# The numerical solution of the pressure Poisson equation for the incompressible Navier-Stokes equations using a quadrilateral spectral multidomain penalty method

J. A. Escobar-Vargas[a,*], P. J. Diamessis[a], C. F. Van Loan[b]

[a] *School of Civil and Environmental Engineering, Cornell University, Ithaca, NY 14853*
[b] *Department of Computer Science, Cornell University, Ithaca, NY 14853*

**Abstract**

We outline the basic features of a spectral multidomain penalty method (SMPM)-based solver for the pressure Poisson equation (PPE) with Neumann boundary conditions, as encountered in the time-discretization of the incompressible Navier-Stokes equations. One one hand, the SMPM discretization enables robust under-resolved simulations without sacrificing high accuracy. On the other, the solution of the PPE is an inevitable requirement when simulating strongly non-hydrostatic flows, such as those occuring in the natural environment. 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. The accuracy and efficiency of the PPE solver are demonstrated through application to two incompressible flow benchmarks, the Taylor vortex and the Lid-driven cavity flow. In addition to providing an efficient tool for the iterative solution of the PPE in incompressible flow simulations, this work presents algorithms which are of interest to the subdiscipline of the numerical linear algebra community focused on the iterative solution of consistent singular non-symmetric linear systems of equations.

*Keywords:* Computational fluid dynamics, Pressure Poisson equation, Neumann boundary conditions, Spectral multidomain method, Non-symmetric matrices, Singular consistent system, Singular value decomposition, Two-level preconditioner.

*Jorge A. Escobar-Vargas, School of Civil and Environmental Engineering, Cornell University, Ithaca, NY 14853, U.S.A. . email: jae47@cornell.edu

## 1. Introduction

Spectral multidomain penalty methods (SMPM) [1, 2, 3] are a class of discontinuous higher-order (spectral) accuracy element-based discretization techniques designed to have superior numerical stability properties as compared to equivalent continuous methods, such as the spectral element (SEM) and spectral multidomain methods [4, 5, 6, 7]. SMPM consist of collocating a linear combination of the governing equation and boundary or patching (subdomain communication) conditions, the latter multiplied by a penalty coefficient, at the boundaries or subdomain interfaces, respectively. Numerical stability at the subdomain interfaces is thereby enabled by weakly enforcing the boundary/patching conditions, leading to a weakly discontinuous solution at the subdomain interfaces. In this regard, the SMPM strongly resembles the discontinuous Galerkin method (DGM) [8, 9]. However, the former is a collocation-based formulation, whereas the latter works with a weighted residual nodal Galerkin formulation with appropriately specified numerical fluxes specified at the subdomain interfaces.

The enhanced numerical stability of SMPM and DGM, combined with the exponential convergence, near-negligible artificial dissipation and dispersion and spatial adaptivity of element-based higher-order discretization techniques, make them particularly attractive as a computational tool for under-resolved fluid flow simulations. One class of fluid flow phenomena where under-resolved fluid flow simulations are inevitable are smaller-scale environmental stratified flow processes, such as localized turbulent bursts [10] and internal solitary waves [11]. Such phenomena are subject to a very broad scale separation and strongly nonlinear and non-hydrostatic dynamics. Currently available computational resources allow the simulation of only the larger, energetic scales of an environmental turbulent flow with the smaller, more viscously-dominated, scales remaining unresolved [12]. The accurate reproduction of internal solitary wave evolution requires the adequate resolution of the steep waveform, a direct result of the waves' strong nonlinearity, without over-resolving the wave interior to allow for a sufficiently large domain for the wave to propagate within [13].

The DGM has been applied primarily to the study of mesoscale phenomena (of scale $O$(10 to 100km ) ) in the ocean and atmosphere [14, 15], where the relevant dynamics are governed by the compressible Navier-Stokes equations (NSE), a hyperbolic system of equations, for which the DGM is ideally suited [8]. However, smaller-scale environmental stratified flow processes are inherently incompressible [16]. Moreover, they are strongly non-hydrostatic and contain significant vertical accelerations, including the vertical pressure gradient term which cannot be neglected. In the case of internal solitary waves, in particular, non-hydrostatic effects provide the necessary physical dispersion that allow these waves to maintain their steep waveform over large propagation distances. Invoking the hydrostatic assumption, typically used in larger-scale geophysical flow modeling [17], will force the simulation to produce highly erroneous phys-

ical results. One must solve numerically either the Poisson or Stokes equation [18] that arises when treating the pressure in an non-hydrostatic incompressible flow simulation [19].

Although DGM solvers of the incompressible NSE [20, 21] have been reported in the literature, none have been applied to strongly under-resolved incompressible flow simulations, including those of environmental stratified flows. In contrast, a SMPM model has been developed and used by the second author of this paper for high Reynolds number incompressible stratified turbulent flows in domains with one non-periodic direction (the vertical) [22, 23, 12]. One-dimensional spectral subdomains, based on Legendre polynomials, are used in the vertical. The computational domain is a rectangular box the with periodic horizontal directions which used a Fourier polynomial discretization.

The time-discretization, originally proposed by Karniadakis and co-workers [24] (hereafter referred to as KIO), used in the above SMPM model requires the solution of a Poisson equation for the pressure with Neumann boundary conditions. However, solution of this Poisson equation is only required for the horizontal zero Fourier mode. In practice, on account of incompressibility and the use of rigid and impermeable top and bottom boundaries, the zero Fourier mode for the pressure mode is directly set to zero and the corresopnding linear system of equations does not have to be solved.

Should more complex domain geometries and non-periodic streamwise boundary conditions be necessary, a quadrilateral spectral subdomain discretization [3] is needed. In the context of the KIO time discretization, the solution of the above Poisson-Neumann problem for the pressure is now inevitable. 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 [25]. In the KIO scheme, the compatibility condition is inherently satisfied at the spatially continuous level [24]. However, under-resolution and the presence of the penalty terms can cause a violation of the compatibility condition (see reference [26] and §5.3 of this paper), 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 [18], have been efficiently addressed through the development of appropriate preconditioning techniques [27, 28, 29]. 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 [7]. 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, this paper 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.

## 2. Incompressible Navier-Stokes equations: temporal discretization

In this section we review the two time-discretization strategies, typically applied by the higher-order method community to the incompressible Navier-Stokes equations [30]:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{F} \quad , \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0 \quad , \tag{2}$$

where $\mathbf{u}$ is the velocity vector, $\rho$ is the density, $p$ is the pressure, $\nu$ is the kinematic viscosity of the fluid, and $\mathbf{F}$ contains any additional forcing terms. In the case of environmental stratified flow processes, the forcing term accounts for the restoring force of gravity in the vertical momentum equation, commonly represented in the form of the Boussinesq approximation [31]. In what follows, we will neglect this term, as its role is not critical to the solution of the PPE.

The defining differences between possible temporal discretization strategies for equations in (1) and (2) are whether the the pressure gradient and viscous terms are coupled or not and whether boundary conditions are required for the pressure field. As a result, one may classify these strategies into main groups: Stokes-based solvers and fractional-time-stepping methods.

### *2.1. Stokes-based solvers*

In this approach the non-linear terms are solved explicitly and treated as forcing terms ($\mathbf{f}$) of a generalized Stokes problem [30], where the velocity and pressure are coupled through a discrete operator with the following structure

$$\begin{pmatrix} \mathbf{H} & -B^T \\ -B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{0} \end{pmatrix} \quad . \tag{3}$$

In (3), $\mathbf{H}$ is the discrete analog of the Helmholtz operator, and $B$ and $B^T$ are the discretized gradient operator and divergence operators respectively [30]. The Uzawa algorithm [18, 5] is used to solve this system of equations which is modified such that a block upper triangular matrix is obtained and a block back substitution is subsequently performed. This procedure decouples velocity from pressure at the discrete level. One of the well-documented challenges of this approach is the fulfillment of the inf-sup condition [32], which establishes the coupling between velocity and pressure in order to satisfy existence and uniqueness (up to an additve constant) of the solution. In practical terms, fulfillment of this condition necessitates a staggered grid, where the pressure field is approximated with a space of basis functions of lower degree than the one used for the velocity, i.e. $\mathbb{P}^{N-2}$ vs $\mathbb{P}^N$. As a result, no boundary conditions are required for the pressure. As an alternative treatment of the inf-sup condition, a stabilization term can be added to the equations [33].

