machine precision, with the SEM offering comparable performance. The SEM and SMPM are the most and least energy conserving, respectively. As discussed in section 3.4.1, the energy conservation properties of the DGM are highly dependent on the formulation of the numerical flux and the use of spectral filtering (see Fig. 3.12).

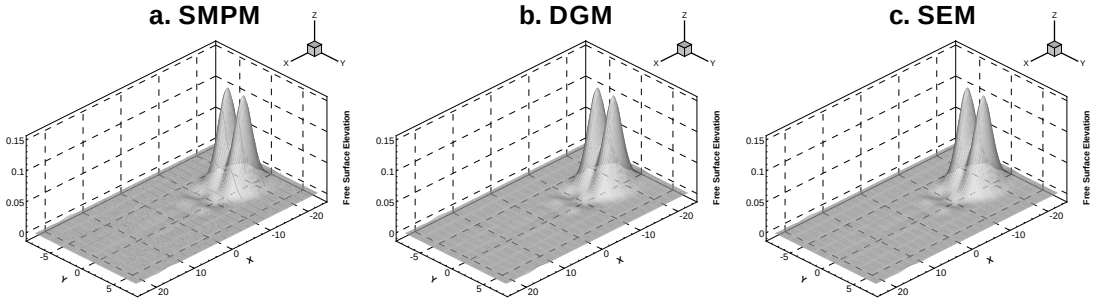


Figure 3.7: Qualitative comparison of the Non-linear Rossby wave results with 24×8 subdomains, $N = 12$, and at time $t = 40$.

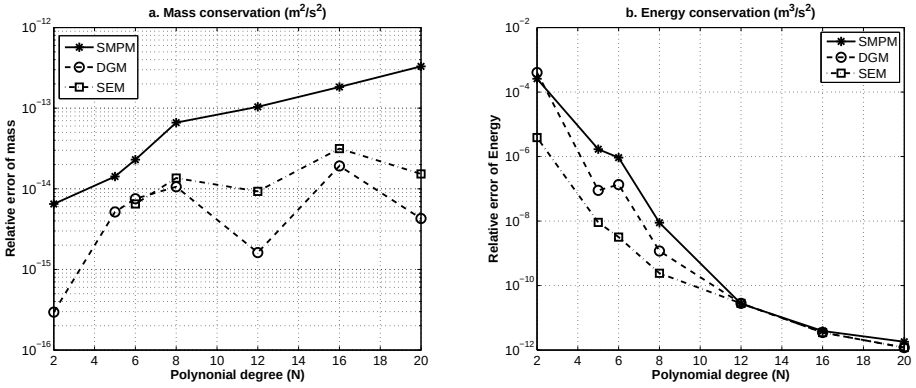


Figure 3.8: Non-linear Rossby wave results for 24×8 subdomains at $t = 40$. a) Relative error in mass. b) Relative error in energy.

Nonlinear Stommel Problem

The same configuration (forcing parameters, dimensions of the physical domain, and boundary conditions) is used as in the linear Stommel problem. However, the fully nonlinear set of Eqs. (3.4) are now solved. In this case, a shift of the gyre toward the northwest part of the basin is expected due to the effect of the nonlinear terms.

Figure 3.9 shows the steady state results, for a domain with 5×5 subdomains. Similar trends are observed for all three methods. Note that in this particular case, the differences in subdomain interface treatment between SMPM and DGM give rise to challenges of numerical stability for the former, when values of polynomial degree $N \geq 12$

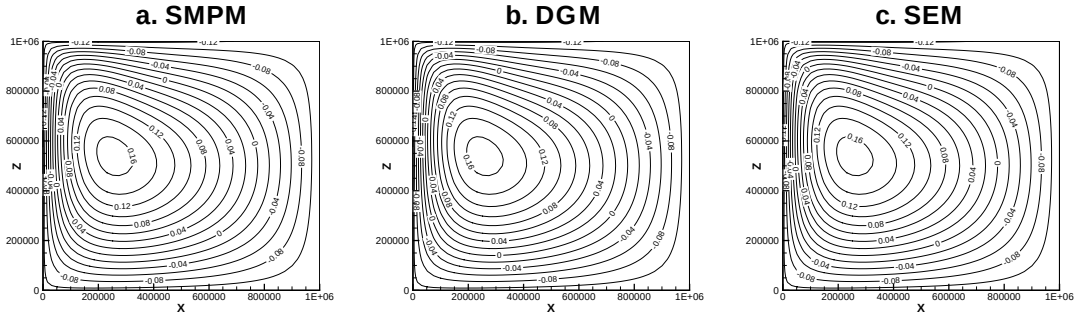


Figure 3.9: Nonlinear Stommel problem for 5×5 subdomains and $N = 12$ at $t = 400$ days

are used. In the SMPM, when 5×5 subdomains are used and $N \geq 12$, weak spurious oscillations develop in the top left corner of the domain and intensify, as time advances, eventually forcing a catastrophic blow-up of the solution. As a counter-measure, a 16-th order Boyd-Vandeven filter [87] is used, which attenuates only the very highest modes of the solution, to suppress these oscillations. This problem does not occur for the DGM, as the spurious oscillations are damped by the dissipative term $\delta_{diss} = 1$ in the numerical flux. The sensitivity of the DGM and SMPM to the presence of dissipative terms is examined in greater detail in section 3.4.3. Figure 3.10 shows the behavior of the relative error in mass as a function of N, which is similar to what is observed for the corresponding linear problem (Fig. 3.6) . Results are restricted to $N \leq 8$, as high-order polynomial approximations require the use of a spectral filter to preserve stability.

Nonlinear Riemann Problem

This modification of the circular dam break problem [92] is considered as a platform to assess the performance of the three methods in simulating strongly nonlinear flows, i.e. flow fields with distinct sharp spatial gradients. The initial condition, a Gaussian bump

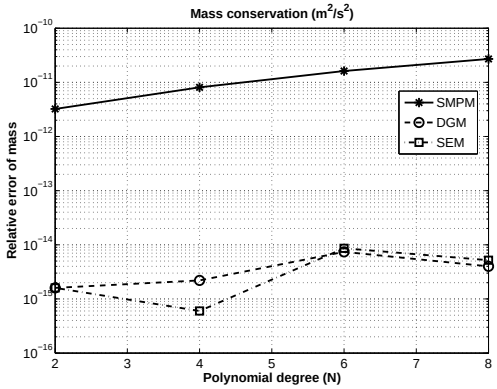


Figure 3.10: Relative error in mass as a function of polynomial order for the Non-linear Stommel problem. 5×5 subdomains at $t = 360$ days.

(used instead of a cylindrical step function), is characterized by such a sharp gradient and has free surface and velocity fields given by:

$$h(x, y, t_0) = H + A \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}\right) \tag{3.71}$$

$$u(x, y, t_0) = 0$$

$$v(x, y, t_0) = 0 \tag{3.72}$$

where $(x, y) \in [0, 1] \times [0, 1]$, $g = 9.8$, $H = 1$, $A = 0.2$, $x_0 = y_0 = 0.5$, and $\sigma = 0.05$. The flow is driven by gravity as in the standing wave problem. Simulations are run for $t \in [0, 0.2]$, i.e., up to a short time after the first reflection of the initial wave from the domain boundaries where reflecting boundary conditions are applied.

Figure 3.11 shows results for conservation properties in the case of a 5×5 subdomains. In terms of mass conservation, it is difficult to discern which method offers superior performance. The energy conservation properties of each method improve with increasing N. At a given value of N, the DGM is found to produce a slightly larger relative error in terms of the total final energy. Note that for the time for which the simulations were run, no filtering was needed to preserve numerical stability at all values of N and subdomain

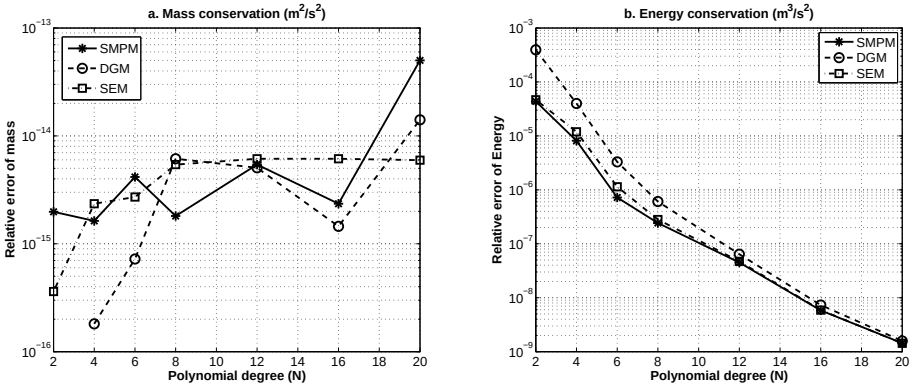


Figure 3.11: Nonlinear Riemann problem for 5×5 subdomains at $t = 0.2$. Panels (a) and (b) are the same as Fig. 3.8

.

thicknesses considered. Nevertheless, the smoothness of the solution is damaged at later times, as weak spurious wiggles emerge. As in the case of the non-linear Stommel problem, in the DGM, the dissipation term in the Rusanov flux stabilizes the solution while keeping it free of spurious oscillation, although somewhat adversely impacting the energy conservation properties of the method. The role of spectral filtering and dissipative terms on the conservation properties for the DGM is further discussed in section 3.4.1.

3.4 Discussion

3.4.1 Mass and Energy Conservation

All three methods are found to have very good conservation properties, a direct result of their formulation, see e.g. [28] for SMPM, [48] for DGM, and [116] for SEM. The DGM conserves mass up to machine precision. The SMPM is found to lose mass over long model times with the corresponding relative error as much as four orders of mag-

nitude larger than that for the DGM. This error increases with number of time steps. Such observations might initially seem perplexing, given the analytical demonstration of Don et al. [28] that the averaging method-based penalty scheme is conservative. For all SMPM-driven test-cases we have found that the mass loss (not shown here) is a linear function of time, with a decay rate that is of the order of machine epsilon. The linear Stommel problem has a total mass loss that reaches values of 10^{-10} at higher N, a value even higher than that observed for the standing and Kelvin wave test-cases where $1/50$th the maximum time step is used. This difference is simply because 10^6 time steps are required for the linear Stommel problem to reach steady-state. Consequently, we attribute the observed loss of mass to an accumulation of round-off error.

The energy conservation properties of all three methods improve with increasing N, although both SMPM and DGM are found to be inferior in this regard to the SEM. Note that in simulations where no energy sink terms (such as bottom friction in the Stommel problems) are present, the performance of the discontinuous techniques in terms of energy conservation can be strongly influenced by spectral filtering and the structure of the numerical flux terms, such as the dissipative term used within the Rusanov flux. Figure 3.12 shows the differences in conservation of mass and energy in the DGM, for the Riemann problem, when spectral filtering, through a 10th-order Boyd-Vandeven filter [87], is added to the simulation or the dissipation term is neglected in the numerical flux. The absence of both the dissipative term in the numerical flux and spectral filtering provides for the best energy conservation properties, although such behavior does not necessarily guarantee a smooth and stable solution for such a strongly nonlinear problem.

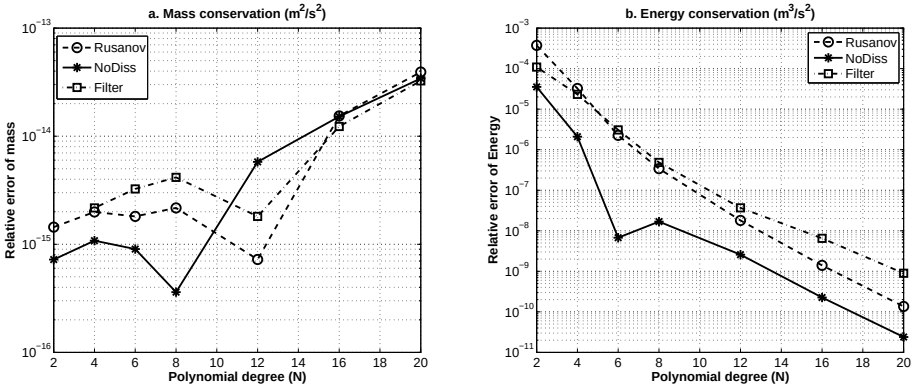


Figure 3.12: Comparison of conservation properties of the DGM for the Riemann problem. Results for 5×5 at $t = 0.2$. (a) Mass conservation. (b) Energy conservation.

3.4.2 Effect of time step on convergence and conservation properties

For the purpose of demonstrating that the temporal discretization error does not dominate over the spatial error, we now perform an analysis of the effect of time step, Δt, size, on the convergence and conservation properties of each of the three methods. The base time step corresponds to that associated with a simulation with Courant Number of 0.4. Δt is then progressively decreased by a factor of 2, 10 and 50 (denoted by $D2, D10, D50$ respectively). In Figure 3.13 the convergence plots for the free surface elevation h of the standing wave test case are presented for all three methods. For a given N, the increase in accuracy of all three methods is visible as Δt is decreased. Once a factor of 50 reduction is reached exponential convergence is obtained until $N = 8$.

The same exercise has been performed to assess the role of time-step on mass and energy conservation in all three methods. The results show (see Figure 3.14 and Figure 3.15) that the SMPM mass loss increases with decreasing Δt. This observation is consistent

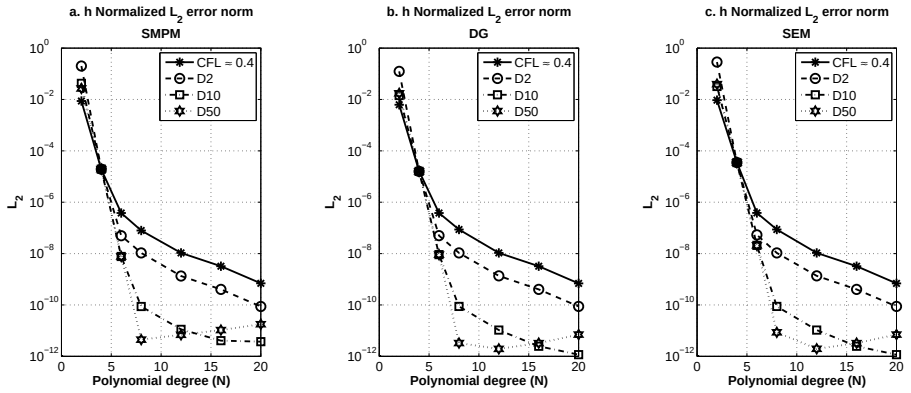


Figure 3.13: Convergence plots for the standing wave problem when different Δt sizes are used.

with the loss, at a linear decay rate of order machine epsilon, in the SMPM discussed in section 3.4.1. In contrast, the DGM and SEM conserve mass to the order of machine epsilon regardless of the value of Δt. On the other hand, conservation of energy is improved by the three methods once the polynomial degree increases or the size of Δt decreases.

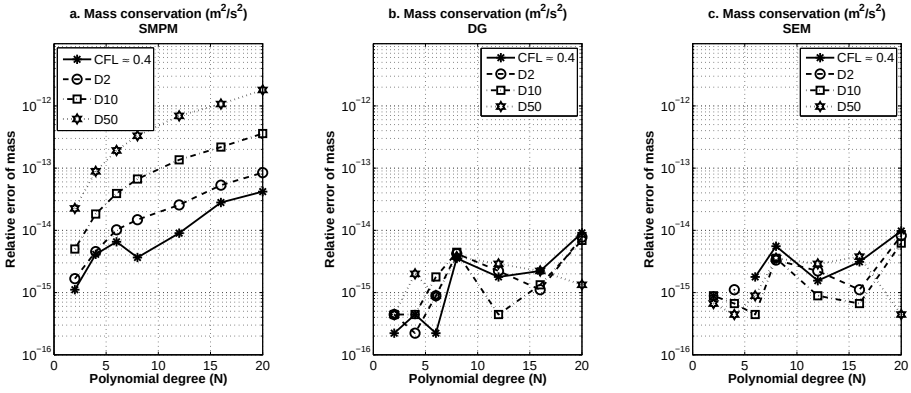


Figure 3.14: Conservation of mass for the standing wave problem when different Δt sizes are used.

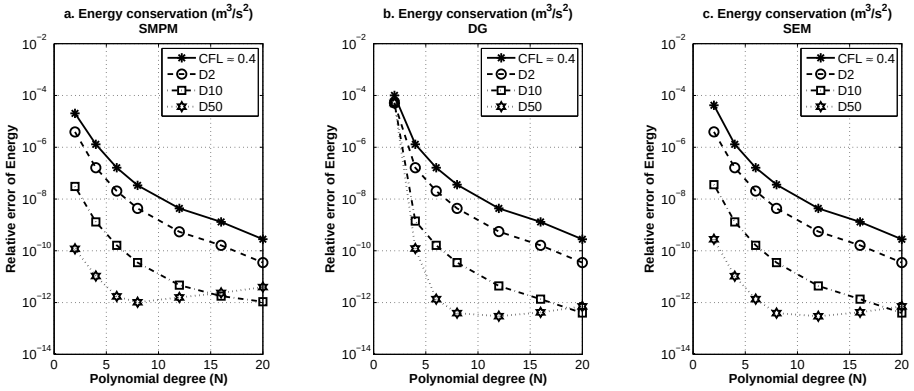


Figure 3.15: Conservation of energy for the standing wave problem when different Δt sizes are used.

3.4.3 Effect of Filtering

In the interfacial treatment of the SMPM, there is no dissipative term that removes spurious high wavenumber oscillations that develop in highly nonlinear simulations. Thus, spectral filtering is needed when such simulations are run for long integration times, namely when sharp localized features emerge in the simulations (e.g., nonlinear Riemann problem) or even when the structure of the solution is apparently smooth and free of any localized features (e.g., nonlinear Stommel problem). In contrast, in the case of the DGM , the dissipation term introduces a dissipation mechanism that stabilizes the solution and renders it oscillation-free; for a very simple flow problem, this term reduces to a simple upwinding scheme. By neglecting it, the DGM-generated solution also becomes unstable. Without resorting to recasting the nonlinear terms in skew-symmetric form [12] and in the absence of an over-integration-based de-aliasing strategy [78] (both which are out of the scope of this work), spectral filtering is required to recover stability. In terms of mass and energy conservation, the performance of the DGM appears to be very similar when spectral filtering and no dissipative term is used or when only the dissipative term is used (Fig. 3.12).

The performance of both SMPM and DGM is further examined in problems where significantly sharp features are present. The dam-break problem [120] is simulated with a cylindrical step-function of the free surface elevation as an initial condition and with $(x, y) \in [-20, 20] \times [-20, 20]$ and $t \in [0, 0.1]$. The effect of filtering (with a Boyd-Vandeven filter of $p = 10$) and the dissipative term on the solution are shown in Fig. 3.16.

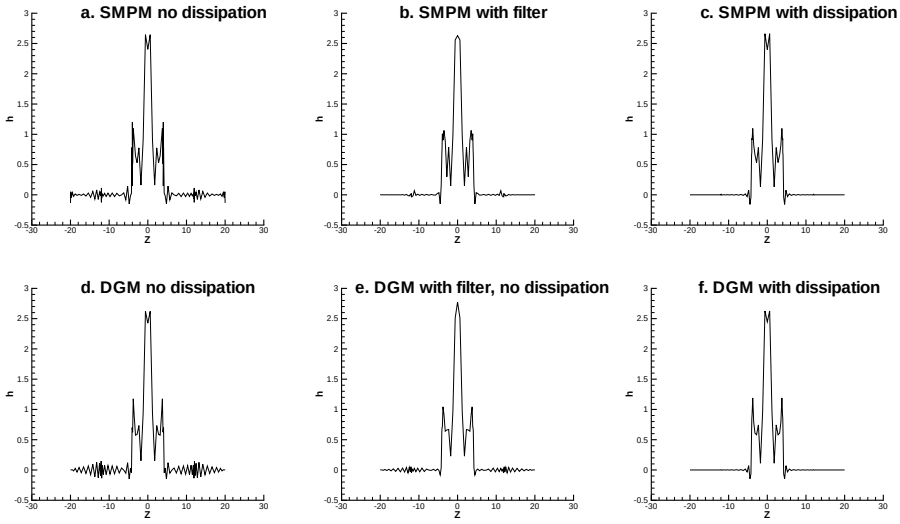


Figure 3.16: Cross section of the Dam-break problem for 5×5 subdomains, and $N = 20$ at $t = 0.1$. (a) SMPM without dissipation. (b) SMPM with filtering (Filter order $p = 10$). (c) SMPM with dissipative term. (d) DGM without dissipative term ($|\lambda|(\mathbf{q}^{\mathbf{R}} - \mathbf{q}^{\mathbf{L}})$. (e) DGM with filtering (Filter order $p = 10$), and without dissipation term. (f) DGM full Rusanov flux.

In the absence of a dissipative term in the DGM and any spectral filtering for both methods (panels a and d), spurious oscillations are localized in the vicinity of subdomain interfaces for the SMPM, whereas, in the DGM, these oscillations are more evenly distributed throughout the computational domain. When spectral filtering is applied to both methods (panels b and e), the oscillations are strongly damped in the subdomain interior where the effect of the filter is focused [53]. Nevertheless, some weaker oscillations remain at the subdomain interfaces [53]. If no spectral filtering is applied but

an additional dissipative term is added to the penalty term in the SMPM (panel c), the solution has a nearly identical structure with the one computed by the DGM with the full Rusanov flux. For the purpose of comparison, Fig. 3.17 shows the filtered solution obtained from the SEM which is contrasted to its filtered counterparts (no Rusanov flux term present) computed from DGM and SMPM (Figs. 3.16b and e). The results for SEM with filtering show stronger spurious oscillations than SMPM or DGM with dissipation or spectral filtering.

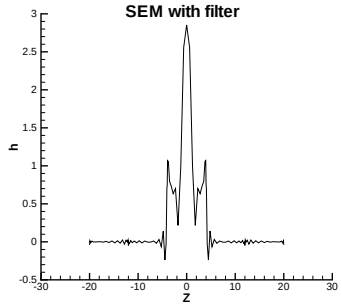


Figure 3.17: Cross section of the filtered Dam-break problem for 5×5 subdomains, and $N = 20$ at $t = 0.1$ (Filter order $p = 10$) for SEM.

3.4.4 Computational Efficiency and implementation

For all test cases, the order of magnitude of the CPU time per time step has been found to be comparable for both DGM and SMPM and increases when the number of degrees of freedom increases due to h or p refinement. Figure 3.18a shows the computational time for all three methods considered in this manuscript (SMPM, DGM and SEM) for different values of N for the Riemann problem with 5×5 subdomains and the same time step value for each method, corresponding to Courant Number = 0.4.

