

SYMMETRY-PRESERVING DISCRETIZATION OF NAVIER-STOKES ON UNSTRUCTURED GRIDS: COLLOCATED VS STAGGERED

F.X.Trias¹, N.Valle¹, A.Gorobets² and A.Oliva¹

¹ Heat and Mass Transfer Technological Center,
Technical University of Catalonia, C/Colom 11, 08222 Terrassa (Barcelona)
{francesc.xavier.trias,nicolas.valle,asensio.oliva}@upc.edu

² Keldysh Institute of Applied Mathematics, 4A, Miusskaya Sq., Moscow 125047, Russia.
andrey.gorobets@gmail.com

Key words: Symmetry-preserving discretization, collocated mesh, staggered mesh, unstructured grids

Abstract. The essence of turbulence are the smallest scales of motion. They result from a subtle balance between convective transport and diffusive dissipation. Mathematically, these terms are governed by two differential operators differing in symmetry: the convective operator is skew-symmetric, whereas the diffusive is symmetric and positive-definite. On the other hand, accuracy and stability need to be reconciled for numerical simulations of turbulent flows around complex configurations. With this in mind, a fully-conservative discretization method for general unstructured grids was proposed [Trias *et al.*, J.Comp.Phys. 258, 246-267, 2014]: it exactly preserves the symmetries of the underlying differential operators on a collocated mesh. However, any pressure-correction method on collocated grids suffer from the same drawbacks: the cell-centered velocity field is not exactly incompressible and some artificial dissipation is inevitable introduced. On the other hand, for staggered velocity fields, the projection onto a divergence-free space is a well-posed problem: given a velocity field, it can be uniquely decomposed into a solenoidal vector and the gradient of a scalar (pressure) field. This can be easily done without introducing any dissipation as it should be from a physical point-of-view. In this work, we explore the possibility to build up staggered formulations based on collocated discrete operators.

1 INTRODUCTION

We consider the simulation of turbulent, incompressible flows of Newtonian fluids. Under these assumptions, the dimensionless governing equations in primitive variables read

$$\partial_t u + (u \cdot \nabla)u = \nu \nabla^2 u - \nabla p, \quad \nabla \cdot u = 0, \quad (1)$$

where u is the velocity field, p is the kinematic pressure and ν is the kinematic viscosity. The basic physical properties of the Navier-Stokes (NS) equations (1) can be deduced from the symmetries of the differential operators. In a discrete sense, it suffices to retain such operator symmetries to preserve the analogous (invariant) properties of the continuous equations [1]. However, for unstructured meshes, it is still a common argument that accuracy should take precedence over the properties of the operators. Contrary to this, our philosophy is that operator symmetries are critical to the dynamics of turbulence and

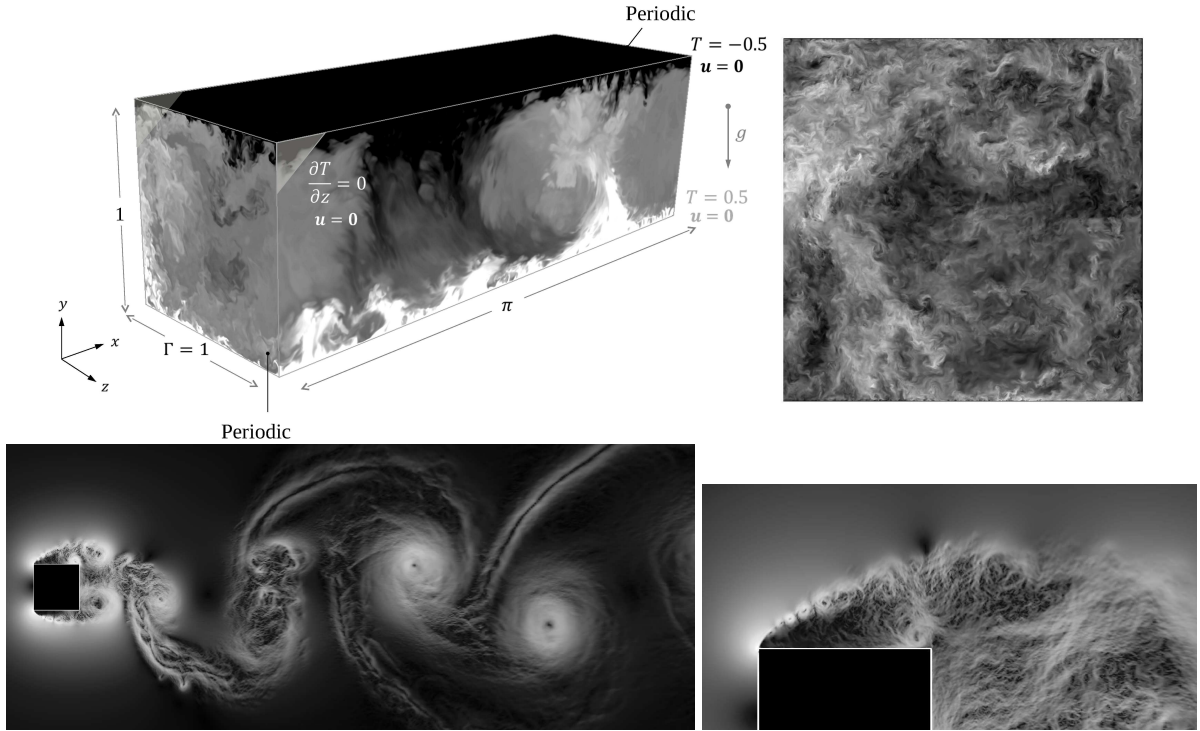


Figure 1: Examples of DNSs computed using symmetry-preserving discretizations. Top: air-filled ($Pr = 0.7$) Rayleigh-Bénard configuration studied in Ref. [3]. Instantaneous temperature field at $Ra = 10^{10}$ (left) and instantaneous velocity magnitude at $Ra = 10^{11}$ (right) for a span-wise cross section are shown. The latter was computed on 8192 CPU cores of the MareNostrum 4 supercomputer on a mesh of 5.7 billion grid points. Bottom: DNS of the turbulent flow around a square cylinder at $Re = 22000$ computed on 784 CPU cores of the MareNostrum 3 supercomputer on a mesh of 323 million grid points [4].

must be preserved [1, 2]. Namely, the convective operator is represented by a skew-symmetric coefficient matrix and the diffusive operator by a symmetric, positive-definite matrix. These ideas are briefly revised in the next section. Then, their extension to unstructured meshes is discussed in the following sections.