References [34, 27, 28, 18] discuss various preconditioning strategies used in the numerical computation of the pressure component of the Stokes system of equations. Such techniques typically rely on coarse grid preconditioners, fast diagonalization method, additive Schwarz, and two-level preconditioners among others [18].

The time discretization associated with the Stokes equations and underlying Uzawa algorithm has been the foundation of a number of SEM-based investigations of incompressible flows [35, 27], which include studies of environmental flow

processes [36, 37], where enhanced numerical stability is enabled through over-integration-based de-aliasing techniques [38]. However, penalty schemes have so far been developed for single-variable partial differential equations, specifically the advection-diffusion equation [1, 3] and adapted accordingly to the fractional steps of the KIO scheme for one-dimensional subdomains [22]. As a result, it is unclear how one might formulate a penalty scheme for the actual Stokes equations.

### *2.2. Fractional step methods*

In projection methods, the velocity is decoupled from the pressure, the latter having the primary role of enforcing the incompressibility. As a result, a separate Poisson equation for the pressure must be solved. Here we describe a very commonly used approach from this family of methods, the operator splitting scheme proposed by Karniadakis et. al [24],which is a more evolved, higher temporal accuracy, version of the scheme originally proposed by Chorin and Temam [30]. Whereas Chorin and Temam lump the viscous term and nonlinear term in a single fractional step, with the nonlinear treated explicitly, and reserve a separate step for the pressure, the KIO scheme allots the following independent fractional steps to each of the spatial terms in the right-hand side of (1):

$$\frac{\hat{\mathbf{v}} - \sum_{q=0}^{J_i-1} \alpha_q \mathbf{v}^{n-q}}{\Delta t} = \sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{v^{n-q}}), \tag{4}$$

$$\frac{\hat{\hat{\mathbf{v}}} - \hat{\mathbf{v}}}{\Delta t} = \nabla p^{n+1}, \tag{5}$$

$$\frac{\gamma_0 \mathbf{v}^{n+1} - \hat{\hat{\mathbf{v}}}}{\Delta t} = \nu \nabla^2 \mathbf{v}^{n+1}, \tag{6}$$

where $\hat{\mathbf{v}}$ and $\hat{\hat{\mathbf{v}}}$ are intermediate velocity fields. In the first fractional step, (4), the non-linear term is solved explicitly via a stiffly stable scheme [24], also regarded as Adams-Bashforth/Backward Differentiation (AB/BDEk) [39]. In the second step, (5), a Poisson equation with Neumann boundary conditions has to be solved for the pressure to enforce incompressibility by leading to an intermediate velocity field $\hat{\hat{\mathbf{v}}}$ that is divergence-free ($\nabla \cdot \hat{\hat{\mathbf{v}}} = 0$). Finally, in the third fractional step, (6), a modified Helmholtz equation is solved implicitly to obtain the final velocity field ($\mathbf{v}^{n+1}$) at each time step. A more general and detailed analysis of projection methods for incompressible flows is presented in [40] and specifically for high-order methods in [30].

In its original presentation, the application and analysis of the KIO scheme is performed in the framework of the SEM method [24]. As a result, no concerns arise in terms of the compatibility condition which is assumed to be satisfied naturally. Moreover, to the best of our knowledge, there is no discussion in the literature of issues with the singularity of the resulting Poisson matrix which arise from the Neumann boundary conditions. Finally, efficient preconditioners

for the SEM-based KIO approach are based on a transformation of the expansion basis to a low-energy basis [29, 41], which is amenable to block diagonal preconditioning.

Nonetheless, as is elaborated further in §5.2, application of an SMPM-based spatial discretization to the incompressible NS equations within the KIO framework, poses concerns about Poisson matrix singularity and violation of a compatibility condition. Moreover, an extension of preconditioners developed for continuous element-based schemes, such as SEM, and the associated symmetric Poisson matrix to the discontinuous collocation-based SMPM and the corresponding non-symmetric matrix is not a straightforward task. Identifying a robust framework for the iterative solution of the linear system of equations resulting from the SMPM-discretization pressure Poisson equation is the primary objective of this paper.

## 3. The pressure Poisson equation

### *3.1. Compatibility condition*

In the KIO splitting scheme, the PPE is obtained by taking the divergence of Eq. (5)

$$\nabla \cdot \frac{\hat{\hat{\mathbf{v}}} - \hat{\mathbf{v}}}{\Delta t} = \nabla \cdot \nabla p^{n+1}, \tag{7}$$

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

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

A Poisson equation with Neumann boundary conditions therefore results:

$$\nabla^2 p = \nabla \cdot \left( -\frac{\hat{\mathbf{v}}}{\Delta t} \right) = f \quad \text{on} \quad \Omega, \tag{8}$$

$$\frac{\partial p}{\partial n} = \mathbf{n} \cdot \left[ \sum_{q=0}^{J_e - 1} \beta_q \mathbf{N}(\mathbf{v^{n-q}}) + \nu \beta_q \mathbf{L}(\mathbf{v^{n-q}}) \right] = q \quad \text{on} \quad \Gamma. \tag{9}$$

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 [24].

The right hand side $f$ and boundary operator $q$ must satisfy a compatibility condition for the PPE (Eq. (8)-(9)) to have a solution. Specifically, the Poisson-Neumann problem 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. (8) over the whole domain we obtain

$$\int_\Omega \nabla^2 p \, d\Omega = \int_\Omega f \, d\Omega, \tag{10}$$

and by employing Gauss' theorem

$$\int_{\Omega} \nabla^2 p \, d\Omega = \int_{\Gamma} n \cdot \nabla p \, d\Gamma, \tag{11}$$

$$\int_{\Omega} f \, d\Omega = \int_{\Gamma} q \, d\Gamma. \tag{12}$$

Therefore, the Poisson-Neumann problem (8)-(9) has a solution only if (12) is satisfied [24, 25, 42, 43]. As already indicated in §2.2, in the original presentation of the KIO scheme, it is emphasized that the boundary integral of (11) is transformed by Gauss' theorem into a volume integral where the divergence of the second term in the original integrand vanishes. As a result,

$$\sum_{q=0}^{J_e-1} \beta_q \int_{\Omega} \nabla \cdot (\mathbf{N})^{n-q} \, d\Omega = \int_{\Omega} \nabla \cdot \left( \frac{\hat{\mathbf{v}}}{\Delta t} \right) \, d\Omega \tag{13}$$

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

*3.2. 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

$$\mathbf{n} \cdot \nabla p = g \, \text{on} \, \Gamma, \tag{14}$$

any function of the form

$$p(\mathbf{x}) = g(\mathbf{x}) + h, \tag{15}$$

where $h$ is an indeterminate additive constant, satisfies the boundary conditions (9) 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 $\hat{\hat{\mathbf{v}}}$ through (5) since only the the gradient of the pressure field ($\nabla p$) 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.

## 4. The penalty-based discrete pressure Poisson equation

*4.1. Spectral multidomain penalty method (SMPM)*

Within each quadrilateral subdomain, on a Gauss-Lobatto-Legendre (GLL) grid, any function $p(x, z, t)$ is approximated in nodal form as the tensor product of its Lagrange interpolating polynomials of degree $N$ [3] in each spatial

dimension:

$$p(x,y,t) \approx \sum_{i=0}^{N}\sum_{j=0}^{N} p(x_i,y_j,t) l_i(x) l_j(y), \tag{16}$$

where $l_i(x)$ ( $l_j(y)$ ) is the $i-$th ( $j-$th ) Lagrange interpolating polynomial computed for GLL node $x_i$ ( $(y_j)$ ) for a fixed value of $j$ ( $i$ ), respectively.