Figure 3.18b shows the time needed to advance a simulation to the same final time

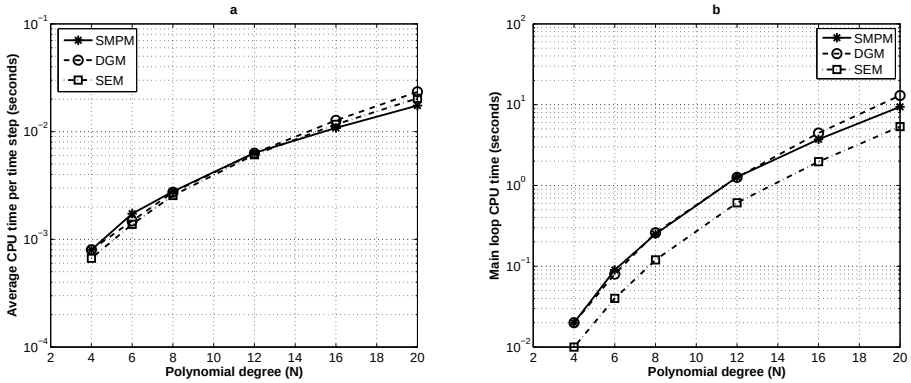
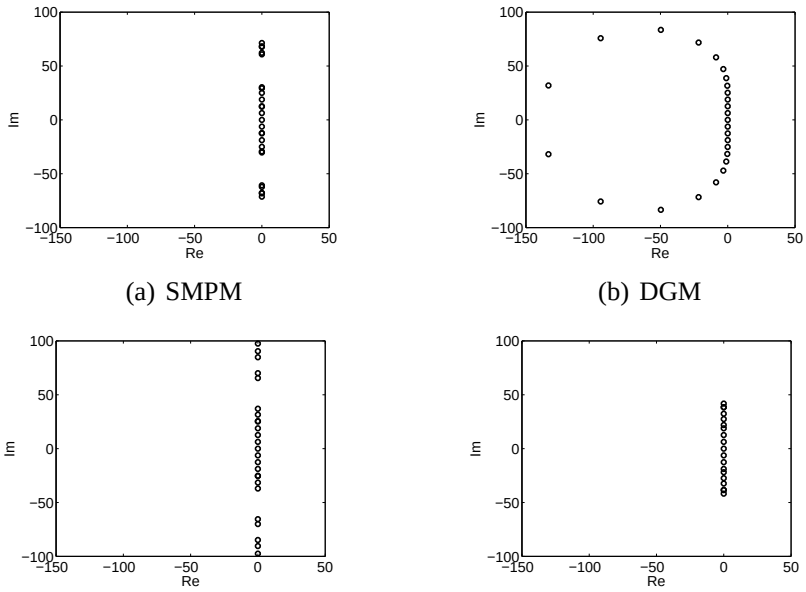


Figure 3.18: CPU time for the Riemann problem. 5×5 subdomains with different polynomial orders at $t = 0.2$. (a) All methods with Courant Number $= 0.4$ and (b) DGM and SMPM with Courant Number = 0.4 and SEM with Courant Number = 0.8.

as Fig. 3.18a, where the Courant Number is set to the empirically computed maximum value that enables a stable simulation for each method. SEM simulations are found to support double the maximum Courant Number value of DGM and SMPM and are thus twice as fast. DGM and SMPM simulations were also performed with a Courant Number value slightly above the empirically obtained stable limit value. In this case, DGM was found to destabilize faster than SMPM.

Theoretical justification for these observations is gained by examining the eigenvalue spectra of the discretized 1-D linear advection operator for each of the three discretization methods for a periodic domain with 5 subdomains and $N = 4$ (Fig. 3.19). In the absence of the dissipation term in DGM, and as expected, all three methods have purely imaginary eigenvalues. The extreme eigenvalues of DGM are roughly 25 % larger than their SMPM counterparts and double the corresponding SEM eigenvalues. Incorporation of the numerical flux term in DGM gives rise to eigenvalues with a negative real part which equip the numerical solution with the necessary numerical dissipation. Moreover, the separation between the eigenvalues with the largest absolute imaginary

values is reduced with respect to the case without dissipation but is still slightly larger than that in SMPM and almost double that of SEM. Taking into account the stability region of the SSP-RK34 scheme (which is stable along the imaginary axis) for Courant numbers below this eigenvalue separation can explain why SEM can attain double the Courant Number of DGM and SMPM and why DGM explodes a little faster than SMPM for a marginally unstable time step.



(a) SMPM (b) DGM

(c) DGM no dissipation (d) SEM

Figure 3.19: Eigenvalue distribution of the 1D discrete linear advection operator $(u\frac{d}{dx})$ for all three methods, with an advective velocity of $u = 1$. In all cases $x \in [-1, 1]$, 5 subdomains, $N = 4$.

In terms of implementation, in the context of the SWE, both the SMPM and DGM can be written as a system of time-dependent ordinary differential equations where the vector of unknowns is the solution vector at the grid points [84]. In the matrix-vector product that appears on the right hand side of this system of equations, the associated matrix is simply a spectral differentiation matrix (Eq. (2.2)) for the SMPM due to its underlying collocation method framework with any modifications to this matrix incurred through communication with points on the edge of the neighboring subdomain. Sim-

ilar modifications on account of the numerical flux term enter the construction of the corresponding right hand side matrix for the DGM, the core of which is built through additional numerical integration and, therefore, cost. This cost is, nevertheless, offset over the course of a long unsteady simulation. In summary, for hyperbolic systems of equations, the cost of SMPM and DGM are very similar. However, we expect the SMPM to have an advantage when elliptic operators are introduced since the addition of a Laplacian for the SMPM becomes simply a matter of introducing a Laplacian differentiation matrix whereas in DGM either local discontinuous Galerkin or interior penalty methods have to be introduced [18, 5, 21]. For SEM, the addition of Laplacian operators introduces only a slight cost.

CHAPTER 4

INCOMPRESSIBLE NAVIER STOKES SOLVER [1]

4.1 Incompressible Stratified flow model

4.1.1 Governing equations and Boundary Conditions

This study considers incompressible stratified flow governed by the Navier-Stokes equations under the Boussinesq approximation [85, 126]:

$$\frac{\partial \mathbf{u}}{\partial t} = \underbrace{-\frac{1}{2}[\mathbf{u}\cdot\nabla\mathbf{u} + \nabla(\mathbf{u}\cdot\mathbf{u})] + \mathbf{F}_g}_{\mathbf{N(u)}} - \frac{1}{\rho_0}\nabla p' + \nu \underbrace{\nabla^2\mathbf{u}}_{\mathbf{L(u)}} \quad , \tag{4.1}$$

$$\frac{\partial \rho'}{\partial t} = -\nabla\cdot(\mathbf{u}(\rho' + \overline{\rho}(z))) + \kappa\nabla^2\rho' \quad , \tag{4.2}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{4.3}$$

$$\text{where} \qquad \mathbf{F}_g = -g\frac{\rho'}{\rho_0}\hat{\mathbf{k}} \ . \tag{4.4}$$

The five unknowns to solve for are the velocity vector $\mathbf{u} = (u, v, w)$, and the pressure and density perturbations p' and ρ', respectively. The non-linear term in the momentum equations (4.1) is written in the skew-symmetric form to minimize aliasing effects in the numerical solution [12, 24, 128]. The perturbations p' and ρ' originate from the

[1] The contents of this chapter are published on the article [32] *A spectral quadrilateral multidomain penalty method solver for high-Reynolds number stratified incompressible flows* written by Jorge Escobar-Vargas, Peter Diamessis, and Takahiro Sakai

decomposition of the corresponding total values into [85]:

$$p = \overline{p}(x, y, z) + p'(x, y, z, t) \ , \tag{4.5}$$

$$\rho = \rho_0 + \overline{\rho}(z) + \rho'(x, y, z, t) \ . \tag{4.6}$$

Following the Boussinesq approximation, the reference pressure, $\overline{p}(x, y, z)$ and density, $\rho_0 + \overline{\rho}(z)$ are in hydrostatic balance:

$$\frac{\partial \overline{p}}{\partial z} = -(\rho_0 + \overline{\rho})g \ . \tag{4.7}$$

We seek to simulate a broader range of small-scale environmental flow processes than those attainable by the solver previously developed by the second author [25]. To this end, the computational domain is designed to flexibly enable non-periodic boundary conditions along all boundaries in the stream vs. depth plane. Possible choices of boundary conditions include homogeneous and non-homogeneous Dirichlet, Neumann, a mixed (Robin) conditions which are applied to both momentum and advection-diffusion equations. The boundary conditions for the pressure are of purely numerical nature and their discussion is thus deferred to 4.2.1.

In this work, we will only consider two-dimensional simulations within the framework of a quadrilateral Legendre multidomain discretization. Throughout the chapter, z will be used to denote the vertical direction, according to the standard convention in environmental fluid mechanics. Three-dimensional simulations are readily accessible by incoprorating a third, periodic spanwise direction subject to the Fourier discretization discussed in reference [25] and will be the subject of future, physics-focused publications.

4.2 Numerical Method

4.2.1 Temporal discretization

For the temporal discretization a high-order time splitting scheme proposed by Karniadakis et al. [74], and refereed as the KIO scheme, is used. This scheme is a high-temporal-accuracy variant of the projection techniques introduced by Chorin and Temam [20]. A more general and detailed analysis of projection methods for incompressible flows is presented in [60] and specifically for high-order methods in reference [15]. The scheme used in this work is the same with the one used by the earlier singly non-periodic spectral multidomain incompressible Navier-Stokes solver developed by the second author [25]. According to this scheme, if one integrates Eqs. (4.1)-(4.3) in time from level t_n to t_{n+1} one obtains the following semi-discrete equations decomposed into three fractional steps for $\mathbf{u}$, by means of the intermediate velocities $\hat{\mathbf{u}}$ and $\hat{\hat{\mathbf{u}}}$:

$$\frac{\hat{\mathbf{u}} - \sum_{q=0}^{J_i-1} \alpha_q \mathbf{u}^{n-q}}{\Delta t} = \sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{u}^{\mathbf{n-q}}), \tag{4.8}$$

$$\frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} = \nabla \overline{p}^{n+1}, \tag{4.9}$$

$$\frac{\gamma_0 \mathbf{u}^{n+1} - \hat{\hat{\mathbf{u}}}}{\Delta t} = \nu \nabla^2 \mathbf{u}^{n+1}. \tag{4.10}$$

The pressure is thus decoupled from the velocity in this time-advancement scheme, thereby avoiding the emergence of spurious pressure modes and the use of a staggered grid or the incorporation of stabilization terms into the governing equations [24].

The splitting procedure for ρ' consists of two steps analogous to Eqs.(4.8) and (4.10). The coefficients α_q, β_q of Eq. (4.8) and γ_0 of Eq. (4.10) correspond to a 3rd order Stiffly Stable Scheme (SS3) [74]. Their values may be found in references [74, 100]. Such a time-advancement scheme allows for a maximum CFL number as high as one.

In the first step (Eq.(4.8)), the non-linear term ($\mathbf{N}(\mathbf{u})$), defined on Eq. (4.1) is advanced explicitly via a third order stiffly stable scheme (SS3) [74]. In the second fractional step, the incompressibility constraint (Eq. (4.3)) is enforced by requiring that the second intermediate velocity $\hat{\hat{\mathbf{u}}}$ is divergence free, i.e. $\nabla \cdot \hat{\hat{\mathbf{u}}} = 0$. This enforcement is done by applying the divergence operator to both sides of equation (4.9), and consequently solving a Poisson pressure equation with Neumann type boundary conditions. Details on the solution of this system of equations are presented in [33]. Finally, in the third fractional step, the linear terms $\mathbf{L}(\mathbf{u})$ are solved with an implicit modified Helmholtz equation solver, in order to obtain the final velocity ($\mathbf{u}^{n+1}$) field at each time step. The values of the coefficients α_q, β_q of Eq. (4.8) and γ_0 of Eq. (4.10) can be found on [74, 100]. The weakly dissipative nature of such an approximation is helfpul in stability-sensitive under-resolved problems.

The quantity $\bar{p}^{n+1}$:

$$\int_{t_n}^{t_{n+1}} \nabla p' dt = \Delta t \nabla \bar{p}^{n+1} \quad . \tag{4.11}$$

is an intermediate scalar field, the pseudopressure, that ensures that the final velocity $\mathbf{v}^{n+1}$ is incompressible. Hereafter, we will refer to this quantity as the pressure. In Eq. (4.9), the incompressibility constraint is enforced, i.e. it is assumed that $\nabla \cdot \hat{\hat{v}} = 0$ and the Poisson equation is solved for the pressure:

$$\nabla^2 \bar{p}^{n+1} = \nabla \cdot \left(-\frac{\hat{\mathbf{u}}}{\Delta t} \right) = f. \tag{4.12}$$

The boundary conditions for the velocity field are enforced in Eqs. (4.8)-(4.10) and an analogous approach is followed for ρ'. However, the correct choice of boundary conditions for the pressure Poisson equation (PPE) Eq. (4.12) is dictated by the need for temporal accuracy of the splitting scheme, but most importantly, by the fulfillment of a

compatibility condition (see section 4.2.4). Satisfaction of the compatiability condition is of paramount importance for existence of a solution for $\overline{p}^{n+1}$ [54, 74, 101, 33]. For the temporal discretization used in this work, the appropriate boundary conditions for the PPE are given by [74]

$$\frac{\partial \overline{p}}{\partial n} = \mathbf{n} \cdot \left[\sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{u^{n-q}}) + \nu \beta_q \mathbf{L}(\mathbf{u^{n-q}}) \right] \tag{4.13}$$

where the coefficient β_q take the same values as in the SS3 scheme described on Eqs. (4.8)-(4.10). Further detail on the derivation of these boundary conditions and their role with respect to the compatibility condition is given in references [74, 33]. Guermond and Shen [59, 60] demonstrate that the splitting scheme (Eqs. (4.8), (4.9), (4.10) and (4.12)) is $O(\Delta t^2)$ accurate. Finally, note that, throughout the remainder of this text, we will use p to represent the (pseudo)pressure as defined in (4.11).

4.2.2 Spatial discretization

Spectral multidomain penalty method

In an under-resolved simulation relying on element-based higher-order accuracy discretization techniques, a preferred location for the appearance of spurious Gibbs oscillations are the physical boundaries and subdomain interfaces [65, 25]. Since at these locations the highest resolved Legendre modes are most oscillatory [12], the numerical noise, caused by aliasing associated with the nonlinear terms in the governing equations, is most likely to manifest itself. This issue is compounded by the strong enforcement of boundary conditions and across-subdomain communication (patching conditions) [25].

In such a problematic situation, SMPM provide an efficient means of enabling numer-

ical stability, without loss of high accuracy, at the subdomain interfaces by collocating both governing equation and boundary/patching condition at the physical boundary/subdomain interface interior. As a result, the solution transitions more stably and smoothly from the subdomain's boundary into its interior. Inherent in a penalty scheme is that boundary conditions are enforced weakly as is also C_0 and C_1 continuity of the function at the interfaces, i.e. the discretization is discontinuous [64, 25]. The weak error associated with the weak enforcement of boundary conditions and intra-subdomain continuity is no greater than the order of the numerical scheme [64, 25].

In what follows, we discuss the penalty formulation associated with each of the operators (i.e., non-linear advective, pressure, and viscous/diffusive) within the temporal discretization scheme described in §4.2.1. In the penalty formulation of the nonlinear and viscous terms, all relevant equations are written in terms of the *u*-velocity without loss of generality.

Non-linear advective operator The explicit nonlinear term advancement in Eq. (4.8) is treated as a hyperbolic equation and the associated formulation is adapted accordingly [25]. Writing the time derivative in continuous form, for the sake of compactness, for a point *k* located at a subdomain interface, we have [65]

$$\frac{\partial u^k}{\partial t} = \mathbf{N}(u^k) - \tau_k Q_(\mathbf{x})[\alpha_k u^k(\mathbf{x}, t) - g^k(\mathbf{x}, t)] \tag{4.14}$$

where τ_k are the penalty coefficients, $Q(\mathbf{x})$ are effectively Dirac delta functions which are non-zero only at the interfaces of the subdomain (where the penalty terms are nonzero). The coefficient α_k and the function $g^k(\mathbf{x}, t)$ are defined below.

The appropriate value of the penalty parameter τ_k is computed based on conservation of energy considerations of the linear advection equation [66]. Numerical stability is

established by setting

$$\tau_k = \begin{cases} \frac{1}{2\omega}\frac{2}{L_x}, & \text{at vertical interfaces} \\ \frac{1}{2\omega}\frac{2}{L_z}, & \text{at horizontal interfaces} \end{cases} \tag{4.15}$$

where L_x and L_z are the horizontal and vertical extents of the subdomain k, and $\omega = 2/N(N+1)$ is the GLL weight. If $\mathbf{n}$ is the vector normal to a subdomain interface i, and $U_p^{(i)}$ the velocity at a point p along the particular interface, the coefficient α_k is given by

$$\alpha_k = \begin{cases} |U_p^{(i)}| & \text{if} \quad U_p^{(i)} \cdot \mathbf{n} < 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.16}$$

Finally, the boundary/patching operators, $g^k(\mathbf{x}, t)$ of the k-th subdomain are given by

$$g^k(\mathbf{x}, t) = \begin{cases} \alpha_k u^*(\mathbf{x}, t) & \text{at interfaces} \\ \alpha_k f(\mathbf{x}, t) & \text{at physical boundaries} \end{cases} \tag{4.17}$$

where $u^*(\mathbf{x}, t)$ is the velocity (or density) at the corresponding interface of the neighboring subdomain, and $f(\mathbf{x}, t)$ is the externally prescribed value of the boundary condition at the subdomain interface under consideration.

Viscous/diffusive operator In this case, the starting point of our presentation is Eq. (4.10), which, on account of the fully implicit scheme for the viscous/diffusive terms, is recast as a modified Helmholtz equation

$$-\frac{\hat{\mathbf{u}}}{\gamma_0} = \frac{\nu\Delta t}{\gamma_0}\mathbf{L}(\mathbf{u^{n+1}}) - \mathbf{u^{n+1}} \quad . \tag{4.18}$$

Setting $\varepsilon = \nu\Delta t/\gamma_0$, the penalized form of equation (4.18) is

$$-\frac{\hat{\mathbf{u}}}{\gamma_0} = \varepsilon\mathbf{L}(\mathbf{u^{n+1}}) - \mathbf{u^{n+1}} - \tau_k^k Q_(\mathbf{x})[\alpha\mathbf{u^k}(\mathbf{x}, t) + \beta\varepsilon\mathbf{n} \cdot \nabla\mathbf{u^k}(\mathbf{x}, t) - g^k(\mathbf{x}, t)] \tag{4.19}$$

where τ_k is the penalty coefficient, and $Q(\mathbf{x})$ has the same definition as in the advective term treatment. The constant coefficients α and β are weights for the continuity at the

function and derivative respectively, which in practice are set to one. The penalty coefficient τ_k is also computed through energy conservation considerations for the discrete diffusion equations [65]. At the subdomain interfaces, τ_k, lies within the bounds

$$\frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa - 2\sqrt{\kappa^2+\varepsilon\kappa}\right]\frac{2}{L_x^I} \leq \tau_k \leq \frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa + 2\sqrt{\kappa^2+\varepsilon\kappa}\right]\frac{2}{L_x^I} \tag{4.20}$$

where $\kappa = \omega\alpha/\beta$ and ω has the same definition as in the advective term treatment. As indicated by Eq. (4.20), one can experiment with the exact choice of τ_k and, thus, the degreeoof enforcement of the continuity of the function and derivative between subdomains. The closer the value of τ_k to the upper limit of Eq. (4.20), the stronger the enforcement at the interfaces. In practice, for the subdomain interfaces, the value of the penalty coefficient is chosen to provide maximum stability and efficiency (the latter in terms of the associated iterative solver), and its value is typically case dependent. For boundary conditions, the following penalty coefficients are used [66, 25]

$$\tau_k = \begin{cases} \frac{\varepsilon}{\alpha\omega^2}\left(\frac{2}{L}\right)^2 & \text{for Dirichlet boundary conditions} \\ \frac{1}{\beta\omega}\left(\frac{2}{L}\right) & \text{for Neumann boundary conditions} \end{cases} \tag{4.21}$$

where L represents the dimension of the subdomain normal to the boundary. Finally, the boundary/patching operators, $g^k(\mathbf{x},t)$ of the k-th subdomain are given by

$$g^k(\mathbf{x},t) = \begin{cases} \gamma\mathbf{u^k}(\mathbf{x},t) + \delta\varepsilon\mathbf{n}\cdot\nabla\mathbf{u^k}(\mathbf{x},t) & \text{at subdomain interfaces} \\ f(\mathbf{x},t) & \text{at physical boundaries} \end{cases} \tag{4.22}$$

where γ and δ are constant coefficients with the same function as α and β.

Pressure Poisson equation The penalty formulation of the PPE (Eq. (4.12)) is similar to the one presented above for the viscous/diffusive operator. As in previous work [71, 33], one treats the elliptic PPE as a steady state version of the diffusion equation and sets the coefficient ε equal to 1.

$$\nabla^2 p^k + \tau_k Q(\mathbf{x})\left(\alpha p^k + \beta\mathbf{n}\cdot\nabla p^k - g^*(\mathbf{x})\right) = f^k, \tag{4.23}$$

where the coefficients τ_k, $Q(\mathbf{x})$, α and β have the same definitions as those appearing in Eq. (4.20), and f is the forcing term defined on Eq. (4.12). The upper and lower bounds for τ_k are the same as those appearing in Eq. (4.20). As in the viscous/diffusive term treatment, we chose an intermediate value of τ_k which provides maximum numerical stability and, most importantly, optimum efficiency of the iterative PPE solver. Finally, the boundary/patching operators $g^*(\mathbf{x})$ of the k-th subdomain are

$$g^*(\mathbf{x}) = \begin{cases} \gamma p^* - \delta \mathbf{n} \cdot p^* & \text{for patching conditions} \\ f_p(\mathbf{x}, t) & \text{for boundary conditions} \end{cases} \tag{4.24}$$

Note that, as indicated in section 4.2.1, the values of the boundary condition $f_p(\mathbf{x}, t)$ are those given by Eq. (4.13) which satisfy the compatibility condition presented in section 4.2.4.

As a general conclusion for this section, there is no special treatment for the subdomain corners. They are treated as any other point along each of the interfaces on which they reside.

4.2.3 Additional Stabilization Measures

Incorporation of a multidomain penalty scheme in the flow solver provides it with enhanced numerical stability properties that enable higher degrees of under-resolution. However, this form of improved numerical stability is restricted only to the vicinity of subdomain interfaces and physical boundaries [25]. For a higher-order polynomial discretization, additional measures need to be implemented to ensure numerical stability of the solution in the interior of a subdomain. To this end, dealiasing by padding [14, 12] is applied during the computation of the nonlinear terms (equation (4.8)) along

with spectral filtering after each of the fractional steps (Eqs. (4.8)-(4.10)) of the temporal discretization. Finally, an interfacial averaging procedure [25] is applied at the subdomain interfaces.