2 SYMMETRY-PRESERVING DISCRETIZATION

2.1 Starting point: staggered Cartesian meshes

The fully conservative discretization of the incompressible NS equations (1) is briefly described in this section. Otherwise stated, we follow the same operator-based notation than in Ref.[1]. The symmetry properties of the underlying differential operators are preserved: the convective operator is represented by a skew-symmetric matrix and the diffusive operator by a symmetric positive-definite matrix. In short, the temporal evolution of the spatially discrete staggered velocity vector, $u_s \in \mathbb{R}^m$, is governed by the following operator-based finite-volume discretization of Eqs.(1)

$$\Omega_s \frac{du_s}{dt} + C(u_s) u_s + Du_s - M^T p_c = 0_s, \quad Mu_s = 0_c, \quad (2)$$

where $p_c \in \mathbb{R}^n$ is the cell-centered pressure scalar field. The dimension of these vectors, n and m , are the number of control volumes and faces on the computational domain, respectively. The sub-indices c and s refer to whether the variables are cell-centered or staggered at the faces. The diffusive matrix, $D \in \mathbb{R}^{m \times m}$ represents the integral of the diffusive flux $-(\mu/\rho)\nabla u \cdot n$ through the faces. Like the underlying differential operator, $\nabla^2 = \nabla \cdot \nabla$, the diffusive operator consists of the product of a divergence matrix, $M_s \in \mathbb{R}^{m \times m}$, and a gradient matrix. The divergence is discretized and the discrete gradient becomes minus the transpose of the discrete divergence (multiplied by a diagonal scaling). This construction leads to a symmetric, positive-definite, approximation of the diffusive operator given by $D = \nu M_s \Omega_v^{-1} M_s^T$, where $\Omega_v \in \mathbb{R}^{m \times m}$ is a diagonal matrix containing the sizes of the control volumes associated with the faces of the velocity control volumes. This equation corresponds to Eq.(37) in [1]. For further details about the discretization of the diffusive operator the reader is referred to this work. The matrix $M \in \mathbb{R}^{n \times m}$ is the face-to-cell discrete divergence operator whereas the integral of the gradient operator is given by minus the transpose of M . The diagonal matrix, $\Omega_s \in \mathbb{R}^{m \times m}$, describes the sizes of the staggered control volumes and the approximate convective flux is discretized as in [1]. The resulting convective matrix, $C(u_s) \in \mathbb{R}^{m \times m}$, is skew-symmetric, *i.e.*

$$C(u_s) + C^T(u_s) = 0. \quad (3)$$

In a discrete setting, the skew-symmetry of $C(u_s)$ implies that

$$C(u_s) v_s \cdot w_s = v_s \cdot C^T(u_s) w_s = -v_s \cdot C(u_s) w_s, \quad (4)$$

for any discrete velocity vectors u_s (if $Mu_s = 0_c$), v_s and w_s . Then, the evolution of the discrete energy, $\|u_s\|^2 = u_s \cdot \Omega_s u_s$, is governed by

$$\frac{d}{dt} \|u_s\|^2 = -u_s \cdot (D + D^T) u_s \leq 0, \quad (5)$$

where the convective and pressure gradient contributions cancel because of Eq.(3) and the incompressibility constraint, $Mu_s = 0_c$, respectively. Therefore, even for coarse grids, the energy of the resolved scales of motion is convected in a stable manner, *i.e.* the discrete convective operator transports energy from a resolved scale of motion to other resolved scales without dissipating energy, as it should be from a physical point-of-view. It is noteworthy to mention that in the last decade, many DNS reference results have been successfully generated using this type of discretization (see Figure 1 and references therein).

2.2 Unstructured meshes. Collocated or staggered?

Accuracy and stability need to be reconciled for numerical simulations of turbulent flows around complex configurations. With this in mind, a fully-conservative discretization method for general unstructured grids was proposed in Ref. [2]: it exactly preserves the symmetries of the underlying differential operators on a collocated mesh. In summary, and following the same notation, the method is based on a set of five basic operators: the cell-centered and staggered control volumes (diagonal matrices), Ω_c and Ω_s , the matrix containing the face normal vectors, N_s , the cell-to-face scalar field interpolation, $\Pi_{c \rightarrow s}$ and the cell-to-face divergence operator, M . Once these operators are constructed, the rest follows straightforwardly from them. Therefore, the proposed method constitutes a robust and easy-to-implement approach to solve incompressible turbulent flows in complex configurations that can be implemented in already

existing codes such as OpenFOAM[®] [5]. However, any pressure-correction method on collocated grids suffer from the same drawbacks: the cell-centered velocity field is not exactly incompressible and some artificial dissipation is inevitable introduced. Conversely, the projection of a staggered velocity onto a divergence-free space is a well-posed problem: it can be uniquely decomposed into a solenoidal vector and the gradient of a scalar (pressure) field. This can be easily done without introducing any dissipation as it should be from a physical point-of-view. This is carefully analyzed in the next section.

3 PRESSURE-VELOCITY COUPLING: A UNIFIED FRAMEWORK

Roughly speaking, algorithms to solve the pressure-velocity coupling can be grouped into two families. Namely, on one hand we find all the existing variants of the SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm originally proposed by Patankar and Spalding [6], such as the SIMPLEC [7], SIMPLER [8] or PISO [9] among many others (the reader is referred to Ref.[10] for details about all this family of SIMPLE-like algorithms). On the other hand, we have the so-called Fractional Step Method (FSM) algorithm originally proposed by Chorin [11] and Teman [12]. SIMPLE-like algorithms were originally intended to solve problems using an implicit time-integration for the non-linear convective term. This implied a linearization of this term that is iteratively solved inside the SIMPLE iterations. Therefore, this family of algorithms are typically used for the solution of laminar flow problems or RANS (also URANS) problems where time-steps significantly larger than the CFL condition can be used. However, for highly unsteady problems such as DNS/LES of turbulent flows the time-step must be necessarily small (typically order of the CFL condition); therefore, the implicit treatment of the convective term becomes inefficient and it is usually replaced by a much simpler explicit treatment. In this case, the pressure-velocity coupling can be solved with one single iteration of the FSM. Following the philosophy of the work by Perot [13], all these different approaches can be viewed as an (approximate) LU decomposition of the discrete system. Namely,