*4.2. Spectral differentiation matrices*

Discrete derivatives of the function $p$ on the local GLL grid are computed by means of spectral differentiation matrices. For the one-dimensional case and a single subdomain, where the global coordinate $x$ and local coordinate $\xi$ in the canonical interval $[-1,1]$ are linked through a linear mapping $x(\xi)$, the $m-$th derivative in the $x-$direction is approximated as [44]

$$\frac{\partial^m p(x_i,t)}{\partial x^m} = \frac{\partial^m p(x_i,t)}{\partial \xi^m}\left(\frac{\partial \xi}{\partial x}\right)^m \approx \left(\frac{\partial \xi}{\partial x}\right)^m \sum_{k=0}^{N} d_{ik}^m p(x_k,t) = \mathbf{D_N^m p}\ . \tag{17}$$

In (17), $d_{ij}^m$ are the entries of the Legendre spectral differentiation matrix $\mathbf{D}_N^m$ computed following the algorithm outlined in Costa and Don [44]. The properties of $\mathbf{D}_N^m$ are outlined in detail in [45].

The computation of derivatives for a single one-dimensional domain can be extended in a straightforward fashion to multiple one-dimensional subdomains, two-dimensional domains and to a two-dimensional multidomain framework by using Kronecker (tensor) products [46, 47, 18]:

- One-dimensional multidomain:

$$\frac{d^m \mathbf{p}}{dx^m} = \left(\mathbf{I}_{nx} \otimes \mathbf{D}_N^m\right)\mathbf{p} \tag{18}$$

- Two-dimensional single domain:

$$\frac{\partial^m \mathbf{p}}{\partial x^m} = \left(\mathbf{I}_N \otimes \mathbf{D}_N^m\right)\mathbf{p} \tag{19}$$

$$\frac{\partial^m \mathbf{p}}{\partial z^m} = \left(\mathbf{D}_N^m \otimes \mathbf{I}_N\right)\mathbf{p} \tag{20}$$

- Two-dimensional multidomain:

$$\frac{\partial^m \mathbf{p}}{\partial x^m} = \left(\mathbf{I}_{nz} \otimes \mathbf{I}_{nx} \otimes \mathbf{I}_N \otimes \mathbf{D}_N^m\right)\mathbf{p} \tag{21}$$

$$\frac{\partial^m \mathbf{p}}{\partial z^m} = \left(\mathbf{I}_{nz} \otimes \mathbf{I}_{nx} \otimes \mathbf{D}_N^m \otimes \mathbf{I}_N\right)\mathbf{p} \tag{22}$$

where $\mathbf{I}_k$ are identity matrices of dimension $k$, $nx$ is the number of subdomains in the $x$-direction, and $nz$ is the number of subdomains in the $z$-direction. Based

on this definition, $\mathbf{I}_N \otimes \mathbf{D}_N^m$ and $\mathbf{D}_N^m \otimes \mathbf{I}_N$ account for the horizontal and vertical derivatives within each subdomain respectively. Additionally, $\mathbf{I}_{nx}$ extends the computed derivatives across all subdomains in the $x-$direction. Finally, $\mathbf{I}_{nz}$ enables the corresponding operation across all subdomains in the $z-$direction. When a spectral differentiation matrix is used in an implicit solve in the spectral multidomain framework, as is done with the $D_N^2$ matrix for the PPE in §4, the above expressions are simply augmented with the necessary penalty terms.

*4.3. Penalty formulation*

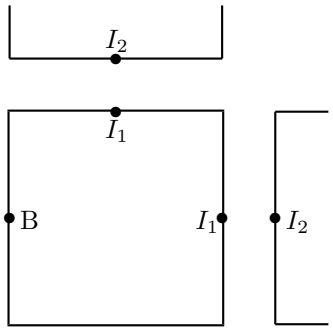


Figure 1: Schematic of a subdomain subject to penalty treatment. $I_1$ denotes a point along an internal interface of the subdomain and $B$ represents a point on a physical boundary. $I_2$ denotes the corresponding interfacial point of the subdomain neighboring point $I_1$

In the penalty formulation, for any collocation point $I_1, I_2$ located along any subdomain interface or physical boundary (see Fig. 1), the Poisson equation is recast as

$$\nabla^2 p + \tau \cdot [condition] = f, \tag{23}$$

where $\tau$ is a penalty coefficient, and the *condition* term in (23) represents the patching or boundary condition at the subdomain or physical boundary, respectively, shown in detail in the next two sections. The weak enforcement of the patching or boundary condition provides for enhanced stability of the numerical scheme. The penalty formulation is applied to each one of the three fractional steps in (4)-(6) in the form of a two-dimensional extension of the one-dimensional multidomain formulation outlined in [22]. Moreover, in this paper, we restrict our presentation to rectangular subdomains, although the extension to arbitrary quadrilaterals is straightforward [3].

*4.3.1. Penalty formulation at subdomain interfaces*

The spatial discretization of the penalized PPE for a point $I_1$ located at an interface (i.e. vertical or horizontal) takes the form [48]

$$\nabla^2 p^{(I_1)} + \tau Q(\mathbf{x}) \left( \alpha p^{(I_1)} + \beta \mathbf{n} \cdot \nabla p^{(I_1)} - g^{(I_2)}(\mathbf{x}) \right) = f^{(I_1)}, \tag{24}$$

where

$$g^{(I_2)}(\mathbf{x}) = \gamma p^{(I_2)} + \delta \mathbf{n} \cdot \nabla p^{(I_2)}. \tag{25}$$

In this case, the variables $\alpha, \beta, \gamma, \delta$ are constants of the penalty method, set to one in practice [1, 22], 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 [1]. Following [22] 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 [1, 22, 3]

$$\tau = \frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa - 2\sqrt{\kappa^2 + \varepsilon\kappa}\right]\frac{2}{L_x^I} \leq \tau \leq \frac{1}{\omega\varepsilon\beta}\left[\varepsilon + 2\kappa + 2\sqrt{\kappa^2 + \varepsilon\kappa}\right]\frac{2}{L_x^I}, \tag{26}$$

where $\omega = 2/(N(N-1))$ is a GLL quadrature weight, $\varepsilon$ is the corresponding diffusion coefficient, set to one [48] and $\kappa = \omega\alpha/\beta$ [1, 48]. 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. (26). Our practical experience dictates that a choice of $\tau$ 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.

*4.3.2. 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. 1) is given by

$$\nabla^2 p^B + \tau Q(\mathbf{x})\left(\beta \mathbf{n} \cdot \nabla p^B - g^B\right) = f^B, \tag{27}$$

with

$$g^B = q^B, \tag{28}$$

where $q^B$ is the prescribed value for the boundary condition at the boundary point $B$, given by Eq. (9), and the remaining variables are the same as for the interfacial case. The penalty term $\tau$ is now defined as [1]

$$\tau = \frac{1}{\beta\omega} \tag{29}$$

where $\omega$ is again the GLL weight at the collocation point $B$.

## 5. Properties of the discrete pressure Poisson equation

### *5.1. The discrete Poisson pressure equation*

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

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{30}$$

where the matrix $\mathbf{A}$ is the discrete analog of the penalized Laplacian and is constructed from the tensor product definitions given in Eq. (21)-(22) augmented with the contribution of the boundary/patching conditions at the boundaries/interfaces. Additionally, $\mathbf{x}$ is the solution vector (i.e. the pressure), and $\mathbf{b} = \nabla \cdot \left(-\frac{\hat{\mathbf{v}}}{\Delta t}\right)$ is the right-hand-side vector which contains information from the convective term and the Neumann boundary conditions (see Eq. (8) -(9)).

Fig. 2 shows the structure of the Poisson matrix $\mathbf{A}$, for a 3 by 3 subdomain example with order of polynomial approximation $N = 8$.

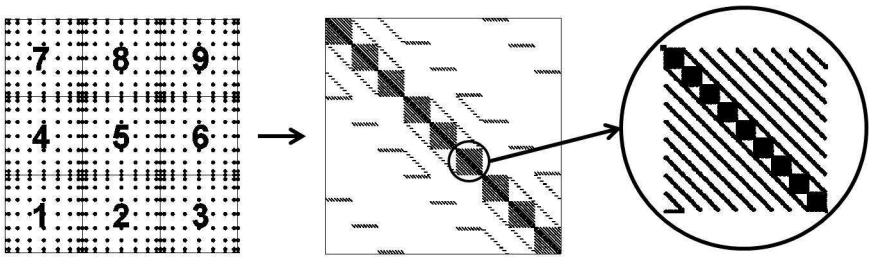


Figure 2: Left panel: SMPM Gauss-Lobatto-Legendre grid on the left ($3 \times 3$ subdomains with $N = 8$). Central panel: Structure of the corresponding Poisson matrix $\mathbf{A}$. Right panel: Structure of the contribution of each subdomain into the global matrix $\mathbf{A}$.