Dealiasing

This type of error is inherent in the convolution sums associated with the modal representation of the non-linear terms **N**(**u**) (see Eq. (4.1) and reference [14]). Higher modes, beyond the highest resolved mode for the particular GLL grid at hand, are generated during the computation of the non-linear terms in Eq. (4.1) and are aliased onto highest resolved modes [12, 84]. In an under-resolved simulation, where any viscously-dominated scales are not captured, this error accumulates over time through the persistent action of the non-linear terms, often to catastrophic numerical instability [53]. For the specific case of the skew-symmetric form of the non-linear terms considered in this work, the two products, $\mathbf{u} \cdot \nabla \mathbf{u}$ and $\nabla \cdot (\mathbf{uu})$, have to be dealiased. For the sake of simplicity, the dealiasing procedure will be shown for $\mathbf{u} \cdot \frac{\partial \mathbf{u}}{\partial x}$. It can be extended in a straightforward manner to all non-linear products in Eq. (4.1) and the advective terms in Eq. (4.2). The dealiasing technique implemented in this work is outlined below in individual steps, following the procedure presented in [14]:

1. In each two-dimensional subdomain, construct the $(N+1) \times (N+1)$ matrices, $\mathbf{U}_N$ and $\frac{\partial \mathbf{U}_N}{\partial x}$, which contain the nodal values of the $N-th$ order inteprolating polynomial function $u^N(x,z,t)$ and its x-derivative, respectively.
2. Compute the matrices of modal coefficients associated with $\mathbf{U}_N$ and $\frac{\partial \mathbf{U}_N}{\partial x}$ on each

subdomain by applying a two-dimensional Legendre transform:

$$\widetilde{\mathbf{U}_N} = \mathbf{M}_N \mathbf{U}_N \mathbf{M}_N^T \tag{4.25}$$

$$\frac{\widetilde{\partial \mathbf{U}_N}}{\partial x} = \mathbf{M}_N \frac{\partial \mathbf{U}_N}{\partial x} \mathbf{M}_N^T \,. \tag{4.26}$$

$\mathbf{M}_N$ is a matrix of dimension $(N+1)\times(N+1)$ that performs the discrete polynomial transform from nodal form to modal form [7]

3. In each subdomain, consider a finer GLL grid of polynomial order M. Choose $M \geq 3N/2 + 1/2$ for an odd value of N , or $M \geq 3N/2 + 1$ for an even value of N. Pad the modal expansions in $\tilde{\mathbf{U}}_N$ and $\frac{\partial \tilde{\mathbf{U}}_N}{\partial x}$ by adding zeros to all modes of order greater than N and construct the matrices $\tilde{\mathbf{U}}_M$ and $\frac{\partial \tilde{\mathbf{U}}_M}{\partial x}$ as

$$\widetilde{u_{ij}^M} = \begin{cases} \widetilde{u_{ij}^N} & \text{for } i, j = 1, \ldots, N \\ 0 & \text{for } i, j = N+1, \ldots, M \end{cases} \tag{4.27}$$

$$\frac{\widetilde{\partial u_{ij}^M}}{\partial x} = \begin{cases} \frac{\widetilde{\partial u_{ij}^N}}{\partial x} & \text{for } i, j = 1, \ldots, N \\ 0 & \text{for } i, j = N+1, \ldots, M \end{cases} \tag{4.28}$$

4. Go back to nodal form with the new expanded discrete modal basis $\tilde{\mathbf{U}}_M$ and $\frac{\widetilde{\partial \mathbf{U}_N}}{\partial x}$

$$\mathbf{U}_M = \mathbf{B}_M \widetilde{\mathbf{U}_M} \mathbf{B}_M^T \tag{4.29}$$

$$\frac{\partial \mathbf{U}_M}{\partial x} = \mathbf{B}_M \frac{\widetilde{\partial \mathbf{U}_M}}{\partial x} \mathbf{B}_M^T \tag{4.30}$$

where $\mathbf{B}_M$ is a matrix, of dimension $(M+1)\times(M+1)$, that performs the transformation from modal space to nodal space [7] on the higher, M-th order grid.

5. Perform the multiplication

$$\left(\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}\right)_M = \mathbf{U}_M \frac{\partial \mathbf{U}_M}{\partial x} \tag{4.31}$$

on the M-th order grid.

6. Convert the discrete function $\left(\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}\right)_M$ onto its modal form

$$\left(\widetilde{\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}}\right)_M = \mathbf{M}_M \left(\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}\right)_M \mathbf{M}_M^T \tag{4.32}$$

7. Remove the aliased modes from the M-th order modal expansion of the nonlinear product through padding

$$\left(\widetilde{u_{i,j}\frac{\partial u}{\partial x}_{i,j}}\right)_N = \left(\widetilde{u_{i,j}\frac{\partial u}{\partial x}_{i,j}}\right)_M \qquad \text{for } i, j = 1, \ldots, N \tag{4.33}$$

8. Finally, return the nonlinear product into nodal form on the N-th order grid:

$$\left(\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}\right)_N = \mathbf{B}_N \left(\widetilde{\mathbf{U}\frac{\partial \mathbf{U}}{\partial x}}\right)_N \mathbf{B}_N^T$$

As shown in section 4.3.3, the dealiasing procedure takes importance when non-linear effects dominate the dynamics of the flow (i.e., high Reynolds number). If it is not applied, the solution produced by the solver can contain small scale errors, which can potentially evolve in time and give an unrealistic picture of the flow.

Spectral filtering

On a particular subdomain, the equivalent modal form of Eq. (2.1) is

$$u(x, z, t) = \sum_{i=0}^{N} \sum_{j=0}^{N} \tilde{u}(x_i, z_j, t) P_i(x) P_j(z) \tag{4.34}$$

where $\tilde{u}(x_i, z_j, t)$ are Legendre modal coefficients evaluated at point (x_i, z_j), and P_k, $k = 0, \ldots, N$ is the k-th Legendre polynomial. Within each subdomain, the filter acts on this expansion series by multiplying each one of the modal coefficients with a weight function:

$$u^f(x, z, t) = \sum_{i=0}^{N} \sum_{j=0}^{N} \sigma_i \sigma_j \tilde{u}(x_i, z_j, t) P_i(x) P_j(z) \,, \tag{4.35}$$

where $\sigma_k, k = 0, \ldots, N$ is a spectral filter function of order p. This study employs the commonly used exponential spectral filter [53, 7, 25]:

$$\sigma_k = \exp\left[-\alpha\left(\frac{k}{N}\right)^p\right], \tag{4.36}$$

where $\alpha = -\ln \epsilon_m$, and ϵ_m is the machine precision. From the discrete perspective, i.e. with respect to its application to the $(N+1)\times(N+1)$ matrix $\mathbf{U}_N$ of nodal values on the particular subdomain, the filtering procedure consists of the following steps:

1. Compute modal coefficients on each subdomain (see previous section)

$$\widetilde{\mathbf{U}_N} = \mathbf{M_N U_N M_N^T} \tag{4.37}$$

2. Multiply each of the modal coefficients $\widetilde{\mathbf{U}}_N$ by the corresponding filter matrix $\mathbf{L_N} = diag(\sigma_0, \ldots, \sigma_N)$

$$\widetilde{\mathbf{U}_N^f} = (\mathbf{L_N M_N})\mathbf{U_N}(\mathbf{M_N^T L_N^T}) \tag{4.38}$$

3. Recover the nodal representation of the filtered function

$$\mathbf{U}_N^f = (\mathbf{B_N L_N M_N})\mathbf{U_N}(\mathbf{M_N^T L_N^T B_N^T}) \tag{4.39}$$

$$= (\mathbf{B_N L_N M_N})\mathbf{U_N}(\mathbf{B_N L_N M_N})^{\mathbf{T}} \tag{4.40}$$

Defining $\mathbf{F}_N = \mathbf{B_N L_N M_N}$, the filtering procedure is effectively reduced to

$$\mathbf{U}_N^f = \mathbf{F}_N \mathbf{U}_N \mathbf{F}_N^T \, . \tag{4.41}$$

A common concern with the implementation of spectral filtering in spatially continuous spectral element methodologies is that filtering does not preserve the patching and boundary conditions and thus specific measures need to be adopted [87, 11, 7]. Such a concern does not exist when using the inherently discontinuous penalty method because

the error induced by the filtering operation is of the same order as the penalty scheme [64], i.e. minimal.

In the incompressible spectral multidomain solver presented in this work, spectral filtering is applied across all three fractional steps when advancing the solution from time level (n) to level ($n+1$). First, further suppress aliasing effects, filtering is applied after advancing the non-linear terms in (4.8). The solution to the PPE (4.12) is then filtered to smooth out any errors induced at the subdomain interfaces due to the discrete estimate of $\nabla \cdot \hat{\mathbf{u}}$ [23]. Finally, we filter the solution of (4.12). Although the filtering of the result of the explicit nonlinear term advancement is most critically important, we find that filtering in the subsequent two fractional steps, which involve linear operators, provides optimal robustness to our solver.

Interfacial averaging

Despite the visible gains in solver robustness enabled by the penalty scheme, spectral filtering and de-aliasing, numerical instability will intermittently appear in the form of gradually growing spikes at a small number of subdomain interfaces. These events are attributed to the inherently discontinuous nature of the penalty scheme and the modification of the interfacial values of the solution following spectral filtering. These interfacial singularities, though limited in space, can grow catastrophically in time. As a countermeasure, an interfacial averaging technique is used [28, 25] when the following criterion is met:

$$\frac{|u_i^k - u_i^*|}{|u_i^k + u_i^*|} > C_{ave} \ , \tag{4.42}$$

where u_i^k represents the value of the solution (velocity component or density) located at one side of the interface i belonging to subdomain k, u_i^* is the corresponding value

on the same interface i originating from the neighboring subdomain. If Eq. (4.42) is satisfied, an averaging operation is performed

$$u_i^k = u_i^* = \begin{cases} 0.5 \times (u_{i-1}^k + u_{i+1}^*) & \text{for vertical interfaces} \\ 0.5 \times (u_{i-N+1}^k + u_{i+N+1}^*) & \text{for horizontal interfaces} \end{cases} \tag{4.43}$$

In practice, we set the coefficient $C_{ave} \leq 0.005$, which results in a very small percentage of the interfacial nodes being subject to the averaging procedure.

4.2.4 Discrete Pressure Poisson equation

The numerical solution of the non-symmetric linear system that arises from the SMPM discretization of the PPE, of Eq. (4.12), is by far the most demanding and costly component of our Navier-Stokes solver. In this section, the most important building blocks for iterative numerical solution of SMPM-discretized PPE are mentioned. For a detailed analysis of the solution and the main characteristics of the problem, the reader is referred to [33] or Chapter 5.

This is a classical problem encountered on the numerical solution of the incompressible Navier-Stokes equations when a projection technique, such as the one used in this work (see section 4.2.1), is used to decouple the velocity field from the pressure field. The result is a Poisson equation of the form (4.12) with Neumann type boundary conditions (4.13). From the continuous perspective, an in order to be able to solve the equation, a compatibility condition has to be fulfilled. This condition requires the volume integral of the forcing terms to be equal to the net flux along the boundaries. In this work, we followed the procedure outlined by Karniadakis et al. [74], provides a detailed presentation on how to compute these values for the PPE, in the context of the temporal discretization of section 4.2.1, as ultimately indicated in Eq. (4.13) of this document.

At the discrete level, the solution of the linear system of equations of the discretized pressure Poisson equation, if exists, is only determined up to an additive constant [24] which translates into the associated Poisson matrix being singular. Moreover, because the compatibility condition is never satisfied exactly, the system of equations has no solution, i.e. is inconsistent. Consistency of the system is enabled by making the right hand side of equation the PPE orthogonal to the left null singular value $\mathbf{u_0}$ of the corresponding Poisson matrix [101].

Once the system of equations is made consistent, a preconditioner matrix **M** has to be constructed to obtain an efficient solution in the framework of the GMRES iterative technique. In this work, we use a custom-designed a two-level preconditioner composed by a coarse-grid component, that accounts for $h-$refinement, and a fine non-overlapping additive Schwarz component, that accounts for $p-$refinement [33].

4.3 Numerical results

Four benchmarks are examined to validate the SMPM-discretized incompressible Navier-Stokes solver. Three of them involve a non-stratified fluid (Taylor vortex, lid-driven cavity and double shear layer), whereas the remaining one include a background density stratification (propagation of an internal solitary wave of depression type). For two benchmark cases, a quantitative assessment of solver accuracy is possible by comparing to an analytical solution. The remaining benchmarks demonstrate accuracy and robustness of the flow solver for problems with strong nonlinearities and/or fully non-periodic boundary conditions.

All the benchmark simulations require the use of the two-level preconditioner described in section 5.6 (Eq. (5.49)) for the iterative solution of the PPE linear system of equations to converge. The total number of iterations on each time step was found to be case-dependent. Typically, this iteration count is below 90 iterations for the highly nonlinear cases, and around 40 iterations for the low-Reynolds number cases. In terms of the CPU time per timestep, approximately 80% of it was dedicated to solving the PPE. The computation of the nonlinear terms becomes three-fold more expensive when a dealiasing procedure is included.

4.3.1 Taylor vortex

The choice of this benchmark is motivated by the availability of an analytical solution, which allows us to assess the accuracy of our solver. The flow field initially consists of a periodic array of vortices whose velocity field diffuses out with time. The left panel of Fig. 4.1 shows the general structure of the streamlines of this flow. The domain is a periodic box with dimensions $(x, z) \in [-1, 1]^2$, $0 \leq t \leq 1$. The Reynolds number of the simulation is $Re = U_m L/\nu = 100$, where U_m is the magnitude of the maximum velocity on the domain, L is the diameter of the vortex, and ν is the kinematic viscosity. The initial condition is computed from the analytical solution [110, 75] for $t = 0$.

$$u(t, x, z) = -\cos(\pi x)\sin(\pi z)\exp\left(\frac{-2\pi^2 t}{Re}\right) \tag{4.44}$$

$$w(t, x, z) = \sin(\pi x)\cos(\pi z)\exp\left(\frac{-2\pi^2 t}{Re}\right) \tag{4.45}$$

$$p(t, x, z) = -\frac{\cos(2\pi x) + \cos(2\pi z)}{4}\exp\left(\frac{-4\pi^2 t}{Re}\right) \tag{4.46}$$

The right panel of Fig. 4.1 shows the convergence plot of the L_∞-error norm of the horizontal velocity as a function of the polynomial degree N (p-refinement over a fixed number of subdomains) and for different mesh sizes (i.e., h-refinement). As expected,

the results show exponential convergence, a characteristic of spectral methods, for every level of *h*-refinement.

In this case the importance of the two-level preconditioner seemed not to be as dramatic

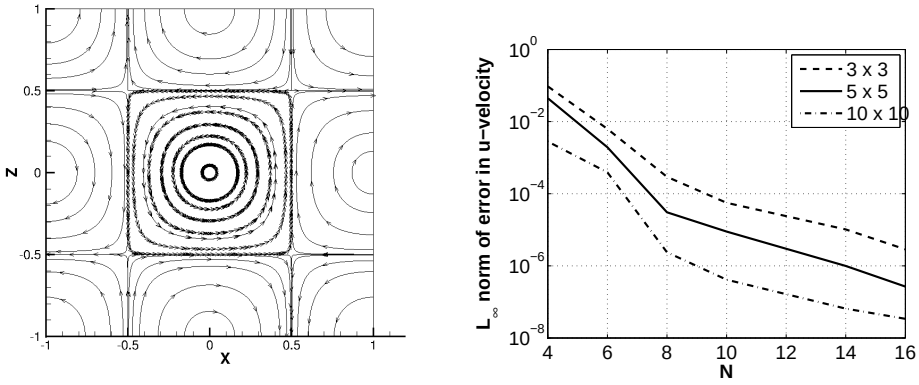


Figure 4.1: Taylor Vortex problem. Left: Structure of the streamlines of the flow. Right: L_∞-error norm of the horizontal velocity as a function of polynomial degree $N$and level of $h-$refinement (3×3, 5×5, and 10×10 subdomains) for $Re = 100$.

as in the other benchmark cases. The authors conjecture that this behavior is due to the smoothness of the solution and the double periodicity of the domain.

4.3.2 Lid-Driven Cavity flow

The lid-driven cavity is a standard benchmark for testing an incompressible flow solver subject to non-periodic boundary conditions [9, 31]. The computational domain is a square box defined over $(x, y) \in [0, 1]^2$. The top boundary moves with a steady velocity, whereas the lateral boundaries and the bottom of the domain are impermeable walls (i.e., no-slip boundary conditions). The Reynolds number for this case is $Re = UL/\nu = 10^3$, where U represents the characteristic velocity at the top of the cavity, L is the characteristic length of the box, and ν is the kinematic viscosity of the fluid. To avoid

the singularities that arise at the top corners due to discontinuities of the u velocity when a spectral method is used [9, 108], we consider a modified lid-driven cavity [100], where the top boundary condition is given by

$$u(x,1) \quad = \quad -16x^2\left(1-x^2\right) \tag{4.47}$$

The structure of the streamlines of the velocity field are shown in Fig. 4.2 for a particular number of uniformly sized subdomains (10×10), and a varying value of polynomial approximation N ($p-$refinement). As expected, the solution agrees well with that corresponding to the previously published spectral solution of Botella et al.[9]. Viscous diffusion from the moving boundary generates a large vortex in the center of the domain. Two smaller vortices then form at each corner, the one at the bottom right corner being the visibly larger one of the two. Once the resolution is increased (Figs. 4.2b, and 4.2c), the vortices are better resolved, which is translated into a more defined structure.

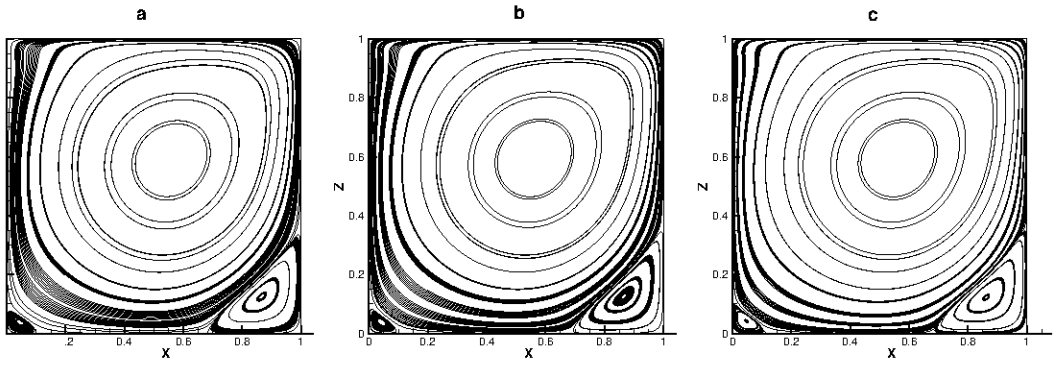


Figure 4.2: Streamlines on the Lid-driven cavity flow at $t = 20$ sec for 10×10 subdomains and $Re = 1000$. a) $n = 9$. b) $n = 12$. c) $n = 15$.

In order to validate the performance of the method for localized resolution, the same case with a non-uniform grid distribution was executed, and the streamlines of the velocity field are shown in Fig. 4.3.

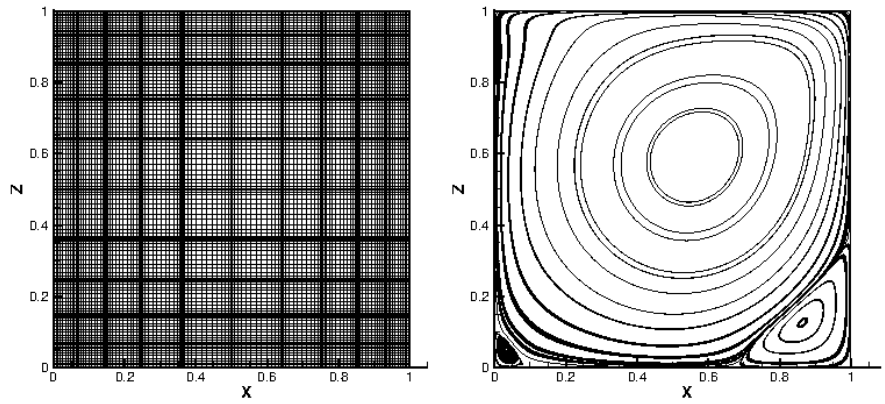


Figure 4.3: Streamlines on the Lid-driven cavity flow at $t = 20$ sec for a 10×10 grid with non-uniform subdomains, $n = 14$, filter order $p = 11$, and $Re = 1000$.

Running with a larger number of uniform or non-uniform sized subdomains (in the latter case, with smaller domains focused at the bottom corners) or with a larger value of N produces the same streamline structure at steady state. As a fina remark, in this test case, the use of the coarse component of the two-level preconditioner is imperative to ensure an efficient iterative solution of the PPE, as the level of h-refinement (i.e. number of subdomains) is already significant, in addition to the non-periodic boundary conditions, the presence of pronounced non-linear effects, and a non-smooth behavior of the flow process. Without this component, the iterative solution of the PPE does not converge. In addition, this test case served as a platform to check the importance of spectral filtering for the stability of the solver.

4.3.3 Double shear layer

The double shear layer is a commonly used benchmark [123, 89] to test the performance of the solver when non-linear effects have a dominant impact in the simulation. It is also a useful platform to assess the effect of the stabilization techniques, discussed in section 4.2.3, on the flow structure at the final time. The physical domain is a doubly-periodic box defined over $(x, z) \in [-1, 1]^2$. Two shear layers in the horizontal velocity field are positioned symmetrically around the horizontal centerline. The two layers are subject to an initial periodic vertical velocity perturbation which, by means of an inviscid instability [119], gives rise to two billow-like vortices. The interaction between each of the vortices and its periodic image gives rise to a straining motion that forces the braid region in-between them to become progressively thinner, without, at the Reynolds numbers considered, undergoing the counteracting effect of viscosity [125]. The braid region gradually becomes marginally resolved, thereby requiring a robust spatial discretization scheme such as the SMPM to avoid numerical instability.

The initial conditions are given by [123]

$$u = \begin{cases} \tanh(\varepsilon(z + 0.5)) & \text{for } z \leq 0 \\ \tanh(\varepsilon(0.5 - z)) & \text{for } z > 0 \end{cases} \tag{4.48}$$

$$w = \delta \cos(\pi x) \tag{4.49}$$

where $\varepsilon = 40$, with $1/\varepsilon$ providing a measure of the thickness of the shear layer and $\delta = 0.05$ is the amplitude of the perturbation in w. The Reynolds number for this simulation is $Re = UL/\nu = 10^4$, where U is the maximum horizontal velocity, L is the characteristic length of the vortices and ν is the kinematic viscosity of the fluid. The vorticity field at time $t = 2$ is shown in Fig.4.4 for filter orders $p = 12$ and $p = 10$.