$$\begin{pmatrix} \mathcal{A} & \Omega \mathcal{G} \\ \mathcal{M} & 0 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ \tilde{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \Omega u^n \\ 0 \end{pmatrix} + \begin{pmatrix} r \\ 0 \end{pmatrix}, \quad (6)$$

where Ω is a (diagonal) matrix that contains the size of control volumes associates with the discrete velocity field, u , \mathcal{G} is the discrete gradient operator and \mathcal{M} is the integrated discrete divergence operator. For convenience, and without loss of generality, both momentum and mass equation are written in integral form. Matrix \mathcal{A} and vector r will depend on the particular choice for the spatial and temporal discretization. For instance, for the finite-volume staggered discretization of the Navier-Stokes equations given in Eq.(2) in conjunction with a second-order Adams-Bashforth scheme for the time-integration of the convective and diffusive terms they read

$$\mathcal{A} = \Omega_s; \quad \Omega = \Omega_s; \quad \mathcal{G} = -\Omega_s^{-1} \mathbf{M}^T; \quad \mathcal{M} = \mathbf{M}, \quad (7a)$$

$$u^{n+1} = u_s^{n+1}; \quad u^n = u_s^n; \quad \tilde{p}^{n+1} = \Delta t p_c^{n+1}, \quad (7b)$$

$$r = \frac{3}{2} \mathbf{R}(u_s^n) - \frac{1}{3} \mathbf{R}(u_s^{n-1}), \quad (7c)$$

where $\mathbf{R}(u_s) \equiv -\mathbf{C}(u_s)u_s - \mathbf{D}u_s$ and Δt is the time-step. Omitting the fact that the (pseudo-)pressure, \tilde{p} , is indefinite, the linear system given in Eq.(6) can be solved via a block LU decomposition

$$\begin{pmatrix} \mathcal{A} & \Omega \mathcal{G} \\ \mathcal{M} & 0 \end{pmatrix} = \begin{pmatrix} \mathcal{A} & 0 \\ \mathcal{M} & -\mathcal{L} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathcal{A}^{-1} \Omega \mathcal{G} \\ 0 & \mathbf{I} \end{pmatrix} \quad \text{where } \mathcal{L} = \mathcal{M} \mathcal{A}^{-1} \Omega \mathcal{G}, \quad (8)$$

where $-\mathcal{L}$ is the Schur complement matrix. Assuming that this block LU decomposition can be inverted, the original system given in Eq.(6) is solved in two steps:

$$\begin{pmatrix} \mathcal{A} & 0 \\ \mathcal{M} & -\mathcal{L} \end{pmatrix} \begin{pmatrix} u^* \\ \tilde{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \Omega u^n \\ 0 \end{pmatrix} + \begin{pmatrix} r \\ 0 \end{pmatrix} \quad (9a)$$

$$\begin{pmatrix} 1 & \mathcal{A}^{-1}\Omega\mathcal{G} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ \tilde{p}^{n+1} \end{pmatrix} = \begin{pmatrix} u^* \\ \tilde{p}^{n+1} \end{pmatrix}, \quad (9b)$$

where u^* is an intermediate velocity. Notice that for the particular choice given in Eqs.(7) this LU decomposition becomes

$$\begin{pmatrix} \Omega_s & 0 \\ \mathbf{M} & -\mathbf{L} \end{pmatrix} \begin{pmatrix} u_s^p \\ \tilde{p}_c^{n+1} \end{pmatrix} = \begin{pmatrix} \Omega_s u^n \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{3}{2}\mathbf{R}(u_s^n) - \frac{1}{3}\mathbf{R}(u_s^{n-1}) \\ 0 \end{pmatrix} \quad (10a)$$

$$\begin{pmatrix} 1 & \mathbf{G} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u_s^{n+1} \\ \tilde{p}_c^{n+1} \end{pmatrix} = \begin{pmatrix} u_s^p \\ \tilde{p}_c^{n+1} \end{pmatrix}, \quad (10b)$$

where $\mathbf{L} = \mathbf{M}\mathbf{G} = -\mathbf{M}\Omega_s^{-1}\mathbf{M}^T$ is the standard discrete Laplacian operator and u_s^p is a predictor velocity. Solving this block LU decomposition is exactly the same than the classical Fractional Step Method, *i.e.*

$$u_s^p = u_s^n + \Omega_s^{-1} \left(\frac{3}{2}\mathbf{R}(u_s^n) - \frac{1}{3}\mathbf{R}(u_s^{n-1}) \right), \quad (11a)$$

$$\tilde{p}_c^{n+1} = \mathbf{L}^{-1}\mathbf{M}u_s^p, \quad (11b)$$

$$u_s^{n+1} = u_s^p - \mathbf{G}\tilde{p}_c^{n+1}. \quad (11c)$$

As mentioned above, solving the pressure-velocity coupling on a staggered mesh (with an explicit time-integration scheme) is a well-posed problem. Difficulties arise for other set-ups. For instance, a collocated arrangement of variables leads to a discrete Laplacian operator with numerical pathologies (the so-called checkerboard problem, *i.e.* nonphysical components of pressure field belong to its kernel). Furthermore, the implicit treatment of the convective or/and diffusive term leads to non-diagonal forms of the matrix \mathcal{A} ; therefore, its exact inversion, \mathcal{A}^{-1} , becomes impractical especially for the construction of the Laplacian operator, $\mathcal{L} = \mathcal{M}\mathcal{A}^{-1}\Omega\mathcal{G}$, and its subsequent inversion. A similar problem arises for incompressible multiphase flows, even for fully-explicit time-integration schemes. In this case, the matrix \mathcal{A} is diagonal but it changes over time due to the density variations. This leads to a discrete Laplacian operator with non-constant (in time) coefficients. In general, these situations are usually tackled with an approximation of the exact LU decomposition given in Eq.(8)

$$\mathcal{F} = \mathcal{L}_{\mathcal{F}} \mathcal{U}_{\mathcal{F}} \quad \text{where} \quad \mathcal{F} = \begin{pmatrix} \mathcal{A} & \Omega\mathcal{G} \\ \mathcal{M} & 0 \end{pmatrix}, \quad \mathcal{L}_{\mathcal{F}} = \begin{pmatrix} \mathcal{A} & 0 \\ \mathcal{M} & -\mathcal{L} \end{pmatrix}, \quad \mathcal{U}_{\mathcal{F}} = \begin{pmatrix} 1 & \mathcal{A}^{-1}\Omega\mathcal{G} \\ 0 & 1 \end{pmatrix}. \quad (12)$$