As shown in the exploded view of the right panel of Fig. 2, the smaller-size blocks originate from the second derivative with respect to $x$ (eqn (21)), whereas the remaining elements account for the second derivative with respect to $z$ (eqn (22)). The additional entries within the matrix $\mathbf{A}$, visible in the central panel of Fig. 2, 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.

### *5.2. 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 $\mathbf{A}$ is given by

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}}, \tag{31}$$

where $\mathbf{U}$ and $\mathbf{V}$ are two orthogonal matrices that contain the left and right singular vectors, and $\Sigma$ is a diagonal matrix with the singular values as its diagonal entries. Fig. 3 shows the singular value distribution for the example matrix shown in Fig. 2. 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 $O(\approx 10^{17})$. §**??** demonstrates explicitly the connection between the zero singular value and the non-uniqueness of the solution associated with the discrete Poissonn equation.

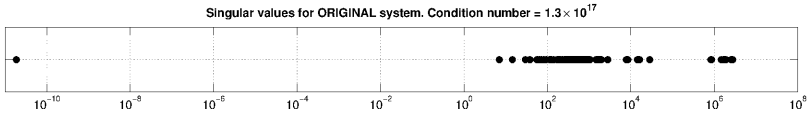


Figure 3: Singular value distribution of the Poisson matrix $\mathbf{A}$: Case of $3 \times 3$ subdomains and $N = 8$

*5.3. Compatibility condition revisited*

The question arises whether Eq. (12) 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. (24) and (27), 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. (12). 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. (12) 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 [26], remains unclear to us. Using the strategy originally proposed by Gottlieb and Streett [26], 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 (12) 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.

*5.4. Consistency of the linear system of equations*

The system of equations (30) is consistent if

$$\mathbf{u_0^T A x} = \mathbf{u_0^T b} = 0, \tag{32}$$

where $\mathbf{u_0}$ is the left null singular vector of the matrix $\mathbf{A}$ [46]. Eq. (32) 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 [25], this rationale is outlined for matrices obtained for low-order schemes and real eigenvalues, in the context of an eigen-decomposition of the matrix $\mathbf{A}$ and its transpose $\mathbf{A}^T$.

In practice the condition (32) 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 (30) orthogonal to the left null singular vector $\mathbf{u_0}$ [25], i.e.

$$\mathbf{A x} = (\mathbf{I} - \mathbf{u_0 u_0^T})\mathbf{b} = \hat{\mathbf{b}} \tag{33}$$

where $\hat{\mathbf{b}}$ is the orthogonal complement of $\mathbf{b}$ onto $\mathbf{u_0}$. Consistency, as represented by Eq. (32), is now ensured to machine epsilon since

$$\begin{aligned} \mathbf{u_0^T A x} &= \mathbf{u_0^T}(\mathbf{b} - \mathbf{u_0 u_0^T b}) \\ &= \mathbf{u_0^T b} - \mathbf{u_0^T u_0 u_0^T b} = 0 \end{aligned} \tag{34}$$

It is important to recall that if the PPE matrix is symmetric a standard eigen-decomposition may be used where there is only one null eigenvector which is a constant vector [25]. In this case, the implementation of (33) 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 §7.

## 6. 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.

### *6.1. Commonly used strategies*

#### *6.1.1. Dirichlet boundary condition at a single point*

This widely used technique consists of imposing a Dirichlet condition at one point along the physical boundaries [49]. 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 $\mathbf{A}$ 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 $\mathbf{A}$. 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 [50]. 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.

#### *6.1.2. 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 [18]:

$$\int_{\Omega} p \, d\Omega = 0. \tag{35}$$

The discrete analog of Eq. (35) consists of adding one row with the Gauss-Legendre integration weights to the global matrix $\mathbf{A}$ 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 (35) can be imposed in the form of a penalty term, i.e. by solving

$$\nabla^2 p + \tau \int_{\Omega} p \, dp = f \quad , \tag{36}$$

which in matrix form becomes

$$\mathbf{A}\mathbf{x} + \tau \mathbf{1}\mathbf{w}^{\mathbf{T}}\mathbf{x} = \mathbf{b} \tag{37}$$

where $\tau$ 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 $(\mathbf{A} + \mathbf{1}\mathbf{w}^{\mathbf{T}})$ 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

$$\mathbf{x} = (\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}})^{-\mathbf{1}}\hat{\mathbf{b}} \tag{38}$$

$$\mathbf{x} = \frac{\mathbf{u_0^T}\hat{\mathbf{b}}}{\sigma_0}\mathbf{v_0} + \sum_{i-1}^{N} \frac{\mathbf{u_i^T}\hat{\mathbf{b}}}{\sigma_i}\mathbf{v_i}, \tag{39}$$

where $\mathbf{u_i}, \mathbf{v_i}$ are the left and right singular vectors of the matrix $\mathbf{A}$, and $\sigma_i$ are the corresponding singular values. Thus, in Eq. (39), 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 (39) 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 (15). 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. (39). The above regularization technique is similar to the one used to enforce consistency of the linear system of equations (see Eq. (33) ). 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 [18], 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.

### *6.2. Strategies for the SMPM-discretized Poisson equation*

#### *6.2.1. Reduced system via Householder matrices*

This approach is based on a combination of the SVD with Householder matrices [46]. 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 $\mathbf{A}$.

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 $\mathbf{A}$. For each one of these two vectors, an orthonormal basis $\mathbf{P}$ and $\mathbf{Q}$ can be built using Householder transformations,

$$\mathbf{P} = \mathbf{I} - 2\frac{\mathbf{h}_L \mathbf{h}_L^T}{\mathbf{h}_L^T \mathbf{h}_L} = [\mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_N}], \tag{40}$$

$$\mathbf{Q} = \mathbf{I} - 2\frac{\mathbf{h}_R \mathbf{h}_R^T}{\mathbf{h}_R^T \mathbf{h}_R} = [\mathbf{q_1}, \mathbf{q_2}, \cdots, \mathbf{q_N}], \tag{41}$$

where $\mathbf{h}_L$ and $\mathbf{h}_R$ are the left and right Householder vectors [46], and $\mathbf{p}_i, \mathbf{q}_i$, with $i = 1, \cdots, n$, are the column vectors of the matrices $\mathbf{P}$ and $\mathbf{Q}$ 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 $\mathbf{Q}_r$

$$\mathbf{P} = [\mathbf{u_0}, \mathbf{p_2}, \cdots, \mathbf{p_N}] \rightarrow \mathbf{P}_r = [\mathbf{p_2}, \cdots, \mathbf{p_N}], \tag{42}$$

$$\mathbf{Q} = [\mathbf{v_0}, \mathbf{q_2}, \cdots, \mathbf{q_N}] \rightarrow \mathbf{Q}_r = [\mathbf{q_2}, \cdots, \mathbf{q_N}]. \tag{43}$$

Following some algebraic manipulations, the reduced system of Eq. (44) is finally written as

$$\mathbf{P}_r^T \mathbf{A} \mathbf{Q}_r \mathbf{y} = \mathbf{P}_r^T \mathbf{b}, \ , \tag{44}$$

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 $\mathbf{A}$ 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 $3 \times 3$ subdomains and $N = 8$ is shown in Fig. 4. The resulting modified singular value distribution is equivalent to eliminating the term $\mathbf{u_0^T b}/\sigma_\mathbf{0}$ from Eq. (39), which translates into a unique solution for the system of equations and a significantly lower condition number for the new matrix $\mathbf{P}_r^T \mathbf{A} \mathbf{Q}_r$.