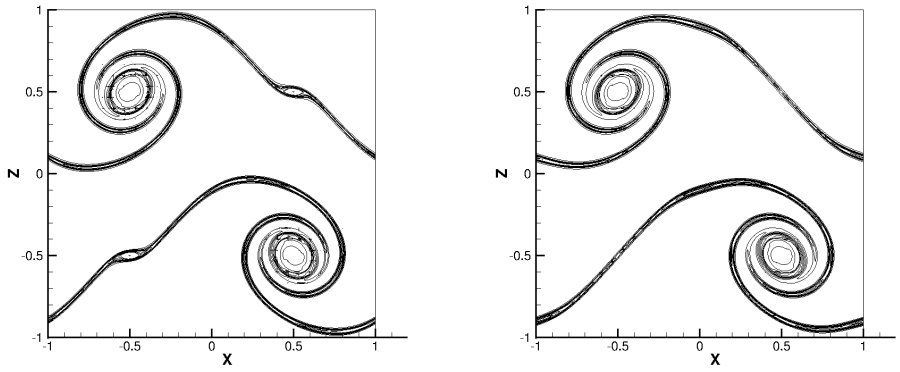


Figure 4.4: Vorticity contours for the dealiased solution for the double shear layer test case at $t = 2s$ for 30×30 subdomains, $n = 14$, $Re = 10^4$. Left: Filter order $p = 12$. Right: Filter order $p = 10$

The vorticity field structure in the right panel of Fig. 4.4 is in agreement with those presented elsewhere [123, 89]. Furthermore, as shown in the left panel of the same figure, inaccurate, i.e. too high, a value of the filter order p can artificially modify the flow dynamics; spurious modes are generated, focused on the subdomain corners close to the braids. These spurious higher modes lead to the formation of unphysical vortices, which grow in time, tearing apart the braids and leading to an inaccurate representation of the flow field. A convergence test was also done, and shown in Fig. 4.5, in terms of $p-$refinement to see how the solution behaves when a different polynomial degree is used for a fixed number of subdomains.

As shown in Fig. 4.5a, a low order polynomial degree leads to a poor resolution of the vortices and braids of the layers. Once the resolution increases (Fig. 4.5b) the expected structure of the vortices is obtained, and it is even better resolved with a higher resolution (Fig. 4.5d and 4.5d). In addition, we have found that the benefits of the dealiasing technique become important when a robust numerical solution is required at as high as Reynolds number as possible. For the double shear layer, at $Re = 10^5$ or higher and resolutions comparable to those considered in Figure 4.4, dealiasing is imperative for a

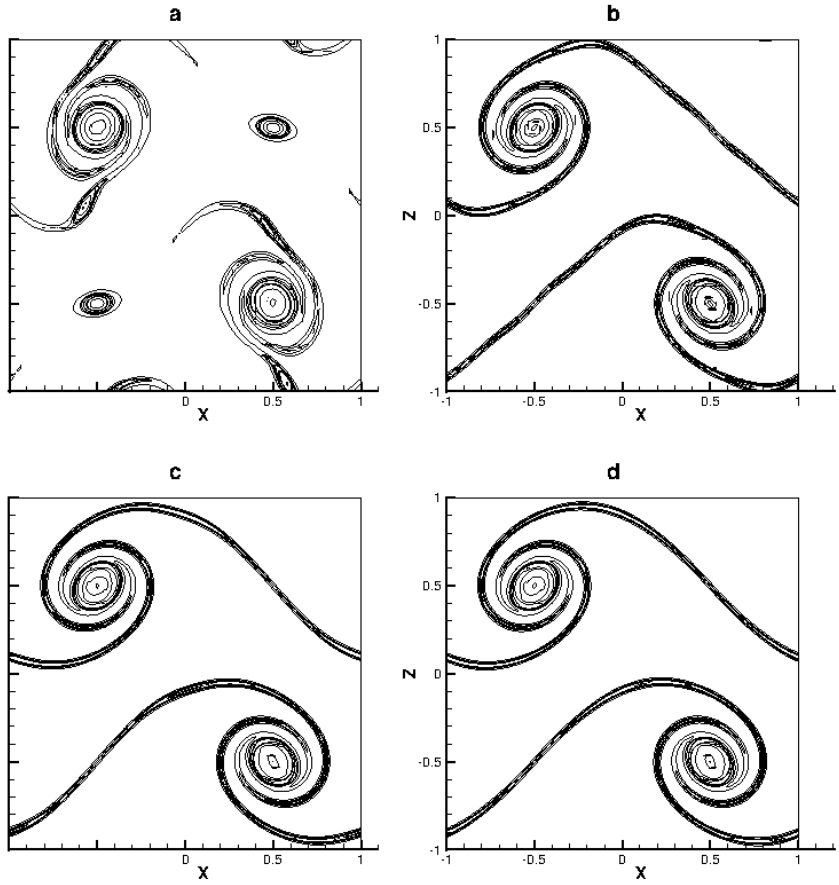


Figure 4.5: $p-$refinement convergence test for the double shear layer. Vorticity contours at $t = 2s$ for 30×30 subdomains and $Re = 10^4$. a) $n = 6$, b) $n = 9$, c) $n = 14$, d) $n = 19$.

both numerically stable and accurate solution. Finally, the simulation in figure 4.4 used uniformly sized subdomains. Additional simulations were performed by positioning shorter subdomains at the vertical levels of the billow vortices, with subdomain heights becoming as small as 1/4 the height of those used in figure 4.4; an identical vorticity field structure resulted.

4.3.4 Propagation of a solitary non-linear internal wave of depression type

Large-amplitude, long internal solitary waves (ISW) are horizontally propagating waves guided by the top and bottom surfaces of the stably stratified ocean or lakes [63, 8]. Such waves are characterized by a distinct balance between strong nonlinearity and physical dispersion which allows them to propagate over long distances while maintaining a very steep waveform. The generation, propagation and dissipation of ISWs are phenomena of immediate relevance to physical oceanography and limnology.

From a purely numerical standpoint, a freely propagating ISW is an ideal platform for diagnosing the dissipative and dispersive properties of a particular spatiotemporal discretization scheme for the incompressible Navier-Stokes equations under the Boussinesq approximation. One has to simply track the propagating the wave in an environment devoid of any mechanisms that will force the wave to dissipate, disperse or undergo wave-scale structural transformations. Such mechanisms include bottom friction, variable bathymetry (bottom topography) and oncoming currents, among others. Moreover, the propagation of a fully-nonlinear, internal solitary wave in a stably stratified fluid is a phenomenon of immediate relevance to physical oceanography and limnology [63].

The velocity and density fields of a fully-nonlinear, fully-nonhydrostatic internal solitary wave field in the Boussinesq limit can be obtained by solving the Dubreil-Jacotin-Long (DJL) nonlinear eigenvalue problem as given by

$$\nabla^2\eta + \frac{N^2(z-\eta)}{c^2}\eta = 0 \tag{4.50}$$

with

$$\eta = 0 \text{ at } z = 0 = H$$

and

$$\eta \to 0 \text{ as } x \to \pm\infty$$

where $\eta(x, z)$ is a vertical displacement of isopycnal (iso-density) surfaces relative to a moving frame of reference with the constant speed c in the horizontal, $x-$direction; H is a constant total depth of the fluid column; $N^2(z)$ is the Brunt-Väisälä frequency as defined by

$$N^2(z) = -\frac{\bar{\rho}(z)g}{\rho_0}, \tag{4.51}$$

where $\bar{\rho}(z)$ is an equilibrium density distribution offset from the reference density ρ_0 and g is a gravitational constant taken as $g = 9.81[m/s^2]$ [85]. Given a solution set (c, η) for the wave phase speed and the wave-induced isopycnal displacement, the corresponding solution for the wave velocity/density fields is obtained through the relation

$$(u, w, \rho) = (-c\eta_z, c\eta_x, \rho_0 + \bar{\rho}(z - \eta)) \tag{4.52}$$

The eigenvalue problem (4.50) is solved by using the Matlab program [29] that implements a nonlinear optimization algorithm formulated by Turkington et al. [122].

The DJL equation is a result of a rigorous derivation from the inviscid, non-diffusive equations of fluid motion without any asymptotic projections [122]. Therefore, a wave solution obtained from the DJL equation is an exact solution to the incompressible Euler equations and is expected to decay in a, numerically simulated, viscous and diffusive environment. It is thus reasonable to expect that the numerical solution of the Navier-Stokes equations initialized with an ISW computed through the DJL equations solver will asymptotically preserve the corresponding inviscid (DJL) solution in the limit of vanishing viscosity and diffusivity.

The computational domain is defined over $(x, z) \in [0, 24] \times [0, H = 2]$ with units in

meters. We impose periodic boundary conditions in the horizontal $x-$direction. Periodicity in the streamwise direction allows us to track the wave over long propagation distances, provided of course that the domain length is sufficiently longer that the actual ISW wavelength. Impermeable, nondeformable, free-slip boundary conditions for velocities ($u_z = 0, w = 0$) along both the top ($z = H$) and bottom ($z = 0$) boundaries of the domain are used. We choose a no-flux condition for the density, $\rho_z = 0$ at the top and bottom surfaces. However, we find that our numerical results are essentially the same even when the no-perturbation condition $\rho = 0$ is used instead. With the use of the stress-free boundary conditions, formation of viscous boundary layers can be avoided and the only physical mechanisms for wave decay are viscous diffusion of momentum and mass diffusion. The equilibrium density profile is prescribed by the analytical function

$$\bar{\rho}(z) = -20\tanh((z - 0.2667)/0.1333) \tag{4.53}$$

with the reference density $\rho_0 = 1020[kg/m^3]$. We set the amplitude of the test wave to $0.3467H$, which is quite large, close to the limit amplitude allowed by Eq.(4.50) for the given stratification. Given the present wave amplitude and the density stratification, the wave speed is found to be $|c| = 0.400928[m/s]$ after solving Eq.(4.50). The corresponding wave field solution is adopted as an initial condition for the incompressible Navier-Stokes solver and is positioned at the center of the domain (see Figure 4.6-a). This inviscidly-balanced, initial state begins to propel itself in a single direction depending on the sign of the phase speed, c (i.e., positive $x-$direction, if $c > 0$; negative $x-$direction, otherwise). The computational domain is divided into forty equally-spaced subdomains in x and four equally-spaced subdomains in z with a polynomial order of $N = 32$ in both x and z directions for each domain. For later reference we define here the wave Reynolds number Re_w in terms of the wave speed c and the total depth H, as given by $Re_w = cH/\nu$, where ν is a kinematic viscosity. The density diffusivity κ is chosen to

be equal to the kinematic viscosity for all test cases in this section.

Figure 4.6 shows the density contours taken at $t = 120[s]$ (after $40,000$ steps with a timestep $\Delta t = 0.003[s]$) for $Re_w = 1.6 \times 10^5$ and $Re_w = 4 \times 10^4$ along with the initial density contour at $t = 0$. During this time interval, the wave has traveled for a distance corresponding to ten wavelengths (i.e. the wave has traveled two cycles through the present, $x-$periodic domain). It is evident from the figure that the shape of the wave is preserved quite well, and no unphysical, dispersive waves are observed after such long time integration. Restricting ourselves to visual inspection, no visible differences in the wave shape across Reynolds numbers are distinguishable in the present figure.

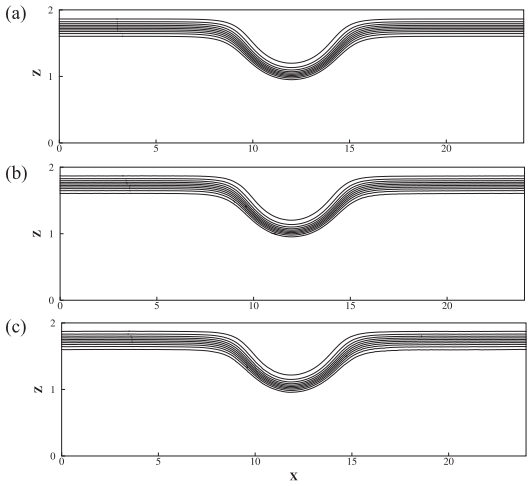


Figure 4.6: Snapshots of density contours of propagating internal solitary wave: a) initial condition at $t = 0[s]$; b) at $t = 120[s]$ for $Re_w = 1.6 \times 10^5$; and c) at $t = 120[s]$ for $Re_w = 4 \times 10^4$. The wave propagates to the left, and the domain is periodic in $x-$direction. Ten equally-spaced contour levels in the range $[1005, 1035][kg/m^3]$ are shown.

In Figure 4.7 ab we measured the amplitude ζ_{max} and the wavelength λ as functions of time t for $Re_w = 1.6 \times 10^5$ and $Re_w = 4 \times 10^4$. The wave amplitude is defined as the maximum displacement of the thermocline (an isopycnal line that passes through the

location in the vertical where the density gradient is the maximum in an undisturbed fluid). The wavelength is then defined as,

$$\lambda = \frac{\int_{\Omega} \zeta(x,t)dx}{\zeta_{max}}, \tag{4.54}$$

where $\zeta(x,t)$ is a displacement function of the thermocline. These wave properties, normalized by their respective initial values ζ_0 and λ_0 in the figure were extracted from the numerical solution by means of spline interpolation. From Figure 4.7 ab, both amplitudes and wavelength generally decay as the time increases, and the rate of the decay is slower for the lower viscosity case ($Re_w = 1.6 \times 10^5$), as expected.

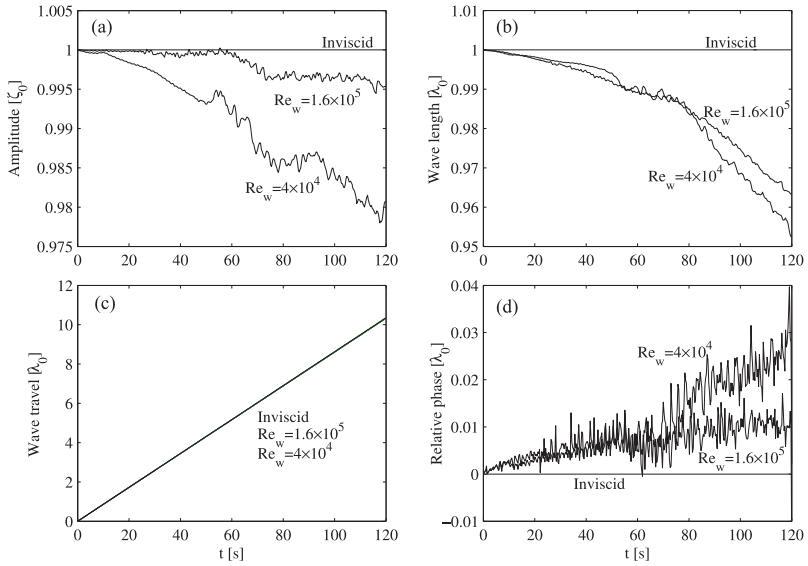


Figure 4.7: Time histories of wave properties obtained from numerical simulations: a) wave amplitude normalized by its initial value; b) wavelength normalized by its initial value; c) wave travel normalized by the initial wavelength; and d) wave phase difference from the inviscid limit.

Small, high frequency fluctuations appearing in Figure 4.7 are primarily caused by discontinuities in the numerical solution inherently present at the subdomain interfaces on account of the penalty treatment. When the wave passes through a subdomain interface, some weak numerical noise is generated through these discontinuities. This noise,

not entirely removed by spectral filtering, remains in the domain, weakly modifying the numerical solution over long time integration. Note that in the present simulations we used rather a weak, exponential filter of order $p = 16$. Although the above numerical noise can be fully eliminated by further reducing the order of the spectral filter, such tuning toward better appearance of the numerical results is not pursued here, as our primary goal is the conservation of the physical properties of a freely propagating wave in weakly-dissipative media with minimal artificial damping.

Figure 4.7-c exhibits the distance travelled by the wave trough as a function of time. This figure, at the leading order, suggests that the wave propagation speed is the same with the speed at the inviscid limit (associated with the solution of the DJL equation) for the values of Re_w considered here. For a more detailed comparison, the relative wave phase (i.e., the difference of the actual simulated wave trough location from that expected in the inviscid limit) is calculated and shown in Figure 4.7-d. The result is, again, normalized by the initial wavelength. The numerical noise mentioned above is again presente here. However, most importantly, the phase difference becomes smaller for the less viscous case ($Re_w = 1.6 \times 10^5$), as expected. Figure 4.7-d also reveals that, for both values of Re_w, the wave propagates slightly faster than what it would in the inviscid limit; specifically, about 0.1 percent faster (relative to the corresponding inviscid phase speed) for $Re_w = 1.6 \times 10^5$ and about 0.3 percent faster for the case $Re_w = 4 \times 10^4$ on average. According to the (inviscid) nonlinear wave theory, a wave of larger amplitude (i.e. the less-dissipated high Rew ISW) propagates faster. In this regard, our particular results are therefore opposite to our expectation. Possible reasons include the time discretization error, the interpolation error in the measurements and the numerical noise mentioned above. Nevertheless, considering the fact that the phase difference is maintained within a few percent of the wavelength after propagating for a distance

of ten wavelengths, we believe that the wave phase propagation is well-captured in the simulations considered here.

CHAPTER 5

POISSON SOLVER [1]

5.1 The pressure Poisson equation

5.1.1 Compatibility condition

In the KIO splitting scheme [74] presented on section 4.2.1, the PPE is obtained by taking the divergence of Eq. (4.9)

$$\nabla \cdot \frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} = \nabla \cdot \nabla p^{n+1}, \tag{5.1}$$

and imposing a divergence-free condition to the intermediate velocity $\hat{\hat{\mathbf{u}}}$

$$\nabla \cdot \hat{\hat{\mathbf{u}}} = 0.$$

A Poisson equation with Neumann boundary conditions therefore results:

$$\nabla^2 p = \nabla \cdot \left(-\frac{\hat{\mathbf{u}}}{\Delta t} \right) = f \quad \text{on} \quad \Omega, \tag{5.2}$$

$$\frac{\partial p}{\partial n} = \mathbf{n} \cdot \left[\sum_{q=0}^{J_e - 1} \beta_q \mathbf{N}(\mathbf{u^{n-q}}) + \nu \beta_q \mathbf{L}(\mathbf{u^{n-q}}) \right] = q \quad \text{on} \quad \Gamma. \tag{5.3}$$

The above expression for the Neumann boundary condition q is used in the KIO scheme to ensure consistency with the AB/BDEk time-discretization of the incompressible N-S equations [74].

The right hand side f and boundary operator q must satisfy a compatibility condition for the PPE (Eq. (5.2)-(5.3)) to have a solution. Specifically, the Poisson-Neumann problem

[1] The contents of this chapter are published on the article [33] *T he numerical solution of the pressure Poisson equation for the incompressible Navier-Stokes equations using a quadrilateral spectral multidomain penalty method* written by Jorge Escobar-Vargas, Peter Diamessis, and Charles Van Loan

is compatible (solvable) only if the volume integral (area integral in two dimensions) of the right hand side is equal to the net flux along the boundaries, i.e. the boundary integral of the boundary conditions. By integrating Eq. (5.2) over the whole domain we obtain

$$\int_{\Omega} \nabla^2 p \, d\Omega = \int_{\Omega} f \, d\Omega, \tag{5.4}$$

and by employing Gauss' theorem

$$\int_{\Omega} \nabla^2 p \, d\Omega = \int_{\Gamma} n \cdot \nabla p \, d\Gamma, \tag{5.5}$$

$$\int_{\Omega} f \, d\Omega = \int_{\Gamma} q \, d\Gamma. \tag{5.6}$$

Therefore, the Poisson-Neumann problem (5.2)-(5.3) has a solution only if (5.6) is satisfied [74, 101, 95, 58]. As already indicated in §4.2.1, in the original presentation of the KIO scheme, it is emphasized that the boundary integral of (5.5) is transformed by Gauss' theorem into a volume integral where the divergence of the second term in the original integrand vanishes. As a result,

$$\sum_{q=0}^{J_e-1} \beta_q \int_{\Omega} \nabla \cdot (\mathbf{N})^{n-q} \, d\Omega = \int_{\Omega} \nabla \cdot \left(\frac{\hat{\mathbf{u}}}{\Delta t} \right) d\Omega \tag{5.7}$$

must hold, which is indeed true through the AB/BDEk time-discretization, i.e. the compatibility condition is naturally satisfied.

5.1.2 Non-uniqueness of the pressure Poisson equation's solution

In addition to the compatibility condition, the Poisson equation does not have a unique solution because, by virtue of its boundary conditions, its solution is some function plus an additive constant. That is, given the Neumann boundary conditions

$$\mathbf{n} \cdot \nabla p = g \, \text{on} \, \Gamma, \tag{5.8}$$

any function of the form

$$p(\mathbf{x}) = g(\mathbf{x}) + h, \tag{5.9}$$

where h is an indeterminate additive constant, satisfies the boundary conditions (5.3) and is a solution to the PPE. Of course, in the spatially continuous (analytical) sense, once the pressure field has been obtained in the second fractional step of the KIO scheme, its determination up to an additive constant is a non-issue when computing $\hat{\mathbf{u}}$ through (4.9) since only the the gradient of the pressure field (∇p) is required. However, for the spatially discretized version of the KIO scheme, the non-uniqueness of solution to the Poisson-Neumann problem generates its own set of challenges as the corresponding linear system of equations is nearly-singular though consistent (provided the compatibility condition is satisfied). The above challenges, in a numerical framework, of compability condition satisfaction and the non-uniqueness of the solution of the PPE motivate a closer look at the SMPM discretization and its impact on the resulting Poisson matrix structure.

5.2 The penalty-based discrete pressure Poisson equation

5.2.1 Penalty formulation at subdomain interfaces

The spatial discretization of the penalized PPE for a point I_1 (see Fig 2.1) located at an interface (i.e. vertical or horizontal) takes the form [71]

$$\nabla^2 p^{(I_1)} + \tau Q(\mathbf{x}) \left(\alpha p^{(I_1)} + \beta \mathbf{n} \cdot \nabla p^{(I_1)} - g^{(I_2)}(\mathbf{x}) \right) = f^{(I_1)}, \tag{5.10}$$

where

$$g^{(I_2)}(\mathbf{x}) = \gamma p^{(I_2)} + \delta \mathbf{n} \cdot \nabla p^{(I_2)}. \tag{5.11}$$

In this case, the variables $\alpha, \beta, \gamma, \delta$ are constants of the penalty method, set to one in practice [66, 25], and $Q(\mathbf{x})$ is a Dirac delta function which ensures that the patching condition is applied only along the subdomain interfaces.

Expressions and limits for the penalty coefficients are derived based on determination of energy bounds in the evolution of the time-dependent linearized Burgers equation [66]. Following [25] the choice of penalty coefficients for the diffusion equation is found to perform robustly for the PPE. As a result, at the subdomain interfaces, the penalty coefficient must be chosen within the limits [66, 25, 65] (see section 4.2.2)

$$\tau = \frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa - 2\sqrt{\kappa^2 + \varepsilon\kappa}\right]\frac{2}{L_x^I} \leq \tau \leq \frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa + 2\sqrt{\kappa^2 + \varepsilon\kappa}\right]\frac{2}{L_x^I}, \qquad (5.12)$$

where $\omega = 2/(N(N-1))$ is a GLL quadrature weight, ε is the corresponding diffusion coefficient, set to one [71] and $\kappa = \omega\alpha/\beta$ [66, 71]. For a horizontal interface I_1, $\frac{2}{L_x^I}$ is a mapping coefficient and L_x^I the length of the subdomain. For a vertical interface, the subdomain height L_z^I is used instead. The degree of enforcement of the patching condition is set by the proximity of the penalty coefficient to the upper limit of Eq. (5.12). Our practical experience dictates that a choice of τ positioned closer to the lower limit works robustly for the problem of interest. Finally, there is no special formulation at the subdomain corners, which are treated as standard points along the vertical or horizontal interfaces.