For instance, in the case of the SIMPLE algorithm (also some variants) the \mathcal{A}^{-1} is approximated by a diagonal matrix, $\tilde{\mathcal{A}}^{-1}$. This leads to the following approximation of the LU decomposition

$$\tilde{\mathcal{F}} = \tilde{\mathcal{L}}_{\mathcal{F}} \tilde{\mathcal{U}}_{\mathcal{F}} \quad \text{where} \quad \tilde{\mathcal{F}} = \begin{pmatrix} \mathcal{A} & \mathcal{A}\tilde{\mathcal{A}}^{-1}\Omega\mathcal{G} \\ \mathcal{M} & 0 \end{pmatrix}, \quad \tilde{\mathcal{L}}_{\mathcal{F}} = \begin{pmatrix} \mathcal{A} & 0 \\ \mathcal{M} & -\tilde{\mathcal{L}} \end{pmatrix}, \quad \tilde{\mathcal{U}}_{\mathcal{F}} = \begin{pmatrix} 1 & \tilde{\mathcal{A}}^{-1}\Omega\mathcal{G} \\ 0 & 1 \end{pmatrix}, \quad (13)$$

Algorithm	Time int.	Basic set of discrete operators					Approximations (if needed)	
		\mathcal{A}	Ω	\mathcal{G}	\mathcal{M}	\mathcal{L}	$\tilde{\mathcal{L}}$	$\tilde{\mathcal{A}}$
SIMPLE	Impl.	\mathcal{A}	Ω	\mathcal{G}	\mathcal{M}	$\mathcal{M}\mathcal{A}^{-1}\mathcal{G}$	$\mathcal{M}\tilde{\mathcal{A}}^{-1}\mathcal{G}$	$diag(\mathcal{A})$
SIMPLEC	Impl.	\mathcal{A}	Ω	\mathcal{G}	\mathcal{M}	$\mathcal{M}\mathcal{A}^{-1}\mathcal{G}$	$\mathcal{M}\tilde{\mathcal{A}}^{-1}\mathcal{G}$	$[\tilde{\mathcal{A}}]_{ii} = \sum_j [\mathcal{A}]_{ij} $
Stg FSM	Expl.	Ω_s	Ω_s	\mathcal{G}	\mathcal{M}	$\mathcal{L} = \mathcal{M}\mathcal{G}$	–	–
Stg FSM 2-fluid	Expl.	$\mathcal{P}_s\Omega_s$	Ω_s	\mathcal{G}	\mathcal{M}	$\mathcal{M}\mathcal{P}_s^{-1}\mathcal{G}$	$\rho_0^{-1}\mathcal{L}$	–
Col FSM	Expl.	Ω_c	Ω_c	\mathcal{G}_c	\mathcal{M}_c	$\mathcal{L}_c = \mathcal{M}_c\mathcal{G}_c$	\mathcal{L}	–

Table 1: Summary of the pressure-velocity coupling algorithms outlined in this paper.

where the approximate discrete Laplacian is given by $\tilde{\mathcal{L}} = \mathcal{M}\tilde{\mathcal{A}}^{-1}\Omega\mathcal{G}$. Then, the SIMPLE algorithm can be viewed as a stationary iterative solver [14]

$$\tilde{\mathcal{F}}x^{k+1} = b + (\tilde{\mathcal{F}} - \mathcal{F})x^k \quad \longrightarrow \quad x^{k+1} = \tilde{\mathcal{F}}^{-1}b + (1 - \tilde{\mathcal{F}}^{-1}\mathcal{F})x^k \quad \text{where } x = \begin{pmatrix} u \\ \tilde{p} \end{pmatrix}, \quad (14)$$

where k refers to the outer loop iterations of the SIMPLE algorithm. In the original SIMPLE algorithm [6], the diagonal matrix $\tilde{\mathcal{A}}$ is given by the diagonal elements of \mathcal{A} , *i.e.* $[\tilde{\mathcal{A}}^{SIMPLE}]_{ij} = [\mathcal{A}]_{ij}\delta_{ij}$, where δ_{ij} is the Kronecker delta. However, the stability of this iterative procedure is only guaranteed if the spectral radius of the associated transfer function, $1 - \tilde{\mathcal{F}}^{-1}\mathcal{F}$, is smaller or equal to one,

$$\rho(1 - \tilde{\mathcal{F}}^{-1}\mathcal{F}) \leq 1, \quad (15)$$

which is not always satisfied in the original SIMPLE algorithm. This motivated the use of under-relaxation factors (for pressure) to alleviate this problem [8]. Other interesting approaches to tackle these stability issues are the subsequent modifications of the SIMPLE algorithm, such as the SIMPLE Consistent (SIMPLEC) algorithm [7] where the diagonal matrix $\tilde{\mathcal{A}}$ is given by sum by rows of the elements of \mathcal{A} in absolute value, *i.e.* $[\tilde{\mathcal{A}}^{SIMPLEC}]_{ii} = \sum_j |[\mathcal{A}]_{ij}|$.