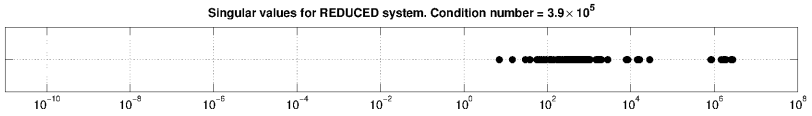


Figure 4: Singular value distribution of the matrix $\mathbf{U_R A V_R^T}$, where $\mathbf{A}$ is the matrix of Fig. 3, obtained with the reduced system technique via Householder matrices. Unlike Fig. 3, the null singular value $\sigma_0$ is now absent.

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

$$\mathbf{x} = \mathbf{Q}_r \mathbf{y} \tag{45}$$

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.

*6.2.2. Augmented system via bordered systems*

An alternative approach is based on the concept of augmented (bordered) systems [51]. In this case, the augmented system of equations is expressed as

$$\begin{pmatrix} \mathbf{A} & \mathbf{d} \\ \mathbf{c}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \alpha \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ g \end{pmatrix} \tag{46}$$

where $\mathbf{c}$ and $\mathbf{d}$ are two vectors of dimension $n$ that satisfy the following conditions

$$\mathbf{d}^T \mathbf{u}_0 \neq 0 \tag{47}$$

$$\mathbf{c}^T \mathbf{v}_0 \neq 0 \tag{48}$$

By expanding Eq. (46) we obtain

$$\mathbf{A}\mathbf{x} + \alpha \mathbf{d} = \mathbf{b} \tag{49}$$

$$\mathbf{c}^T \mathbf{x} = 0 \tag{50}$$

If Eq. (49) is multiplied by $\mathbf{u}_0^T$ the only way in which the system is consistent is for $\alpha = 0$

$$\mathbf{u}_0^T \mathbf{A}\mathbf{x} + \alpha \mathbf{u}_0^T \mathbf{d} = \mathbf{u}_0^T \mathbf{b} \tag{51}$$

on the other hand, by imposing $\mathbf{c} = \mathbf{v}_0$, uniqueness is ensured (see Eq. (**??**)), and the additive constant value is specified by $g$. The singular value distribution of the augmented matrix is shown in Fig. (5).

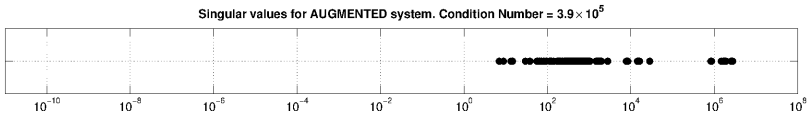


Figure 5: Singular value distribution for the augmented system corresponding to the matrix $\mathbf{A}$ of Fig. 3.

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 $\mathbf{d}$ 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.

## 7. Computation of the left null singular vector

The efficient computation of the left null singular vector (LNSV) $\mathbf{u}_0^{(2\mathbf{d})}$ of the matrix $\mathbf{A}$, 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 §5.4), and the techniques that remove the matrix singularity by reducing or augmenting the system of equations cannot be implemented (see §6). Computing this null vector by performing the full SVD of the Poisson matrix is computationally costly and actually becomes prohibitive as the matrix $\mathbf{A}$ 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 [52] 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 §4.2) is also a highly challenging, if not impossible, task,

which is outside of the scope of this paper.

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 §4.2). 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*.

*7.1. 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. 6 shows an example of the LNSV structure for a domain with $3 \times 3$ subdomains, and $N = 4$. The observed LNSV structure and associated computation procedure outlined below, can be directly extended to any number of subdomains.

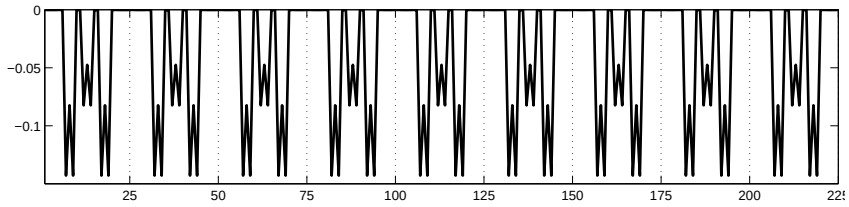


Figure 6: Left null singular vector $\mathbf{u_0}$ for an example of a doubly periodic domain with $3 \times 3$ subdomains, and $N = 4$. The vertical dashed lines separate the contributions of individual subdomains, i.e. the subvector $\mathbf{sv}$ shown in 7.

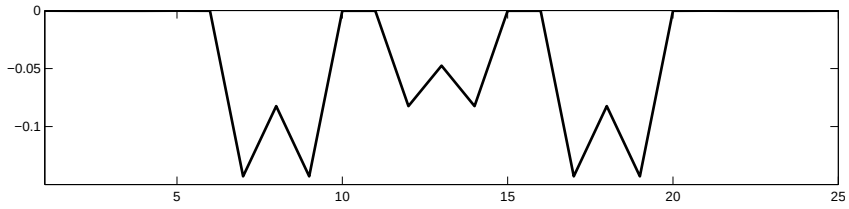


Figure 7: Exploded view of the subvector $\mathbf{sv}$ for an example of a doubly periodic domain with $3 \times 3$ subdomains, and $N = 4$.

Inspection of Fig. 6 shows a repetitive pattern/subvector of total size of $(N+1)^2$ elements, shown in detail in Fig. 7. This subvector is denoted as $\mathbf{sv}$, and it is repeated as many times as the number of subdomains ($nsub$) in the global domain (e.g. $nsub = 9$ for the case of $3 \times 3$ subdomains). A general definition of $\mathbf{sv}$ 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

$$\mathbf{u_0} = \frac{\mathbf{1}_{nsub} \otimes \mathbf{sv}}{\|\mathbf{1}_{nsub} \otimes \mathbf{sv}\|_2} \tag{52}$$

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 $\mathbf{sv}$ 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

$$\mathbf{u_0} = \frac{\mathbf{a} \otimes \mathbf{sv}}{\|\mathbf{a} \otimes \mathbf{sv}\|_2} \tag{53}$$

where $\mathbf{a}$ is a vector of $nsub$ elements, which contains the area of each subdomain.

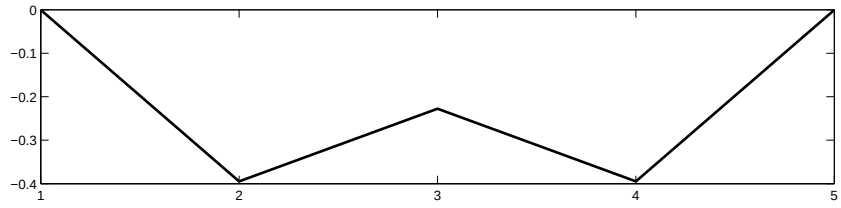


Figure 8: Structure of the vector $\mathbf{u_I}$ for the case of $N = 4$.

Further analysis applied to the vector $\mathbf{sv}$ (Fig.7) reveals an additional level of Kronecker product structure within it. As in Fig. 6, Fig. 7 also shows a repetitive pattern, denoted as $\mathbf{u_I}$ (see Fig. 8), which repeats itself every $N+1$ entries of $\mathbf{sv}$ with varying magnitude. Specifically, the vector $\mathbf{sv}$ can be constructed as

$$\mathbf{sv} = -\mathbf{u_I} \otimes \mathbf{u_I} \tag{54}$$

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 §7.3. Once the vector $\mathbf{sv}$ is computed, the global LNSV $\mathbf{u_0}$ is calculated using on Eq. (53) and (54) as

$$\mathbf{u_0} = -\frac{\mathbf{a} \otimes \mathbf{u_I} \otimes \mathbf{u_I}}{\|\mathbf{a} \otimes \mathbf{u_I} \otimes \mathbf{u_I}\|_2} \tag{55}$$

*7.2. 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 $3 \times 3$ subdomains and $N = 4$, is presented in Fig. 9. 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.