5.2.2 Penalty treatment at physical boundaries

In a similar vein, given that the PPE under consideration is subject to Neumann boundary conditions, the penalty formulation for a point B located on a physical boundary (see Fig.

2.1) is given by

$$\nabla^2 p^B + \tau Q(\mathbf{x}) \left(\beta \mathbf{n} \cdot \nabla p^B - g^B \right) = f^B, \tag{5.13}$$

with

$$g^B = q^B, \tag{5.14}$$

where q^B is the prescribed value for the boundary condition at the boundary point *B*, given by Eq. (5.3), and the remaining variables are the same as for the interfacial case. The penalty term τ is now defined as [66]

$$\tau = \frac{1}{\beta \omega} \tag{5.15}$$

where ω is again the GLL weight at the collocation point *B*.

5.3 Properties of the discrete pressure Poisson equation

5.3.1 The discrete Poisson pressure equation

Once discretized, the pressure Poisson equation can be written as a linear system:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{5.16}$$

where the matrix **A** is the discrete analog of the penalized Laplacian and is constructed from the tensor product definitions given in Eq. (2.6)-(2.7) augmented with the contribution of the boundary/patching conditions at the boundaries/interfaces. Additionally, **x** is the solution vector (i.e. the pressure), and $\mathbf{b} = \nabla \cdot \left(-\frac{\hat{\mathbf{u}}}{\Delta t} \right)$ is the right-hand-side vector which contains information from the convective term and the Neumann boundary conditions (see Eq. (5.2) -(5.3)).

Fig. 5.1 shows the structure of the Poisson matrix **A**, for a 3 by 3 subdomain example with order of polynomial approximation $N = 8$.

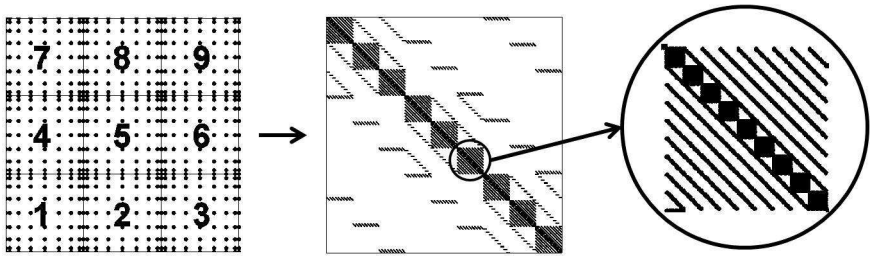


Figure 5.1: Left panel: SMPM Gauss-Lobatto-Legendre grid on the left (3×3 subdomains with $N = 8$). Central panel: Structure of the corresponding Poisson matrix **A**. Right panel: Structure of the contribution of each subdomain into the global matrix **A**.

As shown in the exploded view of the right panel of Fig. 5.1, the smaller-size blocks originate from the second derivative with respect to x (eqn (2.6)), whereas the remaining elements account for the second derivative with respect to z (eqn (2.7)). The additional entries within the matrix **A**, visible in the central panel of Fig. 5.1, correspond to the contribution of boundary and patching conditions. Most of these contributions are rank one matrices. In addition, and because of the intrinsic structure of the differentiation matrix at the subdomain level, the global matrix is non-symmetric.

5.3.2 Singular value distribution of the Poisson matrix

Due to the non-symmetric structure of the matrix, and its complex eigenvalues, its spectral properties are more effectively explored through a singular value analysis. The singular value decomposition (SVD) of the matrix **A** is given by

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}}, \tag{5.17}$$

where **U** and **V** are two orthogonal matrices that contain the left and right singular vectors, and Σ is a diagonal matrix with the singular values as its diagonal entries. Fig. 5.2 shows the singular value distribution for the example matrix shown in Fig. 5.1. From this figure, it is clear that the matrix is effectively singular by virtue of its $O(10^{-12})$ minimum singular value which forces the condition number of the matrix ($\kappa(\mathbf{A}) = \sigma_{max}/\sigma_{min}$) to be of $O(\approx 10^{17})$. Section 5.4 demonstrates explicitly the connection between the zero singular value and the non-uniqueness of the solution associated with the discrete Poissonn equation.

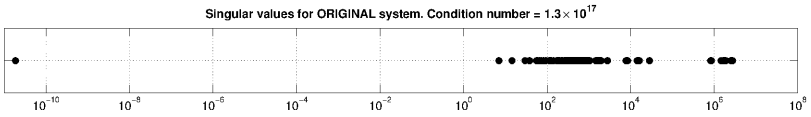


Figure 5.2: Singular value distribution of the Poisson matrix **A**: Case of 3×3 subdomains and $N = 8$

5.3.3 Compatibility condition revisited

The question arises whether Eq. (5.6) is the appropriate compatibility condition for the penalized form of the PPE ? Volume integration of the PPE would have to first be performed over each subdomain. In this case, an additional volume integral arises which effectively is reduced to the along-interface and along-boundary integral of the penalty terms in Eq. (5.10) and (5.13), respectively. There is no guarantee, primarily due to the inherent discontinuity of the SMPM at the subdomain interfaces, that the sum of these integrals across all subdomains will be zero, thereby allowing one to recover Eq. (5.6). One may argue that a modified compatibility condition, which takes into account the integral contribution of the penalty terms, might be more suitable, although the practical

utility of such a modified condition is unclear.

For all practical purposes, in the simulations considered here, the difference between the left and right hand side of Eq. (5.6) was found to be well above machine epsilon, the discrepancy becoming greater with increasing degree of under-resolution. Whether this discrepancy may be strictly attributed to the presence of the penalty terms or whether it is also compounded by GLL quadrature errors in an under-resolved set-up [54], remains unclear to us. Using the strategy originally proposed by Gottlieb and Streett [54], where the right hand side of the PPE is augmented by a constant equal to the difference of the right and left hand sides of (5.6) normalized by the area of the computational domain, did not produce a linear system of equations for which an iterative solver could converge. From a practical standpoint, this observation suggests that, rather than focus on satisfaction of the compatibility condition, it is more important to establish whether the resulting linear system of equations is indeed solvable, i.e. consistent. This issue is addressed in the next section.

5.3.4 Consistency of the linear system of equations

The system of equations (5.16) is consistent if

$$\mathbf{u_0^T A x} = \mathbf{u_0^T b} = 0, \tag{5.18}$$

where $\mathbf{u_0}$ is the left null singular vector of the matrix $\mathbf{A}$ [52]. Eq. (5.18) indicates that the PPE has a solution if the forcing vector $\mathbf{b}$ is orthogonal to the left null singular vector $\mathbf{u_0}$. In reference [101], this rationale is outlined for matrices obtained for low-order schemes and real eigenvalues, in the context of an eigendecomposition of the matrix $\mathbf{A}$ and its transpose $\mathbf{A}^T$.

In practice the condition (5.18) is usually not fulfilled, for reasons outlined in the previous section, and a regularization has to be applied to make the right hand side of (5.16) orthogonal to the left null singular vector $\mathbf{u_0}$ [101], i.e.

$$\mathbf{Ax} = (\mathbf{I} - \mathbf{u_0}\mathbf{u_0^T})\mathbf{b} = \hat{\mathbf{b}} \tag{5.19}$$

where $\hat{\mathbf{b}}$ is the orthogonal complement of $\mathbf{b}$ onto $\mathbf{u_0}$. Consistency, as represented by Eq. (5.18), is now ensured to machine epsilon since

$$\begin{aligned} \mathbf{u_0^T Ax} &= \mathbf{u_0^T}(\mathbf{b} - \mathbf{u_0}\mathbf{u_0^T}\mathbf{b}) \\ &= \mathbf{u_0^T b} - \mathbf{u_0^T u_0 u_0^T b} = 0 \end{aligned} \tag{5.20}$$

It is important to recall that if the PPE matrix is symmetric a standard eigendecomposition may be used where there is only one null eigenvector which is a constant vector [101]. In this case, the implementation of (5.19) is trivial. However, when the matrix is non-symmetric, as is the case with the SMPM, the left null singular vector $\mathbf{u_0}$ is no longer constant and has to be explicitly computed. For a large matrix, typical of environmental flow simulations with many degrees of freedom, the computational cost for a full singular value decomposition (SVD) is prohibitive. As availability of the left null singular vector is of vital importance for the efficient and robust solution of the SMPM-discretized pressure Poisson equation, an alternative procedure to obtain $\mathbf{u_0}$ is presented in section 5.5.

5.4 Null singular vector removal

The singularity of the Poisson matrix can pose a significant impediment to the iterative solution of the associated linear system of equations. In this section, we provide an

overview of strategies to remove the null singular vector, including either commonly used ones and also strategies developed specifically for the SMPM-discretized Poisson matrix. Note that the former are focused on removing the constant part of the solution, without necessarily considering a singular value decomposition (or eigendecomposition)of the matrix.

5.4.1 Commonly used strategies

Dirichlet boundary condition at a single point

This widely used technique consists of imposing a Dirichlet condition at one point along the physical boundaries [19]. As a result, the indeterminate additive constant responsible for a non-unique solution is now set equal to the value given by the Dirichlet condition. The null singular value is then shifted to the region where the remaining singular values are clustered and the matrix **A** is no longer singular. Although straightforward in its implementation, when used within the SMPM framework, this technique produces a particularly detrimental spurious effect. The insertion of a Dirichlet condition at a point on a boundary otherwise subject to Neumann conditions, produces a localized spike in the solution. In an incompressible Navier-Stokes simulation, this spike will grow in magnitude and pollute the solution in the interior of the computational domain. Note that this spurious effect is also observed when the Neumann boundary conditions are enforced strongly.

Furthermore, this technique modifies the tensor product structure of the global matrix **A**. As a result, the efficiency of any preconditioning technique at hand, which is based on the original structure of the matrix, is adversely impacted as the system solved is

no longer equivalent to the original one. Finally, use of a Dirichlet pressure boundary condition along an entire boundary of the computational domain might be dictated by the physics of the actual problem at hand, e.g. for an outflow boundary [79]. Such an approach obviously avoids any singularity issues of the Poisson matrix but is not always feasible since the pressure distribution along a physical boundary is not always known a priori.

Constant part removal

Taking into account that the solution of the system of equations can be determined up to an additive constant, an alternative approach to make the solution unique is by forcing its volume integral (i.e. its mean) to be zero [24]:

$$\int_{\Omega} p \, d\Omega = 0. \tag{5.21}$$

The discrete analog of Eq. (5.21) consists of adding one row with the Gauss-Legendre integration weights to the global matrix **A** and solving the overdetermined system of equations in a least squares sense. We did not pursue this option as it is unclear how one may obtain an efficient iterative solution of the resulting normal equations, with concerns of appropriate preconditioner design also being an issue.

In the same vein, the constraint (5.21) can be imposed in the form of a penalty term, i.e. by solving

$$\nabla^2 p + \tau \int_{\Omega} p \, dp = f \ , \tag{5.22}$$

which in matrix form becomes

$$\mathbf{A}\mathbf{x} + \tau \mathbf{1}\mathbf{w}^{\mathbf{T}}\mathbf{x} = \mathbf{b} \tag{5.23}$$

where τ is a penalty coefficient, $\mathbf{1}$ is a vector of all ones with size equal to the total number of degrees of freedom, as well as $\mathbf{w}$ that is a vector containing the Legendre weights for the numerical integration. For the matrix used in this work, the numerical results obtained with this techniques were not satisfactory since the new matrix ($\mathbf{A} + \mathbf{1w^T}$) is dense, which translates into a loss of the block structure, and an inefficient performance of the preconditioners customarily designed for the matrix $\mathbf{A}$.

Alternatively, one can appeal to the SVD of the Poisson matrix to remove the constant component of the PPE solution at the linear algebra level. Specifically, the solution can be rewritten as

$$\mathbf{x} = (\mathbf{U\Sigma V^T})^{-1}\hat{\mathbf{b}} \tag{5.24}$$

$$\mathbf{x} = \frac{\mathbf{u_0^T}\hat{\mathbf{b}}}{\sigma_0}\mathbf{v_0} + \sum_{i-1}^{N} \frac{\mathbf{u_i^T}\hat{\mathbf{b}}}{\sigma_i}\mathbf{v_i}, \tag{5.25}$$

where $\mathbf{u_i}, \mathbf{v_i}$ are the left and right singular vectors of the matrix $\mathbf{A}$, and σ_i are the corresponding singular values. Thus, in Eq. (5.25), the solution is written out in the form of an orthogonal expansion where the basis vectors are the right singular vectors $\mathbf{v_i}$, and the corresponding coefficients are $\mathbf{u_i^T}\hat{\mathbf{b}}/\sigma_i$. The right null vector $\mathbf{v_0}$ can readily be shown to have constant entries. Moreover, for a consistent singular system and exact arithmetic, the coefficient $\mathbf{u_0^T}\hat{\mathbf{b}}/\sigma_0$ is equal to zero divided by zero. Therefore, the first term in (5.25) corresponds to the constant part of the solution and is thus the discrete equivalent of the indeterminate additive constant of the analytical solution to the Poisson-Neumann problem in (5.9). In practice, the constant $\mathbf{u_0^T}\hat{\mathbf{b}}/\sigma_0$ is found to have a non-zero value which is bounded by machine epsilon at its lower limit, and round off errors at its upper limit.

Now, at each time step, the constant part of the solution may be removed by forcing the solution vector $\mathbf{x}$ to be orthogonal to the right null singular vector through

$$\hat{\mathbf{x}} = \mathbf{x} - \mathbf{v_0}\mathbf{v_0^T}\mathbf{x},$$

where $\mathbf{v}_0^T\mathbf{x}$ is the coefficient of the constant component in the orthogonal expansion of Eq. (5.25). The above regularization technique is similar to the one used to enforce consistency of the linear system of equations (see Eq. (5.19)). However, enforcing the orthogonality of the solution to the right null singular vector is effectively a post-processing action, i.e. it is implemented after the solution to the PPE has been iteratively computed and does not guarantee more efficient and robust performance of the iterative solution algorithm. For such a regularization to be implemented in the framework of the actual iterative solution algorithm, such as the conjugate gradient or GMRES methods, one would have to ensure that each new Krylov vector is orthogonal to the right null singular vector. For the conjugate gradient method, the iterative solver of choice for SEM [24], this strategy works well since each iteration gives an improved solution vector, and the final solution is thus orthogonal to the null vector (Paul Fischer, personal communication). When the above condition is imposed within the GMRES framework, the orthogonality among elements of the Krylov subspace is adversely impacted. Should a solution exist, the number of iterations to converge to it will then actually increase significantly. Consequently, more efficient avenues of ensuring a unique solution for the SMPM-discretized PPE are needed.

5.4.2 Strategies for the SMPM-discretized Poisson equation

Reduced system via Householder matrices

This approach is based on a combination of the SVD with Householder matrices [52]. The main goal is, by exploiting the properties of the associated orthogonal matrices, to reduce the $n \times n$ system of equations to an equivalent reduced one, with a null-space of zero dimension and a rank of $n - 1$. Effectively, the reduced matrix is such that it guides

the iterative solution method, GMRES in this case, to operate within a vector space that is orthogonal to the null space of **A**.

To describe the method, let us assume that we have the left and right null singular vectors $\mathbf{u}_0$ and $\mathbf{v}_0$ of the matrix **A**. For each one of these two vectors, an orthonormal basis **P** and **Q** can be built using Householder transformations,

$$\mathbf{P} = \mathbf{I} - 2\frac{\mathbf{h}_L\mathbf{h}_L^T}{\mathbf{h}_L^T\mathbf{h}_L} = [\mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_N}], \tag{5.26}$$

$$\mathbf{Q} = \mathbf{I} - 2\frac{\mathbf{h}_R\mathbf{h}_R^T}{\mathbf{h}_R^T\mathbf{h}_R} = [\mathbf{q_1}, \mathbf{q_2}, \cdots, \mathbf{q_N}], \tag{5.27}$$

where $\mathbf{h}_L$ and $\mathbf{h}_R$ are the left and right Householder vectors [52], and $\mathbf{p}_i, \mathbf{q}_i$, with $i = 1, \cdots, n$, are the column vectors of the matrices **P** and **Q** respectively. It is important to note that, in this construction, $\mathbf{p}_1 = \mathbf{u}_0$ and $\mathbf{q}_1 = \mathbf{v}_0$. Once the bases are built, the null vectors $\mathbf{u_0}, \mathbf{v_0}$ can be eliminated from the basis to obtain a reduced set of basis vectors $\mathbf{P}_r$ and $\mathbf{Q}_r$

$$\mathbf{P} = [\mathbf{u_0}, \mathbf{p_2}, \cdots, \mathbf{p_N}] \rightarrow \mathbf{P}_r = [\mathbf{p_2}, \cdots, \mathbf{p_N}], \tag{5.28}$$

$$\mathbf{Q} = [\mathbf{v_0}, \mathbf{q_2}, \cdots, \mathbf{q_N}] \rightarrow \mathbf{Q}_r = [\mathbf{q_2}, \cdots, \mathbf{q_N}]. \tag{5.29}$$

Following some algebraic manipulations, the reduced system of Eq. (5.30) is finally written as

$$\mathbf{P}_r^T\mathbf{A}\mathbf{Q}_r\mathbf{y} = \mathbf{P}_r^T\mathbf{b}, , \tag{5.30}$$

where $\mathbf{y} = \mathbf{Q}_r^T\mathbf{x}$. The SVD of the reduced matrix $\mathbf{P}_r^T\mathbf{A}\mathbf{Q}_r$ shows that its singular value distribution is very similar to that of the original matrix **A** but with the main difference that the reduced system is free of the null singular value, i.e. the reduced matrix is non-singular. An example of the distribution of singular values for the matrix of the reduced system corresponding to a Poisson-Neumann problem with 3×3 subdomains and $N = 8$ is shown in Fig. 5.3. The resulting modified singular value distribution is equivalent to

eliminating the term $\mathbf{u_0^T b}/\sigma_\mathbf{0}$ from Eq. (5.25), which translates into a unique solution for the system of equations and a significantly lower condition number for the new matrix $\mathbf{P}_r^T \mathbf{A} \mathbf{Q}_r$.

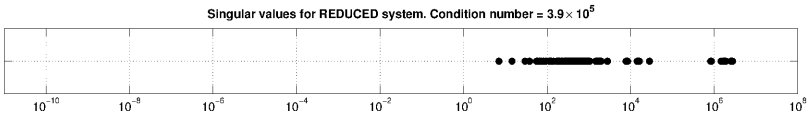


Figure 5.3: Singular value distribution of the matrix $\mathbf{U_R A V_R^T}$, where $\mathbf{A}$ is the matrix of Fig. 5.2, obtained with the reduced system technique via Householder matrices. Unlike Fig. 5.2, the null singular value σ_0 is now absent.

Given that $(\mathbf{Q}_r^T)^{-1} = \mathbf{Q}_r$, the final solution to the system of equations is computed as

$$\mathbf{x} = \mathbf{Q}_r \mathbf{y} \tag{5.31}$$

Note that none of the matrices used in this method are explicitly built and no direct matrix-matrix multiplications are involved. The final solution is constructed through a sequence of matrix-vector multiplications, which are implicit in the solution of a linear system of equations with a Krylov subspace method, such as GMRES.

Augmented system via bordered systems

An alternative approach is based on the concept of augmented (bordered) systems [57]. In this case, the augmented system of equations is expressed as

$$\begin{pmatrix} \mathbf{A} & \mathbf{d} \\ \mathbf{c}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \alpha \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ g \end{pmatrix} \tag{5.32}$$

where **c** and **d** are two vectors of dimension n that satisfy the following conditions

$$\mathbf{d}^T\mathbf{u}_0 \neq 0 \tag{5.33}$$

$$\mathbf{c}^T\mathbf{v}_0 \neq 0 \tag{5.34}$$

By expanding Eq. (5.32) we obtain

$$\mathbf{A}\mathbf{x} + \alpha\mathbf{d} = \mathbf{b} \tag{5.35}$$

$$\mathbf{c}^T\mathbf{x} = 0 \tag{5.36}$$

If Eq. (5.35) is multiplied by $\mathbf{u}_0^T$ the only way in which the system is consistent is for $\alpha = 0$

$$\mathbf{u}_0^T\mathbf{A}\mathbf{x} + \alpha\mathbf{u}_0^T\mathbf{d} = \mathbf{u}_0^T\mathbf{b} \tag{5.37}$$

on the other hand, by imposing $\mathbf{c} = \mathbf{v}_0$, uniqueness is ensured (see section 5.4.1), and the additive constant value is specified by g. The singular value distribution of the augmented matrix is shown in Fig. (5.4).

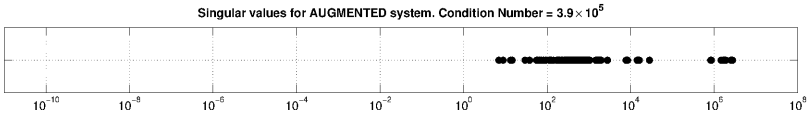


Figure 5.4: Singular value distribution for the augmented system corresponding to the matrix **A** of Fig. 5.2.

As in the case of the reduced system, the augmented system's matrix's singular value is nearly the same with that of original system, free, of course, of the null singular value. When this method is implemented in a Krylov framework (GMRES), within the matrix-vector multiplication, the vector **d** is not needed, since all its elements will be multiplied by the constant $\alpha = 0$.

Between the two methods presented here for the removal of the null singular vector,

we have found the iterative solution of the reduced system generated through Householder matrices to require slightly fewer iterations that the augmented one. Moreover, the number of iterations for the reduced system remains nearly fixed, whereas in the augmented system, that number fluctuates around the fixed value observed in the reduced system. As a result, in our actual simulations, we have elected to use the reduced system approach.

5.5 Computation of the left null singular vector

The efficient computation of the left null singular vector (LNSV) $\mathbf{u_0^{(2d)}}$ of the matrix **A**, denoted as $\mathbf{u_0}$, is one of the primary contributions of this work. Without the left null-singular vector, consistency of the Poisson pressure system of equations cannot be ensured (see section 5.3.4), and the techniques that remove the matrix singularity by reducing or augmenting the system of equations cannot be implemented (see section 5.4). Computing this null vector by performing the full SVD of the Poisson matrix is computationally costly and actually becomes prohibitive as the matrix **A** increases in dimension. Moreover, no analytical estimate of the left null singular vector has been reported in the literature. In this regard, it is worth noting that Weideman and Trefethen [124] show that the eigenvalues and eigenvectors of the second order pseudo-spectral differentiation matrix $\mathbf{D}_N^2$ cannot be obtained analytically. Such an observation suggests that the analytical estimation of the singular vectors of the matrix $\mathbf{D}_N^2$ and, therefore, of the full Poisson matrix (see sections 2.1 5.3.1) is also a highly challenging, if not impossible, task, which is outside of the scope of this work.