As mentioned above, approximations of \mathcal{A} may be needed for incompressible multiphase problems. Let's assume that we use the same staggered formulation in conjunction with a second-order Adams-Bashforth scheme that, for the single phase case, led to the LU decomposition given in Eq.(10). However, for incompressible multiphase flows with interface tracking, density is not constant [15]. This necessarily leads to a modification of the block system of equations. Namely,

$$F_s^P \equiv \begin{pmatrix} P_s\Omega_s & \Omega_s\mathcal{G} \\ \mathcal{M} & 0 \end{pmatrix} = \begin{pmatrix} P_s\Omega_s & 0 \\ \mathcal{M} & -L_P \end{pmatrix} \begin{pmatrix} I & P_s^{-1}\mathcal{G} \\ 0 & I \end{pmatrix}, \quad (16)$$

where $P_s \in \mathbb{R}^{m \times m}$ is a diagonal matrix that contains the density defined at the faces. Although this system is well-posed, the discrete Laplacian operator, $L_P \equiv \mathcal{M}\mathcal{P}_s^{-1}\mathcal{G} = -\mathcal{M}\mathcal{P}_s^{-1}\Omega_s^{-1}\mathcal{M}^T$, becomes very ill-conditioned for high-density ratios and, even worse, the matrix L_P is changing every time-step due to the movement of the interface, *i.e.* $P_s(t)$. This may require efficient and easy-to-built preconditioning techniques [16] or/and approximations that rely on the Laplacian operator with constant coefficients [17]. In the latter case, L_P is replaced by $\rho_0^{-1}\mathcal{L}$ where ρ_0 is the (constant) density corresponding to the light phase. However, this type of modification of the LU decomposition leads to an error in the mass conservation equation

$$\tilde{F}_s^P \equiv \begin{pmatrix} P_s\Omega_s & \Omega_s\mathcal{G} \\ \mathcal{M} & L_P - \rho_0^{-1}\mathcal{L} \end{pmatrix} = \begin{pmatrix} P_s\Omega_s & 0 \\ \mathcal{M} & -\rho_0^{-1}\mathcal{L} \end{pmatrix} \begin{pmatrix} I & P_s^{-1}\mathcal{G} \\ 0 & I \end{pmatrix}, \quad (17)$$

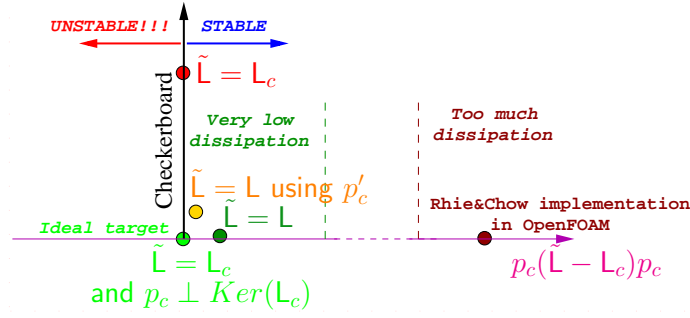


Figure 2: Schematic summary of existing pressure-velocity coupling approaches. Horizontal axis represents the amount of artificial dissipation introduced by the pressure gradient term in the momentum equation whereas vertical axis represents the appearance of checkerboard modes in the numerical solutions.

that will be given by $Mu_s^{n+1} = -(L_P - \rho_0^{-1}L)\tilde{p}_c^{n+1} \neq 0_c$. This error can be significantly reduced providing an approximation of the pressure field, \tilde{p}_c^p , in the right-hand-side of the system,

$$\begin{pmatrix} P_s \Omega_s & \Omega_s G \\ M & L_P - \rho_0^{-1}L \end{pmatrix} \begin{pmatrix} u_s^{n+1} \\ \tilde{p}_c^p \end{pmatrix} = \begin{pmatrix} P_s \Omega_s u_s^n + r_s \\ 0_c \end{pmatrix} + \begin{pmatrix} 0 & -\Omega_s G \\ 0 & 0 \end{pmatrix} \begin{pmatrix} u_s^n \\ \tilde{p}_c^p \end{pmatrix}, \quad (18)$$

where $\tilde{p}_c^p = \tilde{p}_c^{n+1} - \tilde{p}_c^p$ is a (pseudo)pressure correction. Likewise for the SIMPLE algorithm, this algorithm can also be viewed as a stationary iterative solver (with only one iteration per time-step)

$$\tilde{F}_s^P x^{n+1} = b + (\tilde{F}_s^P - F_s^P) x^n \quad \longrightarrow \quad x^{n+1} = (\tilde{F}_s^P)^{-1} b + (1 - (\tilde{F}_s^P)^{-1} F_s^P) x^n \quad (19a)$$

$$\text{where } x = \begin{pmatrix} u_s \\ \tilde{p}_c \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} P_s \Omega_s u_s + r_s \\ 0_c \end{pmatrix}. \quad (19b)$$

Hence, once again the stability of the algorithm will depend on the spectral radius of the transfer function, $1 - (\tilde{F}_s^P)^{-1} F_s^P$. The particular choice $L_P \rightarrow \rho_0^{-1}L$ proposed by the authors [17] can be justified in this context.

A very similar problem exists from collocated formulations. In this case, the discrete Laplacian, L_c , that arises from the LU decomposition of the system

$$F_c \equiv \mathcal{L}_{F_c} \mathcal{U}_{F_c} \quad \text{where} \quad F_c = \begin{pmatrix} \Omega_c & \Omega_c G_c \\ M_c & 0 \end{pmatrix}, \quad \mathcal{L}_{F_c} = \begin{pmatrix} \Omega_c & 0 \\ M & -L_c \end{pmatrix}, \quad \mathcal{U}_{F_c} = \begin{pmatrix} I & G_c \\ 0 & I \end{pmatrix}, \quad (20)$$

reads $L_c = M_c G_c$ where $M_c \equiv M \Gamma_{c \rightarrow s}$ and $G_c \equiv \Gamma_{s \rightarrow c} G$ are respectively the collocated divergence and the gradient operators. These are based on the already-defined face-to-cell divergence operator, M , and the cell-to-face gradient operator, G , in conjunction with a cell-to-face, $\Gamma_{c \rightarrow s}$, and a face-to-cell, $\Gamma_{s \rightarrow c}$, interpolations. For details about the construction of these discrete operators the reader is referred to [2]. The collocated Laplacian operator contains nonphysical pressure components in its kernel, *i.e.* the well-known checkerboard problem. Traditional cure for this, especially in the context of the above-explained SIMPLE-like algorithm, is the pressure-weighted interpolation (PWI) approach proposed by Rhie and Chow [18]. In the context of the FSM, a very similar approach appear 'naturally' replacing the wide-stencil collocated Laplacian operator, L_c , by the compact one, L . In this case, the FSM method for

collocated meshes reads

$$\tilde{F}_c \equiv \mathcal{L}_{\tilde{F}_c} \mathcal{U}_{\tilde{F}_c} \quad \text{where} \quad \tilde{F}_c = \begin{pmatrix} \Omega_c & \Omega_c G_c \\ M_c & L_c - L \end{pmatrix}, \quad \mathcal{L}_{\tilde{F}_c} = \begin{pmatrix} \Omega_c & 0 \\ M & -L \end{pmatrix}, \quad \mathcal{U}_{\tilde{F}_c} = \begin{pmatrix} 1 & G_c \\ 0 & 1 \end{pmatrix}. \quad (21)$$