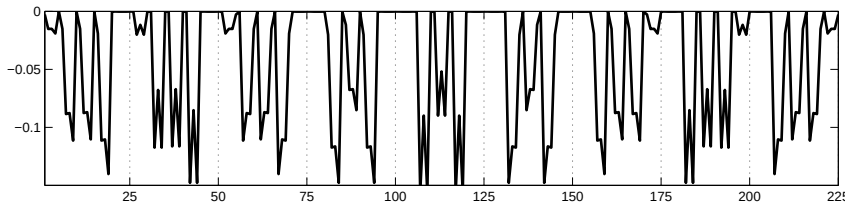


Figure 9: Structure of the of the left null singular vector $\mathbf{u_0}$ for an example non-periodic domain with $3 \times 3$ subdomains, and $N = 4$

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 $3 \times 3$ subdomains in the example set-up of Fig. 9). Nonetheless, there is not a clear repetitive pattern as in the doubly-periodic case, which means that no longer a simple Kronecker product, as the one used in Eq. (53), 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

$$\mathbf{u_0} = \frac{\left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right]^T}{\| \left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right] \|_2} \tag{56}$$

where the concatenation ($|$) operator is applied in a row-wise sense such that the numerator of Eq. (56) is a vector with the following structure (see Eq. (57) )

$$\left[\mathbf{sv}^{(1)}|\ldots|\mathbf{sv}^{(nsub)}\right]^T = \left[\underbrace{\mathbf{sv}_1^{(1)}\ldots\mathbf{sv}_{ns}^{(1)}}_{\mathbf{sv^{(1)}}}\underbrace{\mathbf{sv}_1^{(i)}\ldots\mathbf{sv}_{ns}^{(i)}}_{\mathbf{sv^{(i)}}}\underbrace{\mathbf{sv}_1^{(nsub)}\ldots\mathbf{sv}_{ns}^{(nsub)}}_{\mathbf{sv^{(nsub)}}}\right]^T . \tag{57}$$

In Eq. (56) and (57), $nsub$ 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. (54), the subvector $\mathbf{sv}^{(i)}$ corresponding to a subdomain $i$ is computed as

$$\mathbf{sv}^{(i)} = -a_i(\mathbf{u_z} \otimes \mathbf{u_x}) \tag{58}$$

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. 10 are given in Table 1. 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.

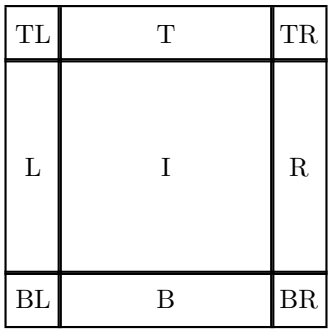


Figure 10: Schematic of a general non-periodic domain. It shows the possible locations of subdomain $i$ subject to different patching/boundary conditions.

| 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 1: 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. 10

### *7.3. 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. 11 shows a schematic of the global domain in which $\mathbf{L}, \mathbf{I}$ and $\mathbf{R}$ indicate the left, internal and right subdomain within it.

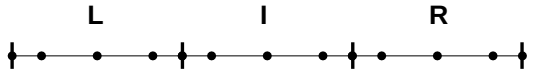


Figure 11: One dimensional base configuration for the generation of the left null singular vector $\mathbf{u_0}$ (case of $N+1=5$).

The corresponding SMPM Laplacian matrix $\mathbf{A}_{1d}$ for the one-dimensional set-up shown in Fig. 11 has dimension $3(N+1)\times 3(N+1)$ ($15\times 15$ in the example) and can be decomposed according to the SVD as:

$$\mathbf{A_{1d}} = \mathbf{U_{1d}}\mathbf{\Sigma_{1d}}\mathbf{V_{1d}^T} \tag{59}$$

where $\mathbf{U_{1d}}$, $\mathbf{V_{1d}}$ and $\mathbf{\Sigma_{1d}}$ are defined in the same way as in (31). 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. (60) and Fig. 12)

$$\left[\mathbf{u_0^{(1d)}}\right]^T = \left[\underbrace{u_0^{(1)}\ldots u_0^{(N+1)}}_{\mathbf{u_L}}\underbrace{u_0^{(N+2)}\ldots u_0^{(2N+2)}}_{\mathbf{u_I}}\underbrace{u_0^{(2N+3)}\ldots u_0^{(3N+3)}}_{\mathbf{u_R}}\right]^T . \tag{60}$$

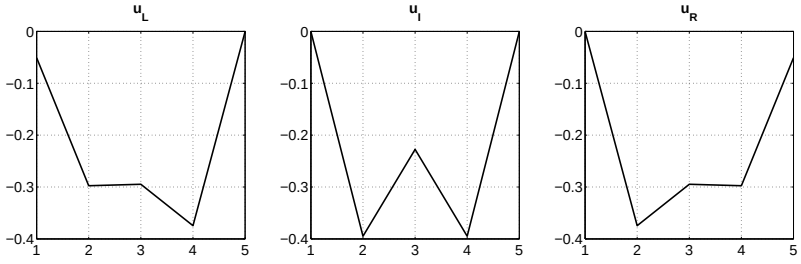


Figure 12: Left null singular vector structure of the one dimensional discrete SMPM Poisson matrix. $\mathbf{u_L}, \mathbf{u_I}$, and $\mathbf{u_R}$ for the case of $N+1=5$

In Eq. (60) and Fig. 12, 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. (55), 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. (56)-(58)). 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 2 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 2: 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 $\mathbf{u_T} = \mathbf{u_R}$

### *7.4. Validation of the procedure*

For the case of the 2D doubly-periodic domain, Fig. 13 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. 14. 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 13 and 14 it was established that the degree of

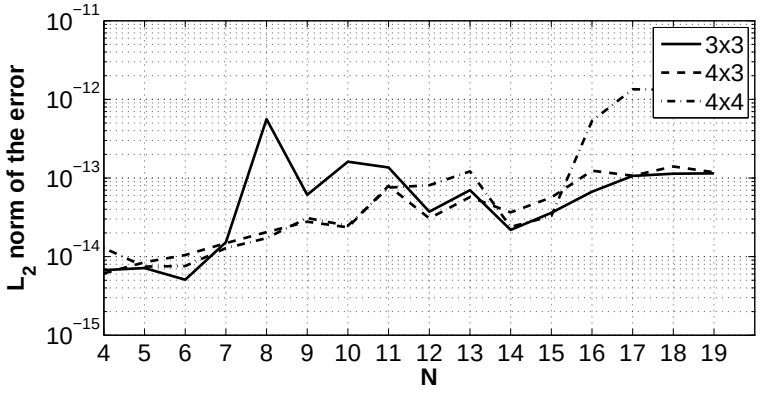


Figure 13: $L_2$ error norm (as compared to the corresponding MATLAB estimate) in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a doubly-periodic domain.

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. (57) and (58) 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. (21) and (22). Nonetheless, the exact nature of this connection remains to be established and is subject of future work.

## 8. 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 ($\mathbf{M}$) 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.

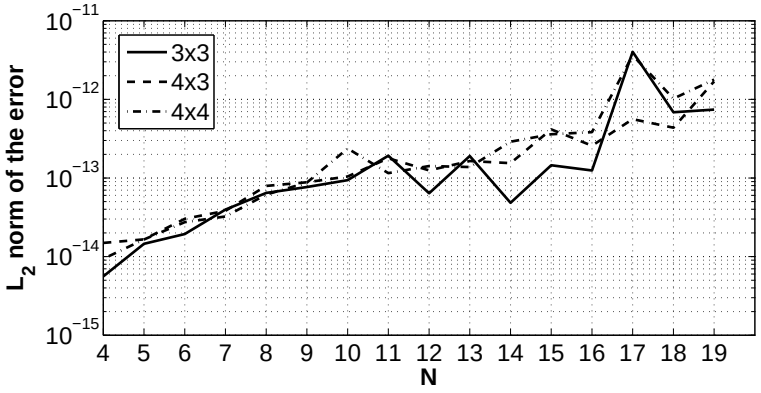


Figure 14: $L_2$ error norm in the computation of the LNSV for different number of subdomains and a varying number of the polynomial degree in a 2D non-periodic domain.

### *8.1. 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 [53].

$$m_{i,i} = a_{i,i} \tag{61}$$

As a first approximation for $\mathbf{M}$, 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.