We instead resort to an alternative approach, whose main idea consists of using the Kronecker (tensor) product properties of the spectral multidomain methods to extend

concepts from one-dimensional domains to two-dimensional domains (see Chapter 2). This approach is validated by an experimental proof where the LNSV computed via Kronecker products is compared with the corresponding one computed with the MATLAB built-in function *svds*.

5.5.1 Doubly-periodic domain

The starting point for describing the particular LNSV computation procedure is a qualitative observation of the structure of the LNSV of the discrete Poisson matrix associated with a doubly-periodic domain. Fig. 5.5 shows an example of the LNSV structure for a domain with 3×3 subdomains, and $N = 4$. The observed LNSV structure and associated computation procedure outlined below, can be directly extended to any number of subdomains.

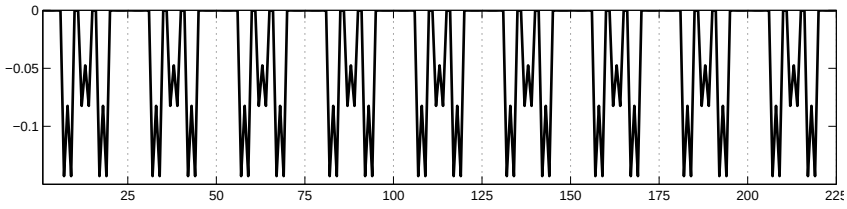


Figure 5.5: Left null singular vector $\mathbf{u_0}$ for an example of a doubly periodic domain with 3×3 subdomains, and $N = 4$. The vertical dashed lines separate the contributions of individual subdomains, i.e. the subvector **sv** shown in 5.6.

Inspection of Fig. 5.5 shows a repetitive pattern/subvector of total size of $(N + 1)^2$ elements, shown in detail in Fig. 5.6. This subvector is denoted as **sv**, and it is repeated as many times as the number of subdomains (*nsub*) in the global domain (e.g. *nsub* = 9

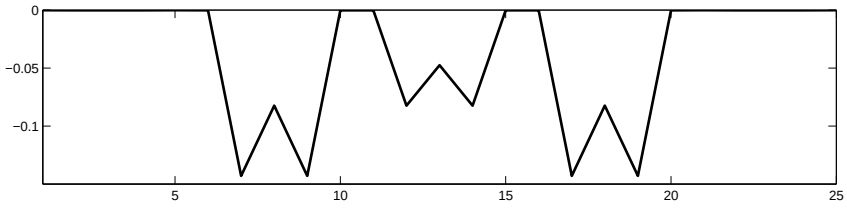


Figure 5.6: Exploded view of the subvector **sv** for an example of a doubly periodic domain with 3×3 subdomains, and $N = 4$.

for the case of 3×3 subdomains). A general definition of **sv** is

$$\mathbf{sv} = \mathbf{u_0}(p : p + (N+1)^2 - 1)$$

with $p = 1 + (j-1)(N+1)^2$, where $j = 1, \ldots, nsub$ represents the j-th subdomain. Based on this definition and our visual observations, we have found that, for the case of uniform-sized subdomains, we can construct the LNSV $\mathbf{u_0}$ as

$$\mathbf{u_0} = \frac{\mathbf{1}_{nsub} \otimes \mathbf{sv}}{||\mathbf{1}_{nsub} \otimes \mathbf{sv}||_2} \tag{5.38}$$

where $\mathbf{1}_{nsub}$ is a vector of ones with *nsub* elements. For the more general case of subdomains with different dimensions, observation indicates that the magnitude of the elements of **sv** scales with the area of the particular subdomain it originates from. For a doubly-periodic domain, with any number of arbitrarily-sized subdomains, the global LNSV is then generally computed as

$$\mathbf{u_0} = \frac{\mathbf{a} \otimes \mathbf{sv}}{||\mathbf{a} \otimes \mathbf{sv}||_2} \tag{5.39}$$

where **a** is a vector of *nsub* elements, which contains the area of each subdomain.

Further analysis applied to the vector **sv** (Fig.5.6) reveals an additional level of Kronecker product structure within it. As in Fig. 5.5, Fig. 5.6 also shows a repetitive pattern, denoted as $\mathbf{u_I}$ (see Fig. 5.7), which repeats itself every $N + 1$ entries of **sv** with varying magnitude. Specifically, the vector **sv** can be constructed as

$$\mathbf{sv} = -\mathbf{u_I} \otimes \mathbf{u_I} \tag{5.40}$$

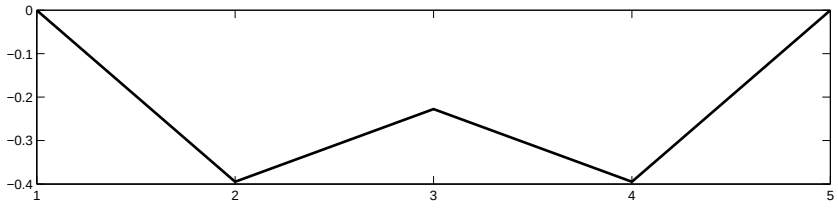


Figure 5.7: Structure of the vector $\mathbf{u_I}$ for the case of $N = 4$.

where $\mathbf{u_I}$ is a vector, regarded as a "core vector", representing the contribution of an appropriately defined subdomain one-dimensional subdomain to $\mathbf{u_0}$. Both a detailed definition and computation procedure of $\mathbf{u_I}$ are offered in section 5.5.3. Once the vector **sv** is computed, the global LNSV $\mathbf{u_0}$ is calculated using on Eq. (5.39) and (5.40) as

$$\mathbf{u_0} = -\frac{\mathbf{a} \otimes \mathbf{u_I} \otimes \mathbf{u_I}}{\|\mathbf{a} \otimes \mathbf{u_I} \otimes \mathbf{u_I}\|_2} \tag{5.41}$$

5.5.2 2D non-periodic domain

The same exercise can be performed for the more general non-periodic case. The structure of the LNSV associated with the discrete Poisson matrix, for an example of 3×3 subdomains and $N = 4$, is presented in Fig. 5.8. Effectively, the example subdomain consists of a central domain surrounded by eight subdomains, each of which has at least one physical boundary that is non-periodic. As in the previous case, the observed LNSV structure and associated computation procedure outlined below, can be directly extended to any number of subdomains.

In this non-periodic example, there are as many different types of patterns as there are subdomains with different combinations of boundary conditions along each interface (e.g. 9 for the case of 3×3 subdomains in the example set-up of Fig. 5.8). Nonetheless, there is not a clear repetitive pattern as in the doubly-periodic case, which means that no

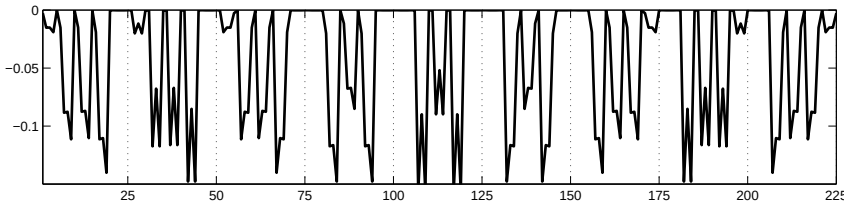


Figure 5.8: Structure of the of the left null singular vector $\mathbf{u_0}$ for an example non-periodic domain with 3×3 subdomains, and $N = 4$

longer a simple Kronecker product, as the one used in Eq. (5.39), can be used to compute $\mathbf{u_0}$. As in the periodic case, the magnitude of the entries in each subvector is related to the area of each subdomain. If we denote as $\mathbf{sv}^{(i)}$ the sub-vector of size $(N+1)^2$ that contains the contribution of the corresponding subdomain i, the global LNSV $\mathbf{u_0}$ can be computed as

$$\mathbf{u_0} = \frac{\left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right]^T}{||\left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right]||_2} \tag{5.42}$$

where the concatenation (|) operator is applied in a row-wise sense such that the numerator of Eq. (5.42) is a vector with the following structure (see Eq. (5.43))

$$\left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right]^T = \left[\underbrace{\mathbf{sv}_1^{(1)}\ldots\mathbf{sv}_{ns}^{(1)}}_{\mathbf{sv^{(1)}}}\underbrace{\mathbf{sv}_1^{(i)}\ldots\mathbf{sv}_{ns}^{(i)}}_{\mathbf{sv^{(i)}}}\underbrace{\mathbf{sv}_1^{(nsub)}\ldots\mathbf{sv}_{ns}^{(nsub)}}_{\mathbf{sv^{(nsub)}}}\right]^T . \tag{5.43}$$

In Eq. (5.42) and (5.43), *nsub* is the total number of subdomains, $ns = (N+1)^2$ is the total number of points per subdomain, and $\mathbf{sv}_j^{(i)}$ is the j-th element of the vector $\mathbf{sv}^{(i)}$, the non-periodic analog of vector $\mathbf{sv}$ defined in the previous section. Similarly to Eq. (5.40), the subvector $\mathbf{sv}^{(i)}$ corresponding to a subdomain i is computed as

$$\mathbf{sv}^{(i)} = -a_i(\mathbf{u_z} \otimes \mathbf{u_x}) \tag{5.44}$$

where a_i is the area of the subdomain i, and $\mathbf{u_z}$ and $\mathbf{u_x}$ are core vectors, non-periodic analogs of $\mathbf{u_I}$ used in the doubly-periodic case, which, however, are determined by the type of boundary the subdomain i has in the vertical and horizontal direction, respectively. The possible choices of $\mathbf{u_z}$ and $\mathbf{u_x}$ for the 9 different type of subdomains (in terms

of their patching/boundary conditions) of Fig. 5.9 are given in Table I. Specifically, these nine different subdomain locations where a subdomain *i* can be located are four corners (BL, BR, TL, and TR), four sides (B, T, L, and R), and the interior (I). Once the vectors $\mathbf{u_B}, \mathbf{u_T}, \mathbf{u_L}, \mathbf{u_R}$ and $\mathbf{u_I}$ are available, the null singular vector $\mathbf{u_0}$ can be computed for any 2D non-periodic domain discretized with rectangular subdomains. In the next section, a procedure to identify and compute these vectors is presented.

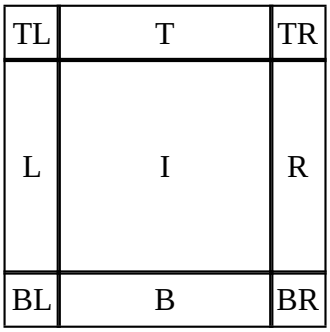


Figure 5.9: Schematic of a general non-periodic domain. It shows the possible locations of subdomain *i* subject to different patching/boundary conditions.

Location	$\mathbf{u_x}$	$\mathbf{u_z}$
BL	$\mathbf{u_L}$	$\mathbf{u_B}$
B	$\mathbf{u_I}$	$\mathbf{u_B}$
BT	$\mathbf{u_R}$	$\mathbf{u_B}$
L	$\mathbf{u_L}$	$\mathbf{u_I}$
I	$\mathbf{u_I}$	$\mathbf{u_I}$
R	$\mathbf{u_R}$	$\mathbf{u_I}$
TL	$\mathbf{u_L}$	$\mathbf{u_T}$
T	$\mathbf{u_I}$	$\mathbf{u_T}$
TR	$\mathbf{u_R}$	$\mathbf{u_T}$

Table I: Possible choices of the core vectors $\mathbf{u_x}$ and $\mathbf{u_z}$, used in the computation of $\mathbf{u_0}$, depending on the location of the subdomain under consideration, as shown in Fig. 5.9

5.5.3 Computation of the core vectors

We now focus on the estimation of the core vectors $\mathbf{u_B}, \mathbf{u_T}, \mathbf{u_L}, \mathbf{u_R}$ and $\mathbf{u_I}$. To this end, the starting point is the SMPM-discretized analog of the one-dimensional Laplacian defined over a finite horizontal non-periodic interval with three subdomains. Neumann boundary conditions are applied at the end-points of the full domain, and each subdomain has $N + 1$ collocation points.

For the sake of illustration, the procedure for computing the core vectors is now shown for the case of $N+1 = 5$ Gauss-Lobatto-Legendre points per subdomain. The same procedure applies for any number of points per subdomain. Fig. 5.10 shows a schematic of the global domain in which **L**, **I** and **R** indicate the left, internal and right subdomain within it.

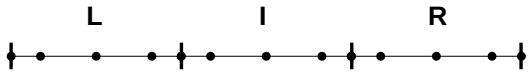


Figure 5.10: One dimensional base configuration for the generation of the left null singular vector $\mathbf{u_0}$ (case of $N + 1 = 5$).

The corresponding SMPM Laplacian matrix $\mathbf{A}_{1d}$ for the one-dimensional set-up shown in Fig. 5.10 has dimension $3(N + 1) \times 3(N + 1)$ (15×15 in the example) and can be decomposed according to the SVD as:

$$\mathbf{A_{1d}} = \mathbf{U_{1d}}\mathbf{\Sigma_{1d}}\mathbf{V_{1d}^T} \tag{5.45}$$

where $\mathbf{U_{1d}}$, $\mathbf{V_{1d}}$ and $\mathbf{\Sigma_{1d}}$ are defined in the same way as in (5.17). From such a decomposition, we isolate the null vectors $\mathbf{u_0^{(1d)}}$ and $\mathbf{v_0^{(1d)}}$. The right null vector $\mathbf{v_0^{(1d)}}$ is constant. However, the left null vector $\mathbf{u_0^{(1d)}}$ is not. The latter vector, of size $3(N + 1)$, can be

partitioned into three sub-vectors of size $N + 1$, with each sub-vector representing the contribution of each subdomain (i.e. L, I and R) to the global 1D null singular vector $\mathbf{u_0^{(1d)}}$ (see Eq. (5.46) and Fig. 5.11)

$$\left[\mathbf{u_0^{(1d)}}\right]^T = \left[\underbrace{u_0^{(1)} \dots u_0^{(N+1)}}_{\mathbf{u_L}} \underbrace{u_0^{(N+2)} \dots u_0^{(2N+2)}}_{\mathbf{u_I}} \underbrace{u_0^{(2N+3)} \dots u_0^{(3N+3)}}_{\mathbf{u_R}}\right]^T . \tag{5.46}$$

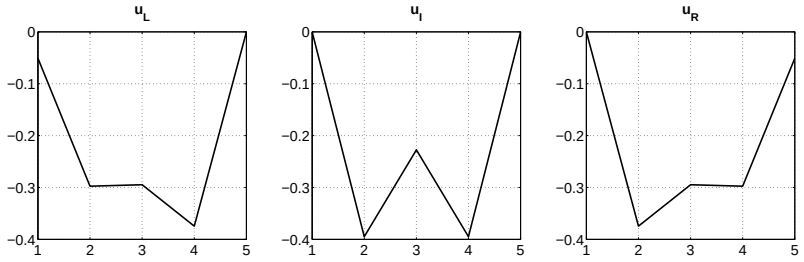


Figure 5.11: Left null singular vector structure of the one dimensional discrete SMPM Poisson matrix. $\mathbf{u_L}, \mathbf{u_I}$, and $\mathbf{u_R}$ for the case of $N + 1 = 5$

In Eq. (5.46) and Fig. 5.11, the vectors $\mathbf{u_L}, \mathbf{u_I}$, and $\mathbf{u_R}$ are the contributions of the left, central and right subdomains to the global null vector $\mathbf{u_0^{(1d)}}$. Note that if the same procedure is followed with the canonical 1D subdomains aligned with the vertical direction, the results are exactly the same as in the horizontal case with $\mathbf{u_B} = \mathbf{u_L}$ and $\mathbf{u_T} = \mathbf{u_R}$. With these considerations, for the case of doubly-periodic domains, the global LNSV is computed strictly through the vector $\mathbf{u_I}$ and Eq. (5.41), whereas for the calculation of the LNSV for the non-periodic case, the vectors $\mathbf{u_L}, \mathbf{u_R}$ and $\mathbf{u_I}$ are the ones needed (see Eq. (5.42)-(5.44)). If the domain has a combination of periodic and non-periodic boundary conditions, the corresponding choices of $\mathbf{u_L}, \mathbf{u_R}$ and $\mathbf{u_I}$ have to be take into account depending on the orientation of the periodic direction. Table II shows the choices of $\mathbf{u_x}$ and $\mathbf{u_z}$ for the different types of boundary conditions.

	Periodic		Non-Periodic		x-Periodic		z-Periodic	
Location	$\mathbf{u_x}$	$\mathbf{u_z}$	$\mathbf{u_x}$	$\mathbf{u_z}$	$\mathbf{u_x}$	$\mathbf{u_z}$	$\mathbf{u_x}$	$\mathbf{u_z}$
BL	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_L}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_L}$	$\mathbf{u_I}$
B	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_I}$	$\mathbf{u_I}$
BT	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_L}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_R}$	$\mathbf{u_I}$
L	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_I}$
I	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$
R	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_I}$
TL	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_L}$	$\mathbf{u_R}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_L}$	$\mathbf{u_I}$
T	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_I}$	$\mathbf{u_I}$
TR	$\mathbf{u_I}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_R}$	$\mathbf{u_I}$	$\mathbf{u_R}$	$\mathbf{u_R}$	$\mathbf{u_I}$

Table II: Possible choices of the core vectors $\mathbf{u_x}$ and $\mathbf{u_z}$ used in the computation of $\mathbf{u_0}$ for different choices of boundary conditions, recalled that $\mathbf{u_B} = \mathbf{u_L}$ and $\mathbf{u_T} = \mathbf{u_R}$

5.5.4 Validation of the procedure

For the case of the 2D doubly-periodic domain, Fig. 5.12 shows the L_2 norm of the difference between the LNSV computed for the full matrix via the MATLAB function *svds*, and the LNSV computed with the procedure outlined above. The error analysis is performed for different number of subdomains and polynomial degrees. The results show that the error is of the order of $O(< 10^{-12})$, and is independent of the number of subdomains.

The same error analysis is done for the non-periodic case, and presented in Fig. 5.13. As in the periodic case, the results show that the error is of the order of $O(< 10^{-12})$, and is independent of the number of subdomains.

The procedure outlined above is a much more efficient means to obtain the LNSV of the pressure Poisson matrix than the prohibitively costly, for realistic problems, full SVD. All that is needed a priori are the vectors $\mathbf{u_L}, \mathbf{u_I}$, and $\mathbf{u_R}$. Moreover, as shown in Fig.s

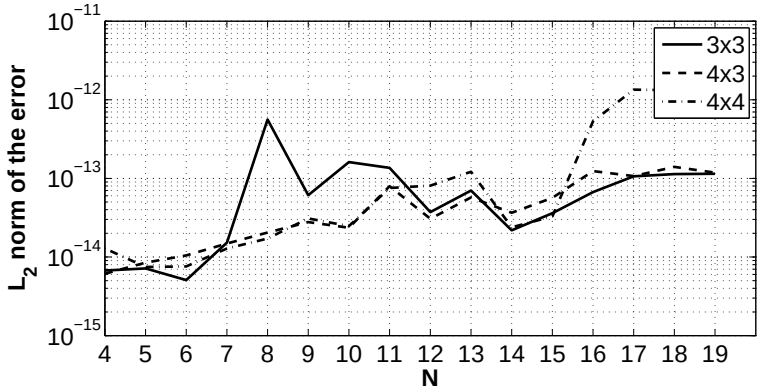


Figure 5.12: L_2 error norm (as compared to the corresponding MATLAB estimate) in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a doubly-periodic domain.

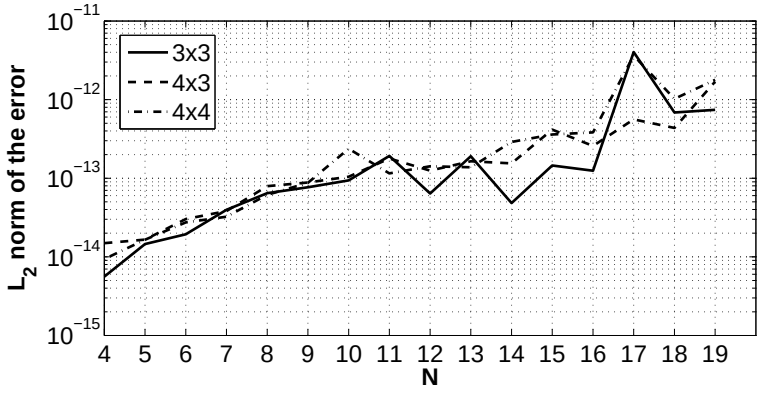


Figure 5.13: L_2 error norm in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a 2D non-periodic domain.

5.12 and 5.13 it was established that the degree of approximation of the two-dimensional LNSV obtained via tensor products of the contributions of the one-dimensional LNSV is highly accurate. A rigorous proof of Eq. (5.43) and (5.44) still remains to be offered. Such a proof is deferred to future studies.

Finally, the use of Kronecker products to compute the left null singular vector suggest a connection with the Kronecker product structure of the discrete Laplacian given by the

sum of Eq. (2.6) and (2.7). Nonetheless, the exact nature of this connection remains to be established and is subject of future work.

5.6 Preconditioners

Although we have established a framework that guarantees that a solution for SMPM-discretized PPE exists and is unique, we still need to ensure that the iterative scheme used towards this computing solution is computationally efficient. To this end, it is imperative that an efficient preconditioner matrix (**M**) be developed. In this study, the design of a preconditioner has been found to be highly sensitive to the type of boundary conditions applied to it, which cannot be different than the Neumann conditions applied to the original system. Furthermore, consistency of the preconditioned system of equations must be preserved to obtain a physically meaningful solution. In what follows, we outline the basic components of three preconditioning strategies, classical diagonal Jacobi, block-Jacobi and a two-level preconditioner, designed with the particular characteristics of the SMPM discretization in mind.

5.6.1 Diagonal Jacobi preconditioner

This classic and straightforward strategy uses as a preconditioner a diagonal matrix that consists of the diagonal elements of the global matrix [106].

$$m_{i,i} = a_{i,i} \tag{5.47}$$

As a first approximation for **M**, we find that the diagonal Jacobi preconditioner works well in the simulation of viscously-driven flows, such as the Taylor vortex, but when

applied to a flow with strongly nonlinear characteristics (e.g. the lid driven cavity), it is highly inefficient, often with the iterative solution never converging.