Once again, this implies that the method has an intrinsic error in the mass conservation equation given by $M_c u_c^{n+1} = -(L_c - L) \tilde{p}_c^{n+1} \neq 0$. Even more important, the method has an error in terms of kinetic energy given by $\tilde{p}_c^{n+1} (L_c - L) \tilde{p}_c^{n+1} \neq 0$. This error can have severe implications for DNS/LES simulations of turbulent flows since this artificial dissipation can significantly affect the dynamics of the small scales even overwhelming the dissipation introduced by the subgrid-scale LES models. This was clearly observed for LES simulations using the standard implementation of OpenFOAM[®] [19]. In this regard, Figure 3 schematically represents some of the exiting pressure-velocity coupling algorithms. It is worth to notice that the ideal target, *i.e.* no artificial dissipation and no checkerboard, can be achieved by explicitly removing those nonphysical components of the pressure field that belong to the kernel of L_c . This was proposed and successfully tested in [20] for Cartesian meshes where these nonphysical pressure modes are known *a priori*. In theory, this approach is also applicable to unstructured meshes; however, for practical problems it becomes far too expensive to compute. Alternatively, it is possible to minimize the amount of dissipation while still keeping the solution virtually free of checkerboard modes by preserving the symmetries of the discrete operators [2]. The unified framework presented here facilitates this type of analysis. Namely, the source and type of errors become more clear. For instance, for collocated formulations there is an unavoidable error in terms of mass, $M_c u_c^{n+1} = -(L_c - L) \tilde{p}_c^{n+1} \neq 0$ and kinetic energy, $\tilde{p}_c^{n+1} (L_c - L) \tilde{p}_c^{n+1} \neq 0$ due to the approximation introduced in the Schur complement matrix, $L_c \approx L$. Since we cannot avoid this (L_c has non-physical pressure modes in its kernel) we can try to do our best. For instance, we can minimize physical dissipation adding a pressure prediction, \tilde{p}_c^p in the right-hand-side term likewise in Eq.(18) leading to mass and kinetic energy errors in terms of pressure correction, \tilde{p}_c' . If pressure prediction is taken to $\tilde{p}_c^p = \tilde{p}_c^n$, *i.e.* second-order Van Kan projection method [21], then the error in kinetic energy becomes $O(\Delta t^4)$ instead of $O(\Delta t^2)$. This framework also allows to easily analyze the stability of the method: likewise Eq.(19), it can be easily shown that the stability relies on the transfer function, $1 - \tilde{F}_c^{-1} F_c$ where F_c and \tilde{F}_c are respectively defined in Eqs.(20) and (21). The analysis can be simplified noticing that $\mathcal{U}_{\tilde{F}_c} = \mathcal{U}_{F_c}$,

$$1 - \tilde{F}_c^{-1} F_c = 1 - (\mathcal{L}_{\tilde{F}_c} \mathcal{U}_{\tilde{F}_c})^{-1} \mathcal{L}_{F_c} \mathcal{U}_{F_c} \stackrel{\mathcal{U}_{\tilde{F}_c} = \mathcal{U}_{F_c}}{=} \mathcal{U}_{F_c}^{-1} (1 - \mathcal{L}_{\tilde{F}_c}^{-1} \mathcal{L}_{F_c}) \mathcal{U}_{F_c}. \quad (22)$$

Hence, the stability of the method finally relies on the matrix $1 - \mathcal{L}_{\tilde{F}_c}^{-1} \mathcal{L}_{F_c}$. This can be simplified further plugging the definitions of \mathcal{L}_{F_c} and $\mathcal{L}_{\tilde{F}_c}$ given in Eqs.(20) and (21), respectively,

$$1 - \mathcal{L}_{\tilde{F}_c}^{-1} \mathcal{L}_{F_c} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} \Omega_c & 0 \\ M & -L \end{pmatrix}^{-1} \begin{pmatrix} \Omega_c & 0 \\ M & -L_c \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 - L^{-1} L_c \end{pmatrix}. \quad (23)$$

Therefore, at the end, the stability of the method will only depend on the spectrum of the matrix $L^{-1} L_c$. This has been carefully analyzed in a companion paper [22] where restrictions on the interpolation operators used to construct the wide-stencil Laplacian, L_c , have been imposed to guarantee the stability of the method for any mesh. This type of approaches allows to compute DNS/LES simulations on complex geometries with a very small amount of artificial dissipation [2, 5]. Apparently we cannot get rid of this dissipation unless we use staggered formulations where the pressure-velocity coupled is a well-posed problem. However, in this case, the numerical challenge to be addressed in the discretization of the momentum equation on unstructured grid. This is briefly discussed in the next section.

4 TOWARDS A STAGGERED DISCRETIZATION BASED ON COLLOCATED OPERATORS

Finite-volume collocated discretizations on unstructured meshes is the solution adopted by most of the general-purpose CFD codes such as ANSYS-FLUENT[®], OpenFOAM[®], etc. Despite the above-explained intrinsic errors due to the improper pressure-velocity coupling, this approach is usually preferred over a staggered one due to its simplicity to discretize momentum equations on unstructured grids. Attempts to construct staggered formulations goes back to the work by Perot [23] where NS equations are discretized in rotational form which implies to compute vorticity, $\omega = \nabla \times u$, at the edges of the mesh. The easier alternative to combine collocated discrete operators was already suggested in the same work [23] and subsequently explored by other researchers [24]. For a number of reasons, at the end, the winner seems to be the collocated arrangement. This can be compactly written as follows

$$\tilde{F}_c x_c^{n+1} = T_c x_c^n \quad \text{where} \quad T_c = \begin{pmatrix} \Omega_c + \partial_t^c & M^T \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad x_c = \begin{pmatrix} u_c \\ \tilde{p}_c \end{pmatrix}, \quad (24)$$

and \tilde{F}_c is given in Eq.(21) and ∂_t^c represents all the terms (except pressure gradient) that contributes to the time-derivative of the collocated velocity field, u_c , in the momentum equation, *i.e.* convective and diffusive terms (plus any other extra term in the momentum equation). Notice that the transpose of the divergence, M^T , is inside T_c to include the above-explained pressure-correction approach, *i.e.* second-order Van Kan projection [21]. In this way, the one time-step iteration to numerically solve the NS equations on a collocated grid would finally read

$$NS_c \equiv \tilde{F}_c^{-1} T_c : x_c^n \longrightarrow x_c^{n+1}. \quad (25)$$