### *8.2. Block Jacobi (non-overlapping additive Schwarz)*

The element-based character of the SMPM [3] furnishes a natural domain decomposition, which is reflected in the block structure of the Poisson matrix $\mathbf{A}$ (see §4 and Fig. 2), 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. (24) and (27), respectively). Under these considerations, we can construct a non-overlapping Schwarz preconditioner as

$$\mathbf{M}^{-1} = \sum_{k=1}^{nsub} \mathbf{R}_k^T \mathbf{A}_k^{-1} \mathbf{R}_k \tag{62}$$

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

[28]. 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 §9 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 independent 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. 19). 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.

*8.3. Two-Level preconditioner*

The implementation of this preconditioner draws from the previous work of Fischer and collaborators [27, 28] 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

$$\mathbf{M}^{-1} = \mathbf{R}_0^T \mathbf{A}_0^{-1} \mathbf{R}_0 + \sum_{k=1}^{nsub} \mathbf{R}_k^T \mathbf{A}_k^{-1} \mathbf{R}_k \tag{63}$$

Eq. (63) is effectively Eq. (62) 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 [54] 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. (33) 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 §9, the scalability of the two-level preconditioner is compared to that of the additive Schwarz (Block-Jacobi), and diagonal Jacobi.

## 9. Numerical results

### *9.1. 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. The flow field initially consists of a periodic array of vortices whose velocity field diffuses out with time. Periodic boundary conditions are imposed in both directions. The initial condition is based on the analytical solution [20, 5] obtained for $(x, y) \in [-1, 1]^2$:

$$u(t, x, y) = -\cos(\pi x)\sin(\pi y)\exp\left(\frac{-2\pi^2 t}{Re}\right) \tag{64}$$

$$w(t, x, y) = \sin(\pi x)\cos(\pi y)\exp\left(\frac{-2\pi^2 t}{Re}\right) \tag{65}$$

$$p(t, x, y) = -\frac{\cos(2\pi x) + \cos(2\pi y)}{4}\exp\left(\frac{-4\pi^2 t}{Re}\right) \tag{66}$$

Simulations are run over the time interval $0 \leq t \leq 1$ with a Reynolds number of $Re = U_m L/\nu = 100$, where $U_m$ is the magnitude of the maximum velocity on the domain, $L$ is the diameter of the vortex, and $\nu$ is the kinematic viscosity. The availability of the exact solution (64)-(66) enables an assessment of the accuracy of the numerical solution. Fig. 15 shows the convergence plot of the $L_\infty$-norm of the horizontal velocity as a function of polynomial degree (for a fixed number of subdomains, $p$-refinement), and Fig. 16 shows the same error norm as a function of the total number of degrees of freedom (DOF) ($hp-$refinement).

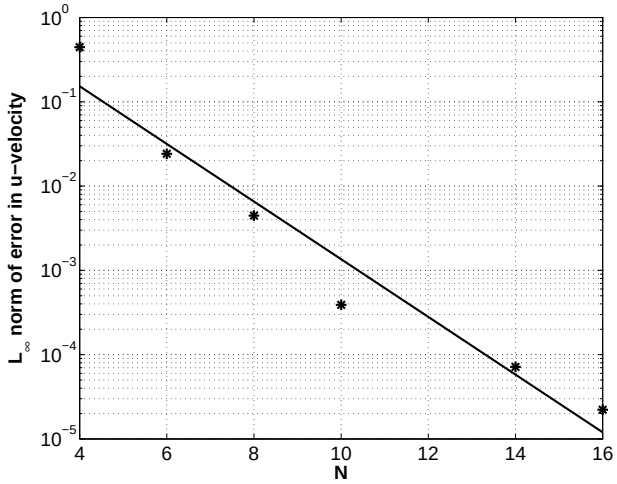


Figure 15: $L_\infty$-norm of the horizontal velocity for various polynomial degrees and a fixed number ($3 \times 3$ in this particular case)of subdomains. The solid line represents a least-squares exponential best-fit.

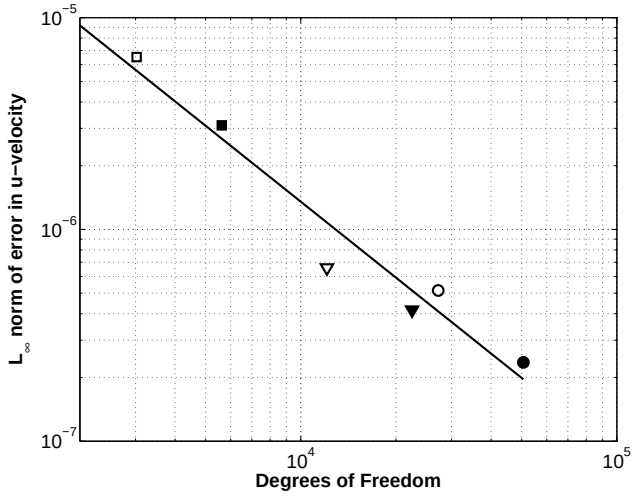


Figure 16: $L_\infty$-norm of the horizontal velocity for a varying number of total degrees or freedom (obtained through $hp-$refinement) for the Taylor Vortex problem at three different values of total number of subdomains. Square symbols represent $5 \times 5$ subdomains, triangles $10 \times 10$ subdomains, and circles $15 \times 15$ subdomains. For each particular number of total subdomains, open symbols represents $N = 10$, and filled symbols $N = 14$. The solid line represents a least-squares power-law best-fit.

The impact of the above discussed preconditioners on the efficiency of the

numerical solution of the PPE is shown in Fig. 17 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. 17 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. 17 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.

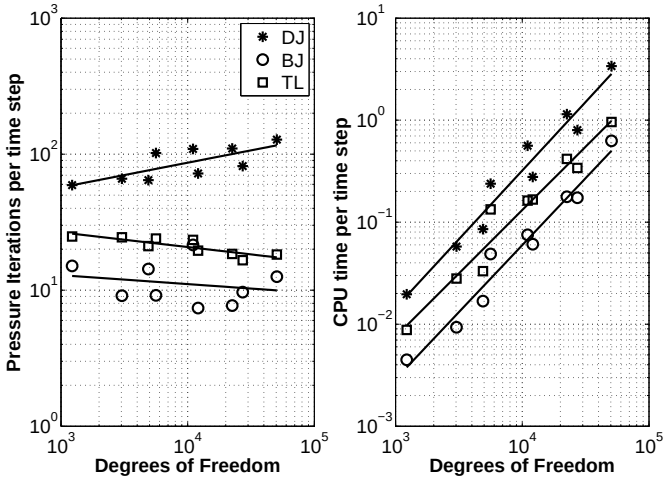


Figure 17: Poisson solver performance for different preconditioners for the Taylor vortex problem. Left panel shows the average number of pressure iterations per timestep as a function of total number of degrees of freedom (DOF). Right panel shows the average CPU time per timestep as a function of total number of degrees of freedom (DOF). In the legend, DJ, BJ and TL correspond to diagonal Jacobi, block-Jacobi and two-level preconditioners, respectively.

### *9.2. Lid-driven cavity flow*

The lid-driven cavity is a standard benchmark for testing the incompressible flow solvers with non-periodic boundary conditions [55, 56]. The computational domain is a square box ( $(x, y) \in [0, 1]^2$) forced by a top boundary translating at a steady velocity. A Reynolds number of $Re = UL/\nu = 10^3$ is considered, where $U$ represents the characteristic velocity at the top of the cavity, $L$ is the characteristic length of the box, and $\nu$ is the kinematic viscosity of the fluid. To

avoid the singularities that arise at the top corners due to discontinuities in the $u$ velocity [55, 57], we consider a modified lid-driven cavity [39], where the top boundary condition is given by

$$u(x,1) = -16x^2\left(1-x^2\right) \quad (67)$$
$$v(x,1) = 0 \quad (68)$$

The structure of the velocity field at steady state, shown in Fig. 18, agrees well with that corresponding to the existing spectral solution of Botella et.al.[55]. Viscous diffusion of vorticity 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. The smaller scale of the vortex at the left corner is likely to be an artifact of the modification to the top boundary condition described above.