5.6.2 Block Jacobi (non-overlapping additive Schwarz)

The element-based character of the SMPM [65] furnishes a natural domain decomposition, which is reflected in the block structure of the Poisson matrix **A** (see section 5.3.1 and Fig. 5.1), where there is a direct one-to-one association between each of the large blocks with a particular subdomain. The block-Jacobi method thus uses the contribution of each subdomain to the global Poisson matrix to form the individual blocks of a preconditioner. Each block contains the SMPM-discretized analog of the Laplacian combined with the contributions of discretized Robin-type boundary conditions at the subdomain interfaces and Neumann conditions at the physical boundaries (Eq. (5.10) and (5.13), respectively). Under these considerations, we can construct a non-overlapping Schwarz preconditioner as

$$\mathbf{M}^{-1} = \sum_{k=1}^{nsub} \mathbf{R}_k^T \mathbf{A}_k^{-1} \mathbf{R}_k \tag{5.48}$$

where *nsub* is the number of subdomains, and $\mathbf{R}_k$ is a restriction/prolongator operator that transfer data from the local to the global problem and vice-versa [42]. Due to the type of boundary conditions applied to each subdomain, the local stiffness matrix $\mathbf{A}_k$ is non-singular and, in the preconditioner setting, it can be inverted directly via an LU decomposition.

Numerical results (see section 5.7 for more detail) show that this preconditioner reduces the number of iterations with respect to the absence of a preconditioner or using only diagonal Jacobi. The number of iterations within the GMRES solver are indepen-

dent of the degree of approximation, i.e., for a small number of subdomains, the block-Jacobi preconditioner deals efficiently with *p*-refinement. However, when *h*-refinement is applied, corresponding to a sizable increase in number of subdomains and degrees of freedom, the number of iterations of the GMRES and computational time of the solver increases linearly (see Fig. 5.15). Thus, this preconditioner is unsuitable for large problems, such as those encountered in environmental fluid mechanics applications. For such problems, a more efficient preconditioning strategy is needed.

5.6.3 Two-Level preconditioner

The implementation of this preconditioner draws from the previous work of Fischer and collaborators [38, 42] and the need for $h-$scalability. It combines the above block-Jacobi method as a preconditioner at the fine-level with a coarse-grid component based on a low-order $N = 1$ SMPM approximation of the Poisson-Neumann problem. The general form of this preconditioner is

$$\mathbf{M}^{-1} = \mathbf{R}_0^T \mathbf{A}_0^{-1} \mathbf{R}_0 + \sum_{k=1}^{nsub} \mathbf{R}_k^T \mathbf{A}_k^{-1} \mathbf{R}_k \tag{5.49}$$

Eq. (5.49) is effectively Eq. (5.48) augmented by the additional term $\mathbf{R}_0^T \mathbf{A}_0^{-1} \mathbf{R}_0$ that accounts for the coarse grid correction. $\mathbf{R}_0$ is an interpolation matrix [7] that projects a scalar field across different Gauss-Lobatto-Legendre grids and $\mathbf{A}_0$ represents the low-order (coarse-level) analog of the Poisson matrix.

As mentioned in the previous section, the solution of the fine level preconditioner (Block-Jacobi / Additive Schwarz) does not suffer from the problems of a nearly singular system due to the Robin type boundary conditions applied to the subdomain interfaces, which make each one of the blocks non-singular. This is not the case for the

coarse grid preconditioner, where the same problems associated with the global Poisson matrix once again must be addressed. In this regard, a regularization along the lines of Eq. (5.19) has to be applied to the coarse-system solver in order to make it consistent, otherwise the preconditioner cannot be solved for. As with the block-Jacobi preconditioner, the solution of the coarse grid preconditioner is performed with a direct LU solver. In section section 5.7, the scalability of the two-level preconditioner is compared to that of the additive Schwarz (Block-Jacobi), and diagonal Jacobi.

5.7 Numerical results

5.7.1 Taylor vortex

This is the first of two test cases used to assess the performance of the previously outlined iterative solution strategies for the Poisson-Neumann problem within the framework of an incompressible Navier-Stokes equation solver. Details on the analytical expression for this case, and for the convergence properties od the solution can be found in section 4.3.1.

The impact of the above discussed preconditioners on the efficiency of the numerical solution of the PPE is shown in Fig. 5.14 by examining the average number of pressure iterations per time step and average CPU time per time step as a function of the total number of DOF to account for both *h* and *p*-refinement. The left panel in Fig. 5.14 indicates a visible reduction in iteration count when the block Jacobi or the two-level preconditioner is used in place of diagonal Jacobi. For this test case, there is minimal difference between the performance of the block-Jacobi and two-level preconditioners,

with the former requiring a slightly smaller degree of iterations and slightly less CPU time. One might conjecture that the minimally better performance of the block-Jacobi preconditioner might be linked to the smoothness of the highly viscous solution. From the right plot of Fig. 5.14 a similar conclusion can be obtained, i.e. the CPU time per timestep for the GMRES solver to converge with a particular preconditioner reflects the number of iterations per timestep.

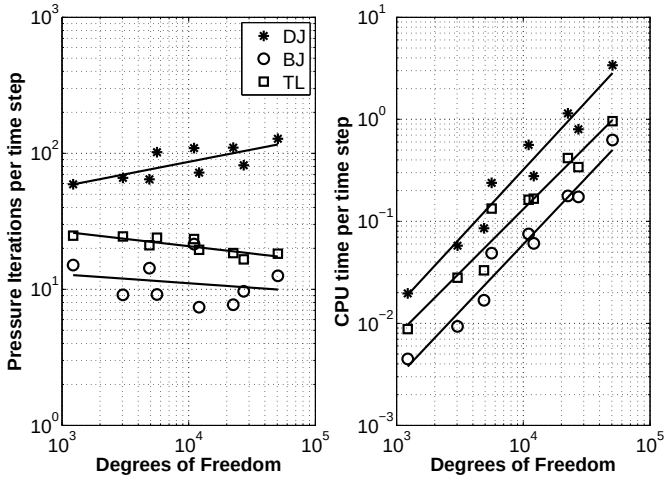


Figure 5.14: Poisson solver performance for different preconditioners for the Taylor vortex problem. Left panel shows the average number of pressure iterations per timestep as a function of total number of degrees of freedom (DOF). Right panel shows the average CPU time per timestep as a function of total number of degrees of freedom (DOF). In the legend, DJ, BJ and TL correspond to diagonal Jacobi, block-Jacobi and two-level preconditioners, respectively.

5.7.2 Lid-driven cavity flow

The lid-driven cavity test case presented on section 4.3.2 was also used to check the performance of the preconditioners for the solution of the PPE in the context of the in-

compressible Navier-Stokes solver. Fig. 5.15 shows the performance of the pressure solver in terms of number of iterations and computational time per timestep for both the block Jacobi (BJ) and two level (TL) preconditioners. Note that no results are shown for the diagonal Jacobi preconditioner as its application did not allow the iterative solver (GMRES) to converge. The left panel of Fig. 5.15 shows the iteration count as a function of polynomial degree N and number of subdomains in each direction for the BJ (white surface) and TL (gray surface) preconditioners. The BJ preconditioner successfully deals with the demands of $p-$refinement by fixing the average number of iterations to a constant when increasing N for a fixed number of subdomains. However, for a given N, once the number of subdomains increases , the iteration count also increases, which indicates that the BJ preconditioner is ineffective in accomodating $h-$refinement. This shortcoming is addressed through the incorporation of a coarse-grid component through a TL preconditioner, as is visible in the same figure. As indicated by the grey surface, use of the TL preconditioner with an increasing number of subdomains keeps the iteration count nearly fixed and well below 100.

The right plot in Fig. 5.15 shows the CPU time per timestep for the two preconditioners as a function of the total number of DOF, for the same cases presented in the left figure. A power law best fit is also shown to enable extrapolation of the performance for both preconditioners for problems with a large number of DOF. For problems with less than 10^4 DOF, where the total number of subdomains is of $O(100)$ or less, the BJ preconditioner is faster. As the total number of DOF increases, typically a result of h-refinement, the TL preconditioner shows a visible gain in speed. Environmental flow simulations, such as those discussed in section 1, typically require $O(10^6)$ DOF in two dimensions which suggests, according to the power law fits, that, on a single processor, the TL preconditioner will be 4 times faster than the BJ preconditioner. In this

regard, when applied to a number of other non-periodic benchmarks, either stratified or non-stratified (e.g., temporally evolving shear layer, stratified lock exchange and propagating fully nonlinear internal solitary wave), considered in greater detail in a separate manuscript in preparation, the performance of the three preconditioners here has been found to be similar to that reported here for the lid-driven cavity.

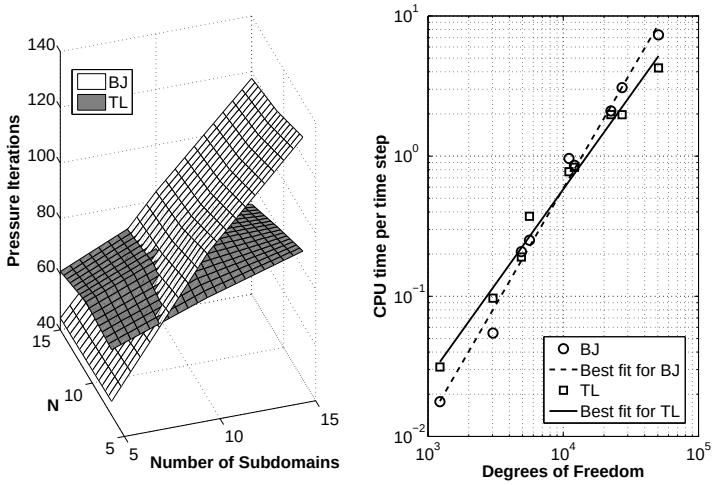


Figure 5.15: Poisson solver performance for the Block Jacobi (BJ) and Two-Level (TL) preconditioners applied to the Lid-driven Cavity flow. Left Panel: pressure iterations as a function of number of subdomains and polynomial degree N (White surface: BJ preconditioner. Grey surface: TL preconditioner). Right panel: CPU time per time step as a function of degrees of freedom (DOF) on the right. Also shown are least-squares power law best-fits.

5.8 Discussion

Various preconditioners previously developed for other high-order element based methods have been applied to our SMPM-discretized PPE. However, the efficient performance of such pre-existing preconditioners has been found to be impeded by the dis-

continuous formulation of SMPM at the subdomain-interfaces, the requirement of Neumann boundary conditions and the non-symmetry of the global Poisson matrix. First, the incomplete LU (ILU) preconditioner [106] was examined, which was found to be impractical for large problems as matrix storage is required. A subsequent step involved a preconditioner based on the finite difference (FD) discretization of the Laplacian operator [24]. In this case, applying the FD discretization at the discontinuous interfaces of the SMPM grid is not a straightforward procedure. As a result, solving the FD preconditioner matrix is a costly task, since the resulting matrix is non-symmetric and nearly singular.

A $p-$multigrid preconditioner has also been tested [105, 104, 37, 111] in order to take advantage of the hierarchy inherent in the Legendre polynomial basis functions used in the SMPM and the fast computation of GLL points and differentiation matrices. The main problem encountered in this approach is the inefficiency of the smoothing steps which require a significant number of iterations (as high as 50) to remove the high frequency oscillations that contaminate the coarser grid solves encountered at subsequent levels of the multigrid cycle. Finally, a projection technique relying on multiple right hand sides of the PPE, obtained from previous timesteps, [39] was also tested in the framework of a TL-preconditioned GMRES iterative solver, with the puprose of further reducing the total number of iterations. A modified Gram-Schmidt orthogonalization was needed instead of the classic Gram-Schmidt for the stable generation of the successive right-hand-sides. Unfortunately, unlike what was observed in its application to a conjugate gradient solver used within a SEM framework [39], when applied to the iterative solution of the SMPM-discretized PPE, this technique did not reveal any decrease in iteration count for the GMRES solver.

As a concluding note to this discussion, the coarse-level preconditioner is constructed using a low-order ($N = 1$) SMPM discretization of the Laplacian operator. Such a small value of N is chosen to allow for a direct solver (LU factorization) to be used for the resulting linear system of equations when computing the preconditioner. An increase to $N = 2$ or 3 polynomial could make this LU decomposition computationally infeasible when the number of subdomains is large, as is the case of an environmental flow simulation.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Shallow Water equations

The performance and properties of two commonly used high-order-accuracy element-based spatial discretization methods, spectral multidomain penalty (SMPM) and discontinuous Galerkin (DGM), are examined in the framework of the inviscid shallow water equations (SWE). Whereas a previous comparison study [55] focused on one-dimensional conservation laws and considered a modally-based Galerkin formulation of SMPM and DGM, this paper applies both techniques to a system of nonlinear conservation equations and considers them in the more frequently used nodal form, in a collocation and Galerkin formulation, for SMPM and DGM, respectively. The two methods are applied to a suite of test cases that are of interest in oceanic shallow water flow: three linear (standing wave, Kelvin wave and linear Stommel problem) and three non-linear (Rossby soliton, nonlinear Stommel problem and Riemann problem). The analysis shows that the methods can be simplified to be the same method when specific choices of the penalty terms (for the SMPM) and numerical flux (for the DGM) and when the same collocation points are chosen for representing the discrete solution. The numerical solutions showed that the methods are extremely similar not only in achieving the same rate of convergence but also in their conservation of energy measures. The key difference between the SMPM and DGM is in their choice of penalty terms that enforce weak boundary conditions across element interfaces. The SMPM has much flexibility in selecting these terms whereas the DGM method is more rigid in its choices in the sense that a Riemann solver must be used; however, this idea offers much flexibility in handling a large variety of flows including those requiring wetting and drying algorithms,

for example. Both methods can be used on fully-unstructured quadrilateral element grids but it is not clear how to extend the SMPM to unstructured triangular elements; in contrast, the formulation of the DGM is quite natural and can be extended to triangles rather straightforwardly, assuming that a good set of interpolation and integration points is known (see, e.g., [46, 51, 86, 68]). The SMPM proved to be slightly more efficient than the DGM, in terms of computational time, and we expect this trend to continue as Laplacian operators (as required by Navier-Stokes or even by more realistic shallow water ocean modeling simulations) are introduced.

6.2 Incompressible Navier Stokes solver

A quadrilateral spectral multidomain penalty method (SMPM) solver has been developed for the numerical solution of the two-dimensional incompressible Navier-Stokes equations under the Boussinesq approximation for the purpose of studying environmental stratified flow processes at high Reynolds numbers. Through the use of variable-size two-dimensional collocation approach in each subdomain, arbitrary boundary conditions and localized resolution can be employed in both spatial directions. A high-accuracy semi-implicit splitting scheme is used, based on a third order stiffly stable scheme for the non-linear term approximation, third order backward differentiation for the temporal derivatives and a high-order numerical boundary condition for the pressure. High spatial accuracy in space is enabled through the use of a local two-dimensional Legendre discretization in each subdomain.

Environmental stratified flow processes, such as turbulence and internal solitary waves (ISWs), are characterized by such values of Reynolds numbers that the associated numerical simulations are inherently under-resolved. To preserve the numerical stability

of the spectral quadrilateral multidomain scheme, while preserving spectral accuracy, in an under-resolved simulation, a penalty technique is implemented at the physical boundaries and subdomain interfaces. The particulars of the penalty formulation vary in each fractional step of the high-order splitting scheme, i.e. explicit non-linear term advancement, pressure Poisson equation and implicit viscous term treatment. Additional stability at the interfaces is provided through adaptive interfacial averaging. In the subdomain interior, numerical stability is ensured through dealiasing the nonlinear term calculation and application of spectral filtering after each fractional step.

Although similar to a previously developed SMPM solver for high Reynolds incompressible stratified flows in domains with one non-periodic direction [25], the solver here has one fundamental difference, the requirement of efficient numerical solution of the SMPM-discretized pressure Poisson equation (PPE), a non-trivial and challenging task. Section 6.3 offers concluding remarks on this issue.

The flow solver's accuracy and robustness were validated against a standard set of incompressible flow benchmarks, namely the Taylor vortex, lid-driven and double shear layer. From an environmental stratified flow process, the canonical problem of ISW propagating at a high Reynolds numbers in a two-layer continuously stratified free-slip horizontal channel was examined. The fully nonlinear ISW, an exact solution to the incompressible Euler equations, propagates at the theoretically prescribed phase speed while maintaining its original wavelength and amplitude, indicating negligible numerical dissipation and dispersion.

The availability of the quadrilateral SMPM solver enables the investigation of a much broader range of environmental stratified flow processes than those attainable with the

equivalent solver that supported only one non-periodic direction. Resolution can now be localized in the streamwise direction, not only the vertical, while arbitrary boundary conditions may be prescribed at the lateral boundaries. A third, periodic, transverse direction may be readily incorporated into the quadrilateral SMPM solver by using a Fourier discretization. An example of an environmental flow process amenable to simulation with a three-dimensional quadrilateral SMPM solver, is the separating turbulent bottom boundary layer under under an ISW wave in uniform depth water [26, 17, 2]. In this case, the turbulence, homogeneous in the transverse, is confined to a small-window near the bed, extending from the trough of the wave towards its rear end. A computational domain spanning the above window avoids the unnecessary cost of resolving the full water column depth and length of the ISW, the large bulk of which is inactive over significant distances of wave propagation.

6.3 Poisson Pressure equation solver

An efficient iterative solution strategy has been developed for the quadrilateral spectral multidomain penalty method (SMPM)-discretized pressure Poisson equation (PPE) with Neumann boundary conditions, implicit in the time-discretization of the two-dimensional incompressible Navier-Stokes equations through a high-order splitting scheme. From the spatially continuous perspective, this system of equations has a solution only if an integral compatibility condition involving the right-hand-side of the PPE and the prescribed value of the Neumann boundary conditions is fulfilled. However, although the compatibility condition is automatically satisfied at the spatially continuous (analytical) level in the context of the above splitting scheme, it is unclear whether it is is the appropriate compability condition for the the SMPM-discretized PPE. Our observations further indicate that, in actual incompressible flow simulations, the resuling linear

system of equations never satisfy the equivalent solvability condition of orthogonality between the right hand side and the null left singular vector of the Poisson matrix. This lack of solvability may be attributed to the discontinuity of the pressure solution across subdomains and to inexact quadrature, the latter a feature of under-resolved simulations. Finally, the particular boundary conditions give rise to a non-unique solution and, therefore, a near-singular Poisson matrix.

For the resulting linear system of equations, satisfaction of the above solvability condition, i.e. consistency of the linear system of equations, is ensured through the regularization that projects the right-hand-side onto the plane orthogonal to the left null singular vector of the global Poisson matrix. Uniqueness of the solution is ensured at the linear algebra level by reducing the system of equations via Householder matrices or via an augmented matrix technique.

A key contribution of this work is the development of a computationally efficient technique to estimate the left null singular vector of the SMPM-discretized Poisson matrix, which avoids the prohibitively costly SVD of the matrix. The two-dimensional left null singular vector is constructed from its one-dimensional equivalent which is computed for a canonical one-dimensional SMPM Poisson matrix defined over three subdomains. Availability of the left null singular vector then enables the above described strategies for ensuring a consistent linear system of equations and a unique solution for the PPE.

Even if a consistent linear system of equations and a unique solution are ensured, the efficient iterative solution of the SMPM-discretized PPE cannot be obtained without an appropriately designed preconditioner. To this end, two preconditioners, a block Jacobi (BJ), and a two-level preconditioner (TL), have been implemented. The performance of

both preconditioners has been assessed through application to two well-known benchmark problems for the numerical solution of the incompressible Navier-Stokes equations: the Taylor vortex and the Lid-driven cavity. The BJ preconditioner is found to prevent the increase in iteration count with increasing $p-$refinement. However, it cannot provide for an efficient solution at high levels of $h-$refinement, i.e. an increasing number of subdomains. For this purpose, a TL preconditioner, a combination of coarse-grid and fine-level approaches has been constructed. Its fine-level component is identical to the standard BJ preconditioner described above. The coarse-level component of the TL preconditioner is based on a low-order SMPM discretization and resolves the issue with high-levels of $h-$refinement. In analogy with the SMPM-discretized Poisson matrix, the coarse-level component of the TL preconditioner requires a similar regularization which ensures that the associated linear system is consistent.

Beyond providing a framework solution of the PPE system of equations, this work has intended to provide a concentrated overview of the techniques used by the higher-order method community in the context of the Poisson-Neumann problem for the pressure field implicit in the numerical solution for the incompressible Navier-Stokes equations. In a similar vein, we hope that the techniques developed here, namely the construction of the left null singular vector and its application to ensuring consistency and a unique solution of the linear system of equations, will be of interest to the sub-discipline of numerical linear algebra focused on the iterative solution of consistent singular non-symmetric systems.

6.4 Future work

From the point of view discontinuous high-order element based methods, future work will be addressed on observing further differences between the SMPM and DGM, test cases with complex geometries, non-smooth solutions or additional forcing terms have to be executed with the methods. The performance of each method has to be also assessed for different time advancement schemes , as well as different types of numerical fluxes to account for the communication between subdomains. Additionally, parabolic and elliptic partial differential equations have to be assessed in the context of compressible and incompressible flows, where more challenging numerical difficulties appear for the implementation of both methods.

In terms of the incompressible Navier-Stokes solver future work will be focused on the implementation of deformable subdomains. Additional efforts will concentrate on the improvement of preconditioners for the iterative solution of the PPE. To this end, ISW propagation and ISW-seafloor interaction is typically simulated in highly anisotropic domains with high aspect ratio subdomains, which can detrimentally affect the efficiency iterative solution scheme for the PPE. Recently developed techniques for improved efficiency of the numerical solution of highly anisotropic elliptic equations [107] carry great potential towards effectively addressing this issue. Finally, the ultimate goal of this effort is to develop a three-dimensional hexahedral subdomain SMPM solver.

Specifically for the numerical solution of the PPE, future work can be oriented towards a detailed comparison of the spectral properties of the Poisson-Neumann matrix for different spatial discretizations and constructing a unified framework for the solution of the nearly-singular systems that arise in the numerical solution of the incompressible

Navier-Stokes equations. In addition, the formulation of the Poisson-Neumann problem within the context of projection techniques can be improved in order to ensure consistency of the pressure linear system of equations directly from the formulation, instead of the regularization technique presented on section 5.3.4. More efficient preconditioning efforts could focus on exploiting the Kronecker product structure of the Poisson matrix, or alternatively, translate to the SMPM the experience gained with algebraic multigrid for continuous and discontinuous finite element type methods [94]. Finally, additional considerations will arise in the computation (as outlined here) of the left null singular vector for the Poisson matrix resulting from a SMPM discretization of a domain with deformed, non-square, subdomains.

The degree of strong enforcement of solution continuity, i.e. patching condition enforcement, at the subdomain interfaces for the Poisson and modified Helmholtz equations is set by the choice of penalty coefficient value, as computed in Eq. (4.20). The choice of the particular penalty coefficient has been found to play a critical role both the numerical stability of the solution the solver and the efficiency of associated iterative implicit solvers (in terms of number of iterations and CPU time). When the coefficient is chosen near the upper limit of Eq. (4.20), continuity across subdomains is enforced strongly yet there is a greater susceptibility towards oscillations at the subdomain interfaces and the number of iterations in the implicit solvers can grow considerably. However, when the coefficient is near the lower limit, continuity across subdomains can become excessively weak, leading to jumps at the interfaces which can either grow catastrophically or produce spurious flow structure. In conclusion, work has to be done in order to establish a procedure to compute the magnitude of the penalty coefficient as a function of Reynolds number, and degree of uniformity of the mesh.