On the other hand, the staggered counterpart should be

$$NS_s \equiv F_s^{-1} T_s : x_s^n \longrightarrow x_s^{n+1}, \quad (26)$$

where F_s represents the well-posed FSM method for staggered variables defined in Eqs.(10). At this point, the difficulty arises in the construction of the discrete operator T_s that should compute the momentum equation (except the gradient of pressure) for the staggered velocity field, u_s ,

$$T_s = \begin{pmatrix} \Omega_s + \partial_t^s & 0 \\ 0 & 0 \end{pmatrix}. \quad (27)$$

In this case, the transpose of the divergence, M^T , is not included since the pressure-velocity coupling is well-posed. Its inclusion would not change the solution of the NS_s mapping. Hence, at the end, everything depends on the way ∂_t^s is computed. Here, as a preliminary study we have studied the effect of changing the way the convective term (inside ∂_t^s) is constructed. For the sake of simplicity, we have only considered implementations that are based on already implemented operators in any standard CFD collocated code. First attempt consists on interpolating the staggered velocity field to cells, compute ∂_t^c , and then interpolate it back to the faces,

$$\partial_t^s = \Gamma_{s \rightarrow c} \partial_t^c \Gamma_{c \rightarrow s}, \quad (28)$$

where $\Gamma_{c \rightarrow s}$ and $\Gamma_{s \rightarrow c}$ are cell-to-face and face-to-cell interpolations which must be related via

$$\Gamma_{s \rightarrow c} = \Omega_c^{-1} \Gamma_{c \rightarrow s}^T \Omega_s, \quad (29)$$

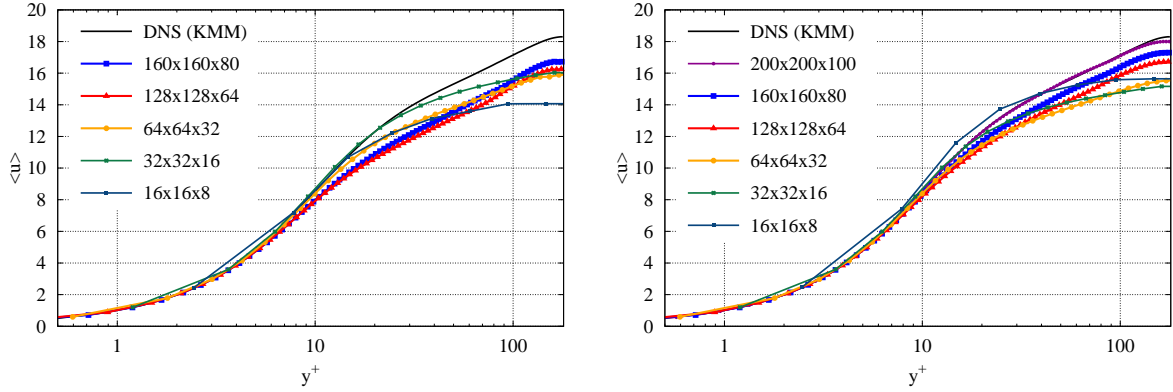


Figure 3: Results for a turbulent channel flow at $Re_\tau = 180$ for different meshes compared with the DNS [25]. Left: staggered approach proposed in Eq.(28). Right: staggered approach proposed in Eq.(30) with $n = 2$.

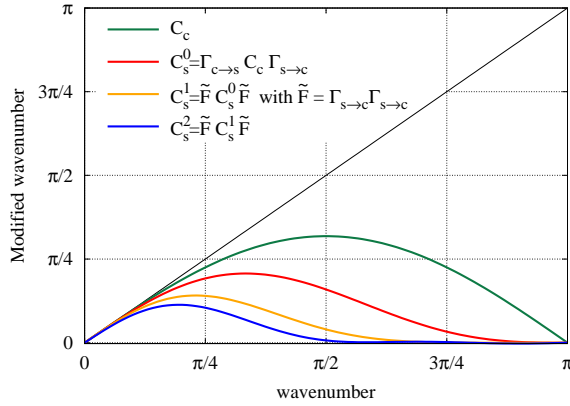


Figure 4: Modified wave-numbers for different approximations of the convective term.

to preserve the (skew-)symmetry of the discrete collocated operators.

Results obtained for a turbulent channel flow at $Re_\tau = 180$ using the approach proposed in Eq.(28) are displayed in Figure 3 (left). Apart from some details about the construction of the interpolation operators, $\Gamma_{c \rightarrow s}$ and $\Gamma_{s \rightarrow c}$, this approach is essentially the same proposed by Perot [23] and later adapted by Hicken *et al.* [24] using shift transformations. Although it is unconditionally stable and results for coarse meshes are rather good (even better than standard staggered formulations), it is clear that it does not work well since it does not converge to the DNS solution. The root of this problem is related to the dispersion errors introduced by the interpolations. Alternatively we propose the following approximation,

$$\partial_t^s = \tilde{F}^n \Gamma_{s \rightarrow c} \partial_t^c \Gamma_{c \rightarrow s} \tilde{F}^n, \quad (30)$$

where $\tilde{F} = \Gamma_{s \rightarrow c} \Gamma_{c \rightarrow s}$ can be view as a symmetric spatial filter that improves the spectral properties of the convective operators (see Figure 4). Notice that for $n = 0$, Eq.(30) simplifies to Eq.(28). Results obtained with $n = 2$ are displayed in Figure 3 (right) showing a great improvement respect to the other approach. Results with $n = 1$ (not shown here) and $n = 2$ are very similar. This clear improvement may

be explained as follows: back and forth interpolations in Eq.(28) worsen the numerical representation for the highest wave-numbers. This additional filtering can be viewed as a regularization (smoothing) that effectively damps the convective transport of scales smaller than twice the grid size (see Figure 4). To study this in more detail, including tests for unstructured grids, is part of our near future research plans.