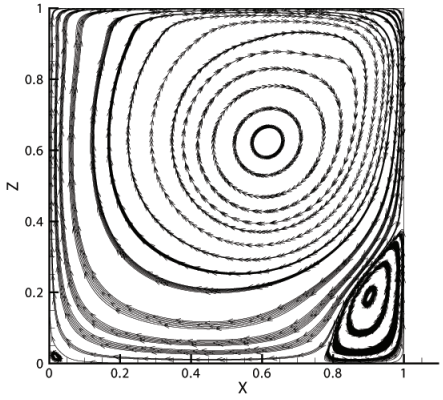


Figure 18: Streamlines on the Lid-driven cavity flow at $Re = 10^3$.

Fig. 19 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. 19 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. 19 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 §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.

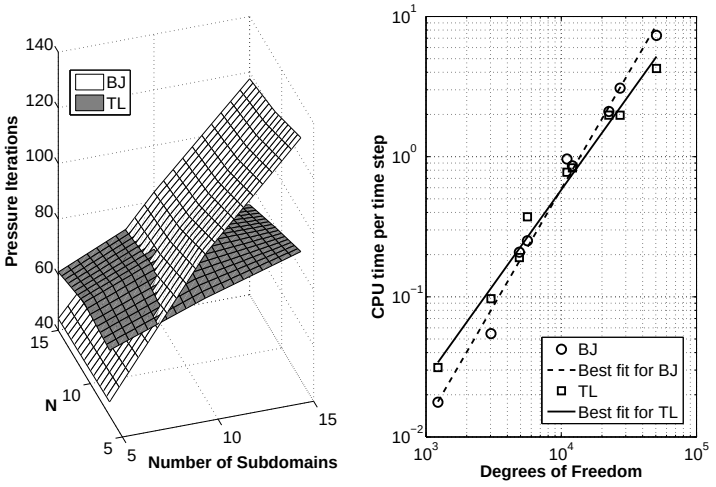


Figure 19: Poisson solver performance for the Block Jacobi (BJ) and Two-Level (TL) preconditioners applied to the Lid-driven Cavity flow. Left Panel: pressure iterations as a function of number of subdomains and polynomial degree $N$ (White surface: BJ preconditioner. Grey surface: TL preconditioner). 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.

## 10. 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 discontinuous 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 [53] 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 [18]. 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 [58, 35, 59, 60] 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, [61] 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 [61], 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.

## 11. Summary and concluding remarks

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.

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. 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 [62]. 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.

## Acknowledgements

We thank Prof. David Bindel for invaluable advice on the preconditioning strategies and the iterative solution of nearly-consistent systems. We also thanks Prof. Luke Olson for the discussions and guidance in the context of $p-$multigrid preconditioning. Dr. Paul Fischer is additionally thanked for illuminating discussions at the 2009 ICOSAHOM conference. The first author is grateful to Jose Maria Gonzalez Ondina for endless discussions on the subtleties of the PPE solver. The first two authors acknowledge the support of NSF CAREER Award OCE 0845558 .

## References

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

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

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

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

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

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

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

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

[9] J. A. Escobar-Vargas, P. J. Diamessis, 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. (2011).

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

[11] K. R. Helfrich, W. K. Melville, Long non-linear internal waves, Ann. Rev. Fluid Mech. 38 (2006) 395–425.

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

[13] Z. Zhang, O. B. Fringer, S. R. Ramp, Three-dimensional, nonhydrostatic numerical simulation of nonlinear internal wave generation and propagation in the South China Sea, J. Geophys. Res. 116 (2011).

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

[15] M. Restelli, F. X. Giraldo, A conservative discontinous Galerkin semi-implicit formulation for the Navier-Stokes equations in non-hydrostatic mesoscale atmospheric modeling, SIAM J. Sci. Comput. 31 (2009) 2231–2257.

[16] O. B. Fringer, M. Gerritsen, R. L. Street, An unstructured-grid, finite-volume, nonhydrostatic, parallel coastal ocean simulator, Ocean Modelling 14 (2006) 139–173.

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

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

[19] A. Scotti, S. Mitran, An approximated method for the solution of elliptic problems in thin domains: Application to nonlinear internal waves, Ocean Modelling 25 (2008) 144–153.

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

[21] A. Montlaur, High-order discontinuous Galerkin methods for incompressible flows, Ph.D. thesis, Universidad Politecnica de Calatuña, 2009.

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

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

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

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

[26] D. Gottlieb, C. L. Streett, Quadrature imposition of compatibility conditions in Chebyshev methods, J. Sci. Comput. 5 (1990) 223–239.

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

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

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

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

[31] B. Cushman-Roisin, Introduction to Geophysical Fluid Dynamics, Prentice Hall, New Jersey, 1994.

[32] P. M. Gresho, , R. L. Sani, Incompressible Flow and the Finite Element Method, volume 2, John Wiley and Sons, 1998.

[33] C. Xu, R. Pasquetti, Stabilized spectral element computations of high Reynolds number incompressible flows, J. Comput. Phys. 196 (2004) 680–704.

[34] W. Couzy, Spectral element discretization of the unsteady Navier-Stokes equations and its iterative solution on parallel computers, Ph.D. thesis, EPFL, 1995.

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

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

[37] T. M. Ozgokmen, T. Iliescu, P. F. Fischer, Large eddy simulation of stratified mixing in a three-dimensional lock-exchange system, Ocean Modeling 26 (2009) 134–1556.

[38] R. M. Kirby, G. E. Karniadakis, De-aliasing on non-uniform grids: algorithms and applications, J. Comput. Phys. 191 (2003) 249 – 264.

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

[40] J. L. Guermond, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Engrg 41 (2006) 112–134.

[41] L. Grinberg, D. Pekurovsky, S. J. Sherwin, G. E. Karniadakis, Parallel performance of the coarse space linear vertex solver and low energy basis preconditioner for spectral/hp elements, Parallel Comput. 35 (2009) 284–304.

[42] S. A. Orszag, M. Israeli, M. O. Deville, Boundary conditions for incompressible flows, J. Scient. Comput. 1 (1986) 75–111.

[43] P. M. Gresho, R. L. Sani, On pressure boundary conditions for the incompressible Navier-Stokes equations, Int. J. Numer. Methods Fluids 7 (1987) 1111–1145.

[44] B. Costa, W. S. Don, On the computation of high order pseudospectral derivatives, Appl. Num. Math. 33 (2000) 151–159.

[45] R. Baltensperger, M. R. Trummer, Spectral differencing with a twist, SIAM J. Sci. Comput. 24 (2002).

[46] G. Golub, C. F. Van Loan, Matrix Computations, Johns Hopkins University Press, third edition, 1996.

[47] C. F. Van Loan, The ubiquitous Kronecker product, Journal of Computational and Applied Mathematics 123 (2000) 85–100.

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

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

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

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

[52] J. A. C. Weideman, L. N. Trefethen, The eigenvalues of second order spectral differentiation matrices, SIAM J. Numer. Anal. 25 (1988) 1279–1298.

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

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

[55] O. Botella, R. Peyret, Benchmark spectral results on the lid-driven cavity flow, Computers and Fluids 27 (1998) 421–433.

[56] E. Erturk, T. C. Corke, C. Gokcol, Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers, Int. J. Num. Methods Fluids 48 (2005) 747–774.

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

[58] E. M. Rønquist, A. T. Patera, Spectral element multigrid. I. formulation and numerical results, J. Sci. Comput. 2 (1987) 389–406.

[59] K. J. Fidkowski, T. A. Oliver, J. Lu, D. L. Darmofal, p-multigrid solution of high-order discontinuous galerkin discretizations of the compressible navier-stokes equations, J. Comput. Phys. 207 (2005) 92–113.

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

[61] P. F. Fischer, Projection techniques for iterative solution of $Ax = b$ with succesive right-hand sides, Comput. Methods Appl. Mech. Engrg. 163 (1997) 193–204.

[62] L. Olson, Algebraic multigrid preconditioning of high-order spectral elements for elliptic problems on a simplicial mesh., SIAM J. Sci. Comput. 29 (2007) 2189–2209.