In the same vein, in regards to the high order time-splitting scheme used in this work [74], the imposition of penalized inflow/outflow velocity boundary conditions for the viscous fractional step in the context of Neumann or mixed Dirichlet-Neumann type is non-trivial [15]. Similar issues arise in the treatment of the boundary values at the physical boundaries during the update step (Eq. (4.9)), after the PPE has been solved. Additional work has to be addressed in this regard in order to be able to expand the range of possibilities for which the incompressible Navier-Stokes solver can be used for.

BIBLIOGRAPHY

[1] A. M. Abdilghanie and P. J. Diamessis. On the generation and evolution of numerically simulated large-amplitude internal gravity wave packets. *Theor. Comput. Fluid Dyn.*, 26(1-4):205–224, 2012.

[2] P. Aghsaee, L. Boegman, P. J. Diamessis, and K. G. Lamb. Boundary-layer-separation-driven vortex shedding beneath internal solitary waves of depression. *J. Fluid Mech.*, 690:321–344, 2012.

[3] D. Alevras. Simulation of the indian ocean tsunami with real bathymetry using a high-order triangular discontinuous Galerkin shallow water model. Master's thesis, Naval Postgraduate School, 2009.

[4] J. D. Anderson. *Computational Fluid Dynamics*. McGraw Hill International Editions, 1995.

[5] D. N. Arnold, F. Brezzi, B. Cockburn, and D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Num. Anal.*, 39:1749–1779, 2002.

[6] R. Baltensperger and M. R. Trummer. Spectral differencing with a twist. *SIAM J. Sci. Comput.*, 24(5):1465–1487, 2002.

[7] H. M. Blackburn and S. Schmidt. Spectral element filtering techniques for large eddy simulation with dynamic estimation. *J. Comput. Phys.*, 186:610–629, 2003.

[8] L Boegman. *Currents in Stratified Water Bodies 2: Internal Waves*, pages 539–558. Academic Press - Elsevier, 2009.

[9] O. Botella and R. Peyret. Benchmark spectral results on the lid-driven cavity flow. *Comput. Fluids*, 27:421–433, 1998.

[10] J. P. Boyd. Equatorial solitary waves. Part 3: Westward-travelling modons. *J. Phys. Oceanogr.*, 15:46–54, 1985.

[11] J. P. Boyd. Two comments on filtering (artificial viscosity) for Chebyshev and Legendre spectral and spectral element methods: Preserving boundary conditions and interpretation of the filter as a diffusion. *J. Comput. Phys.*, 143:283–288, 1998.

[12] J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover, Mineola, New York, 2001.

[13] G. Brethouwer, P. Billant, E. Lindborg, and J. M. Chomaz. Scaling analysis and simulation of strongly stratified turbulent flows. *J. Fluid Mech.*, 585:343–368, 2007.

[14] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods. Fundamentals in Single Domains*. Springer-Verlag, 2006.

[15] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods. Evolution to Complex geometries and Applications to Fluid Dynamics*. Springer-Verlag, 2007.

[16] Mark H. Carpenter, David Gottlieb, and Chi-Wang Shu. On the conservation and convergence to weak solutions of global schemes. *J. Sci. Comput.*, 18:111–132, 2003.

[17] M. Carr, P. A. Davies, and P. Shivaram. Experimental evidence of internal solitary wave-induced global instability in shallow water benthic boundary layers. *Phys. Fluids*, 20:Art. No. 0666031, 2008.

[18] P. Castillo, B. Cockburn, I. Perugia, and D. Schötzau. An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. *SIAM J. Num. Anal.*, 38(5):1676–1706, 2000.

[19] J. Chen. How to solve Poisson equation with neumann boundary values. Presentation: w3.pppl.gov/m3d/talks/HowDoUSolveNeumannBC.ppt, 2004.

[20] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.*, 22(104):745–762, 1968.

[21] B. Cockburn, B. Dong, and J. Guzmán. A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems. *Math. Comput.*, 77:1887–1916, 2008.

[22] B. Cockburn and C-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *J. Sci. Comput.*, 16:173–261, 2001.

[23] B. Costa and W. S. Don. On the computation of high order pseudospectral derivatives. *Appl. Num. Math.*, 33:151–159, 2000.

[24] M. O. Deville, P. F. Fischer, and E. H. Mund. *High Order Methods for Incompressible Fluid Flow*. Cambridge University Press, 2002.

[25] P. J. Diamessis, J. A. Domaradzki, and J. S. Hesthaven. A spectral multidomain penalty method model for the simulation of high Reynolds number localized stratified turbulence. *J. Comput. Phys.*, 202:298–322, 2005.

[26] P. J. Diamessis and L. G. Redekopp. Numerical investigation of solitary internal wave-induced global instability in shallow water benthic boundary layers. *J. Phys. Oceanogr.*, 36(5):784–812, 2006.

[27] P. J. Diamessis, G. R. Spedding, and J. A. Domaradzki. Similarity scaling and vorticity structure in high Reynolds number stably stratified turbulent wakes. *J. Fluid Mech.*, 671:52–95, 2011.

[28] W. S. Don, D. Gottlieb, and J. H. Jung. A multidomain spectral method for supersonic reactive flows. *J. Comput. Phys.*, 192:325–354, 2003.

[29] M. Dunphy, C. Subich, and M. Stastna. Spectral methods for internal waves: indistinguishable density profiles and double-humped solitary waves. *Nonlinear Processes Geophys.*, 18(3):351–358, 2011.

[30] A. P. Engsig-Karup, J. S. Hesthaven, H. B. Bingham, and T. Warburton. DG-FEM solution for nonlinear wave-structure interaction using boussinesq-type equations. *Coastal Eng.*, 55:197–208, 2008.

[31] E. Erturk, T. C. Corke, and C. Gokcol. Numerical solutions of 2D steady incompressible driven cavity flow at high Reynolds numbers. *Int. J. Num. Meth. Fluids*, 48:747–774, 2005.

[32] J. A. Escobar-Vargas, P. D. Diamessis, and T. Sakai. A spectral quadrilateral multidomain penalty method solver for high-Reynolds number stratified incompressible flows. *in preparation for J. Comput. Phys.*, 2012.

[33] J. A. Escobar-Vargas, P. D. Diamessis, and C. F. Van Loan. The numerical solution of the pressure Poisson equation for the incompressible Navier-Stokes equations using a quadrilateral spectral multidomain penalty method. *submitted to J. Comput. Phys.*, 2012.

[34] J. A. Escobar-Vargas, P. J. Diamessis, and F. X. Giraldo. High-order discontinuous element-based schemes for the inviscid shallow water equations: Spectral

multidomain penalty and discontinuous Galerkin methods. *Appl. Math. Comput.*, 218(9):4825 – 4848, 2012.

[35] C. Eskilsson and S. J. Sherwin. A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations. *Int. J. Numer. Meth. Fluids*, 45:605–623, 2004.

[36] J. Felcman and O. Havle. On a numerical flux for the shallow water equations. *Appl. Math. Comput.*, 217(11):5160 – 5170, 2011.

[37] K. J. Fidkowski, T. A. Oliver, J. Lu, and D. L. Darmofal. p-multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations. *J. Comput. Phys.*, 207:92–113, 2005.

[38] P. F. Fischer. An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 133(1):84–101, 1997.

[39] P. F. Fischer. Projection techniques for iterative solution of $Ax = b$ with succesive right-hand sides. *Comput. Meth. Appl. Mech. Eng.*, 163:193–204, 1997.

[40] P. F. Fischer and D. Gottlieb. On the optimal number of subdomains for hyperbolic problems on parallel computers. *J. Supercomput. Appl.*, 11:65–76, 1997.

[41] P. F. Fischer, J. W. Lottes, and S. G. Kerkemeier. nek5000 Web page, 2008. http://nek5000.mcs.anl.gov.

[42] P. F. Fischer, N. I. Miller, and H. M. Tufo. An overlapping Schwarz method for spectral element simulation of three-dimensional incompressible flows. In *Parallel Solution of Partial Differential Equations*, pages 159–180. Springer-Verlag, 2000.

[43] A. E. Gill. *Atmosphere-Ocean Dynamics*. Academic Press, San Diego, 1982.

[44] F. X. Giraldo. The Lagrange-Galerkin spectral element method on unstructured quadrilateral grids. *J. Comput. Phys.*, 147(1):114–146, 1998.

[45] F. X Giraldo. A spectral element shallow water model on spherical geodesic grids. *Int. J. Num. Meth. Fluids*, 35(8):869–901, 2001.

[46] F. X. Giraldo. Higher-order triangle-based discontinuous Galerkin methods for hyperbolic equations on a sphere. *J. Comput. Phys.*, 214:447–465, 2006.

[47] F. X. Giraldo, J. S. Hesthaven, and T. Warburton. Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations. *J. Comput. Phys.*, 181:499–525, 2002.

[48] F. X. Giraldo and M. Restelli. A study of spectral element and discontinuous Galerkin methods for mesoscale atmospheric modeling: Equation sets and test cases. *J. Comput. Phys.*, 227:3849–3877, 2008.

[49] F. X. Giraldo and M. Restelli. High-order semi-implicit time-integrators for a triangular discontinuous Galerkin oceanic shallow water model. *Int. J. Num. Meth. Fluids*, 63:1077–1102, 2010.

[50] F. X. Giraldo and T. E. Rosmond. A scalable spectral element eulerian atmospheric model (SEE-AM) for NWP: Dynamical core tests. *Mon. Weather Rev.*, 132:133–153, 2004.

[51] F. X. Giraldo and T. Warburton. A high-order triangular discontinuous Galerkin oceanic shallow water model. *Int. J. Num. Meth. Fluids*, 56:899–925, 2008.

[52] G.H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Third edition, 1996.

[53] D. Gottlieb and J. S. Hesthaven. Spectral methods for hyperbolic problems. *J. Comput. Appl. Math.*, 128:83–131, 2001.

[54] D. Gottlieb and C. L. Streett. Quadrature imposition of compatibility conditions in Chebyshev methods. *J. Sci. Comput.*, 5:223–239, 1990.

[55] S. Gottlieb and J. H. Jung. Numerical issues in the implementation of high order polynomial multi-domain penalty spectral Galerkin methods for hyperbolic conservation laws. *Commun. Comput. Phys.*, 5:600–619, 2009.

[56] M. J. Gourlay, S. C. Arendt, D. C. Fritts, and J. Werne. Numerical modeling of initially turbulent wakes with net momentum. *Phys. Fluids*, 13:3783–3802, 2001.

[57] W. J. F. Govaerts. *Numerical methods for bifurcations of dynamical equilibria*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[58] P. M. Gresho and R. L. Sani. On pressure boundary conditions for the incompressible Navier-Stokes equations. *Int. J. Num. Meth. Fluids*, 7:1111–1145, 1987.

[59] J. L. Guermond and Jie Shen. Velocity-correction projection methods for incompressible flows. *SIAM J. Numer. Anal.*, 41:112–134, 2003.

[60] J. L. Guermond and Jie Shen. An overview of projection methods for incompressible flows. *Comput. Meth. Appl. Mech. Eng.*, 41:112–134, 2006.

[61] D. B. Haidvogel and A. Beckmann. *Numerical Ocean Circulation Modelling*. Imperial College Press, Cambridge, 1999.

[62] D. A. Hebert and S. M. de Bruyn Kops. Predicting turbulence in flows with strong stable stratification. *Phys. Fluids*, 18:Art. No. 066602, 2006.

[63] K. R. Helfrich and W. K. Melville. Long non-linear internal waves. *Ann. Rev. Fluid Mech.*, 38:395–425, 2006.

[64] J. S. Hesthaven. A stable penalty method for the compressible Navier-Stokes equations: II. One-dimensional domain decomposition schemes. *SIAM J. Sci. Comput.*, 18(3):658–685, 1997.

[65] J. S. Hesthaven. A stable penalty method for the compressible Navier-Stokes equations: III. Multidimensional domain decomposition schemes. *SIAM J. Sci. Comput.*, 20(1):62–93, 1998.

[66] J. S. Hesthaven and D. Gottlieb. A stable penalty method for the compressible Navier-Stokes equations: I. Open boundary conditions. *SIAM J. Sci. Comput.*, 17(3):579–612, 1996.

[67] J. S. Hesthaven, S. Gottlieb, and D Gottlieb. *Spectral Methods for Time-Dependent Problems*. Cambridge University Press, 2007.

[68] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods*. Springer-Verlag, 2008.

[69] M. Iskandarani, D. B. Haidvogel, and J. P. Boyd. A staggered spectral element model with application to the oceanic shalow water equations. *Int. J. Num. Meth. Fluids*, 20:393–414, 1995.

[70] F. G. Jacobitz, S. Sarkar, and C. W. Van Atta. Direct numerical simulations of the turbulence evolution in a uniformly sheared and stably stratified flow. *J. Fluid Mech.*, 342:231–261, 1997.

[71] J. H. Jung. *Multi-domain spectral penalty method for hyperbolic systems: Theory and applications*. PhD thesis, Brown University, 2002.

[72] J. H. Jung and B. D. Shizgal. On the numerical convergence with the inverse polynomial reconstruction method for the resolution of the Gibbs phenomenon. *J. Comput. Phys.*, 224:477–488, 2007.

[73] L. H. Kantha and C. A. Clayson. *Numerical Models of Oceans and Oceanic Processes*. Academic Press, San Diego, 2000.

[74] G. E. Karniadakis, M. Israeli, and S. A. Orszag. High-order splitting methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 97:414–443, 1991.

[75] G. E. Karniadakis and S. Sherwin. *Spectral/hp Element Methods for Computational Fluid dynamics*. Oxford University Press, 2005.

[76] G. E. Karniadakis, S. Sherwin, and M. Kirby. Nektar Web page. http://www.nektar.info.

[77] R. M. Kirby and G. E. Karniadakis. Selecting the numerical flux in discontinuous Galerkin methods for diffusion problems. *J. Sci. Comput.*, 22-23:385–411, 2005.

[78] R.M. Kirby and G.E. Karniadakis. De-aliasing on non-uniform grids: algorithms and applications. *J. Comput. Phys.*, 191(1):249–264, 2006.

[79] M. P. Kirkpatrick and S. W. Armfield. Open boundary conditions in numerical simulations of unsteady incompressible flow. In G. N. Mercer and A. J. Roberts, editors, *Proceedings of the 14th Biennial Computational Techniques and Applications Conference, CTAC-2008*, volume 50 of *ANZIAM J.*, pages C760–C773, 2009.

[80] D. A. Kopriva. A spectral multidomain method for the solution of hyperbolic systems. *Appl. Num. Math.*, 2:221–241, 1986.

[81] D. A. Kopriva. Computations of hyperbolic equations on complicated domains with patched and overset Chebyshev grids. *SIAM J. Sci. Stat. Comput*, 10:120–132, 1989.

[82] D. A. Kopriva. Multidomain spectral solution of the Euler gas dynamics equations. *J. Comput. Phys.*, 96:428–450, 1991.

[83] D. A. Kopriva. Multidomain spectral solution of compressible viscous flows. *J. Comput. Phys.*, 115:184–199, 1994.

[84] D. A.. Kopriva. *Implementing Spectral Methods for Partial Differential Equations: Algorithms for Engineers and Scientists*. Springer Verlag, 2009.

[85] P. K. Kundu and I. M. Cohen. *Fluid Mechanics*. Academic Press, San Diego, 2004.

[86] M. Läuter, F. X. Giraldo, D. Handorf, and K. Dethloff. A discontinuous Galerkin method for the shallow water equations in spherical triangular coordinates. *J. Comput. Phys.*, 227(24):10226–10242, 2008.

[87] J. G. Levin, M. Iskandarani, and D. B. Haidvogel. A spectral filtering procedure for eddy-resolving simulations with a spectral element ocean model. *J. Comput. Phys.*, 137:130–154, 1997.

[88] J. G. Levin, M. Iskandarani, and D. B. Haidvogel. To continue or discontinue: Comparisons of continuous and discontinuous Galerkin formulations in a spectral element ocean model. *Ocean Model.*, 15:56–70, 2006.

[89] J. G. Liu and C. W. Shu. A high-order discontinuous Galerkin method for 2D incompressible flows. *J. Comput. Phys.*, 160:577–596, 2000.

[90] P. A. Madsen, D. R. Fuhrman, and H. A.. Schaffer. On the solitary wave paradigm for tsunamis. *J. Geophys. Res.*, 113(C12012):doi:10.1029/2008JC004932, 2008.

[91] B. Metivet and Y. Morchoisne. Multi-domain spectral technique for viscous flow calculations. *4th GAMM Conference on Numerical Methods in Fluid Mechanics*, 134, 1981.

[92] C. Moler. *Experiments with MATLAB*. The Mathworks, 2009.

[93] A. Montlaur. *High-order discontinuous Galerkin methods for incompressible flows*. PhD thesis, Universidad Politecnica de Calatuna, 2009.

[94] L. Olson. Algebraic multigrid preconditioning of high-order spectral elements for elliptic problems on a simplicial mesh. *SIAM J. Sci. Comput.*, pages 2189–2209, 2007.

[95] S. A. Orszag, M. Israeli, and M. O. Deville. Boundary conditions for incompressible flows. *J. Sci. Comput.*, 1:75–111, 1986.

[96] S. A. Orszag and G. S. Patterson. Numerical simulation of three-dimensional homogeneous isotropic turbulence. *Phys. Rev. Lett.*, 28:76–79, 1976.

[97] T. M. Ozgokmen, P. F. Fischer, J. Duan, and T. Iliescu. Three-dimensional turbulent bottom density currents from a high-order nonhydrostatic spectral element model. *J. Phys. Oceanogr.*, 34(9):2006–2026, 2004.

[98] T. M. Ozgokmen, T. Iliescu, and P. F. Fischer. Large eddy simulation of stratified mixing in a three-dimensional lock-exchange system. *Ocean Model.*, 26:134–1556, 2009.

[99] A. T. Patera. A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *J. Comput. Phys.*, 54:468–488, 1984.

[100] R. Peyret. *Spectral Methods for Incompressible Viscous Flow*. Springer-Verlag, New York, 2002.

[101] C. Pozrikidis. A note on the regularization of the discrete Poisson-Neumann problem. *J. Comput. Phys.*, 172:917–923, 2001.

[102] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos National Laboratory, 1973.

[103] M. Restelli and F.X. Giraldo. A conservative discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in nonhydrostatic mesoscale modeling. *SIAM J. Sci. Comput.*, 31:2231–2257, 2009.

[104] E. M. Rønquist. *Optimal spectral element method for the unsteady three dimensional incompressible Navier Stokes Equations*. PhD thesis, Massachusetts Institute of Technology, 1988.

[105] E. M. Rønquist and A. T. Patera. Spectral element multigrid. i. formulation and numerical results. *J. Sci. Comput.*, 2:389–406, 1987.

[106] Y. Saad. *Iterative Methods for the Solution of Large Sparse Systems of Equations*. SIAM, Philadelphia, second edition, 2005.

[107] E. Santilli and A. Scotti. An efficient method for solving highly anisotropic elliptic equations. *J. Comput. Phys.*, 230(23):8342–8359, 2011.

[108] W. W. Schultz, N. Y. Lee, and J. P. Boyd. Chebyshev pseudospectral method of viscous flow with corner singularities. *J. Sci. Comput.*, 4:1–24, 1989.

[109] K. Sengupta, G. B. Jacobs, and F. Mashayek. Large-eddy simulation of compressible flows using a spectral multi-domain method. *Int. J. Num. Meth. Fluids*, 61, 2009.

[110] K. Shabhazi, P. F. Fischer, and C. Ross Ethier. A high-order discontnuous Galerkin method for the unsteady incompressible Navier-Stokes equations. *J. Comput. Phys.*, 222:391–407, 2007.

[111] K. Shabhazi, D. J. Mavripilis, and N. K. Burgess. Multigrid algorithms for high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations. *J. Comput. Phys.*, 228:7917–7940, 2009.

[112] S. J. Sherwin and M. Casarin. Low-energy basis preconditioning for elliptic substructured solvers based on unstructured spectral/hp element discretization. *J. Comput. Phys.*, 171(1):394–417, 2001.

[113] R. J. Spiteri and S. J. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM J. Numer. Anal.*, 40:469–491, 2002.

[114] H. Stommel. The westward intensification of wind-driven ocean currents. *Trans. Amer. Geophys. Union*, 29:202–206, 1948.

[115] B. R. Sutherland. *Internal Gravity Waves*. Cambridge University Press, 2010.

[116] M. A. Taylor and A. Fournier. A compatible and conservative spectral element method on unstructured grids. *J. Comput. Phys.*, 229(17):5879–5895, 2010.

[117] H. Tennekes and J. L. Lumley. *A First Course in Turbulence*. The MIT press, 1972.

[118] S. A. Thorpe. Fronts formed by obliquely reflecting internal waves at a sloping boundary. *J. Phys. Oceanogr.*, 29:2462–2467, 1999.

[119] S. A. Thorpe. *The Turbulent Ocean*. Cambridge University Press, 2005.

[120] E. F. Toro. *Shock-capturing methods for Free-surface shallow flows*. John Wiley and Sons, 2001.

[121] R. Touma. Central unstaggered finite volume schemes for hyperbolic systems: Applications to unsteady shallow water equations. *Appl. Math. Comput.*, 213(1):47 – 59, 2009.

[122] B. Turkington, A. Eydeland, and S. Wang. A computational method for solitary internal waves in a continuously stratified fluid. *Stud. Appl. Math.*, 85:93–127, 1991.

[123] T. Warburton, L. F. Pavarino, and J. S. Hesthaven. A pseudo-spectral scheme for the incompressible Navier-Stokes equations using unstructured nodal elements. *J. Comput. Phys.*, 164:1–21, 2000.

[124] J. A. C. Weideman and L. N. Trefethen. The eigenvalues of second order spectral differentiation matrices. *SIAM J. Numer. Anal.*, 25:1279–1298, 1988.

[125] C. D. Winant and F. K. Browand. Vortex pairing : the mechanism of turbulent mixing-layer growth at moderate Reynolds number. *J. Fluid Mech.*, 63(02):237–255, 1974.

[126] K. B. Winters, J. McKinnon, and B. Mills. A spectral model for process studies of density stratified flows. *J. Atmos. Ocean. Technol.*, 21(1):69–94, 2004.

[127] Wen-Xian Xie, Li Cai, Jian-Hu Feng, and Wei Xu. Computations of shallow water equations with high-order central-upwind schemes on triangular meshes. *Appl. Math. Comput.*, 170(1):296 – 313, 2005.

[128] Thomas A. Zang. On the rotation and skew-symmetric forms for incompressible flow simulations. *Appl. Num. Math.*, 7(1):27 – 40, 1991.