ACKNOWLEDGMENTS

F.X.T., N.V. and A.O. have been financially supported by the *Ministerio de Economía y Competitividad*, Spain, ANUMESOL project (ENE2017-88697-R). F.X.T. and A.O. are supported by the Generalitat de Catalunya RIS3CAT-FEDER, FusionCAT project (001-P-001722). N.V. was supported by an FI AGAUR-Generalitat de Catalunya fellowship (2017FI B 00616). Calculations were performed on the IBM MareNostrum 4 supercomputer at the BSC. The authors thankfully acknowledge these institutions.

REFERENCES

- [1] R. W. C. P. Verstappen and A. E. P. Veldman, “Symmetry-Preserving Discretization of Turbulent Flow,” *Journal of Computational Physics*, vol. 187, pp. 343–368, 2003.
- [2] F. X. Trias, O. Lehmkuhl, A. Oliva, C.D. Pérez-Segarra, and R.W.C.P. Verstappen, “Symmetry-preserving discretization of Navier-Stokes equations on collocated unstructured meshes,” *Journal of Computational Physics*, vol. 258, pp. 246–267, 2014.
- [3] F. Dabbagh, F. X. Trias, A. Gorobets, and A. Oliva, “Flow topology dynamics in a three-dimensional phase space for turbulent Rayleigh-Bénard convection,” *Physical Review Fluids*, vol. 5, p. 024603, 2020.
- [4] F. X. Trias, A. Gorobets, and A. Oliva, “Turbulent flow around a square cylinder at Reynolds number 22000: a DNS study,” *Computers & Fluids*, vol. 123, pp. 87–98, 2015.
- [5] E. M. J. Komen, J. A. Hopman, E. M. A. Frederix, F. X. Trias, and R. W. C. P. Verstappen, “Energy-preserving discretisation for LES/DNS with unstructured collocated grids in OpenFOAM,” in *14th World Congress in Computational Mechanics and ECCOMAS Congress (WCCM-ECCOMAS 2020)*, (Paris, France (online)), July 2020 (postponed to January 2021 due to COVID19 pandemic).
- [6] S. V. Patankar and D. B. Spalding, “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows,” *International Journal of Heat and Mass Transfer*, vol. 15, pp. 1787–1806, 1972.
- [7] J. Doormal and G.D.Raithby, “Enhancements of the SIMPLE method for Predicting Incompressible Fluid Flows,” *Numerical Heat Transfer*, vol. 7, pp. 147–163, 1984.
- [8] S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Corporation, McGraw-Hill Book Company, 1980.
- [9] R. I. Issa, “Solution of the Implicitly Discretized Fluid Flow Equations by Operator-Splitting,” *Journal of Computational Physics*, vol. 62, pp. 40–65, 1985.
- [10] F. Moukalled and M. Darwish, “A unified formulation of the segregated class of algorithms for fluid flow at all speeds,” *Numerical Heat Transfer, Part B*, vol. 37, pp. 103–139, 2000.

- [11] A. J. Chorin, “Numerical Solution of the Navier-Stokes Equations,” *Mathematics of Computation*, vol. 22, pp. 745–762, 1968.
- [12] R. Temam, “Sur l’approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II),” *Archive for Rational Mechanics and Analysis*, vol. 33, pp. 377–385, 1969.
- [13] J. B. Perot, “An analysis of the fractional step method,” *Journal of Computational Physics*, vol. 108, pp. 51–58, 1993.
- [14] C. M. Klaij, “On the stabilization of finite volume methods with co-located variables for incompressible flow,” *Journal of Computational Physics*, vol. 297, pp. 84–89, 2015.
- [15] N. Valle, F. X. Trias, and J. Castro, “An energy-preserving level set method for multiphase flows,” *Journal of Computational Physics*, vol. 400, no. 1, p. 108991, 2020.
- [16] Àdel Alsalti-Baldellou, F. X. Trias, A. Gorobets, and A. Oliva, “On preconditioning variable Poisson equation with extreme contrasts in the coefficients,” in *14th World Congress in Computational Mechanics and ECCOMAS Congress (WCCM-ECCOMAS 2020)*, (Paris, France (online)), July 2020 (postponed to January 2021 due to COVID19 pandemic).
- [17] M. S. Dodd and A. Ferrante, “A fast pressure-correction method for incompressible two-fluid flows,” *Journal of Computational Physics*, vol. 273, pp. 416–434, 2014.
- [18] C. M. Rhie and W. L. Chow, “Numerical study of the turbulent flow past an airfoil with trailing edge separation,” *AIAA Journal*, vol. 21, pp. 1525–1532, 1983.
- [19] E. M. J. Komen, L. H. Camilo, A. Shams, B. J. Geurts, and B. Koren, “A quantification method for numerical dissipation in quasi-DNS and under-resolved DNS, and effects of numerical dissipation in quasi-DNS and under-resolved DNS of turbulent channel flows,” *Journal of Computational Physics*, vol. 345, pp. 565–595, 2017.
- [20] Shashank, J. Larsson, and G. Iaccarino, “A co-located incompressible Navier-Stokes solver with exact mass, momentum and kinetic energy conservation in the inviscid limit,” *Journal of Computational Physics*, vol. 229, pp. 4425–4430, 2010.
- [21] J. van Kan, “A second-order accurate pressure-correction scheme for viscous incompressible flows,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, pp. 870–891, 1986.
- [22] D. Santos, J. Muela, N. Valle, and F. X. Trias, “On the interpolation problem for the Poisson equation on collocated meshes,” in *14th World Congress in Computational Mechanics and ECCOMAS Congress (WCCM-ECCOMAS 2020)*, (Paris, France (online)), July 2020 (postponed to January 2021 due to COVID19 pandemic).
- [23] B. Perot, “Conservative properties of unstructured staggered mesh schemes,” *Journal of Computational Physics*, vol. 159, pp. 58–89, 2000.
- [24] J. E. Hicken, F. E. Ham, J. Militzer, and M. Koksal, “A shift transformation for fully conservative methods: turbulence simulation on complex, unstructured grids,” *Journal of Computational Physics*, vol. 208, pp. 704–734, 2005.
- [25] J. Kim, P. Moin, and R. Moser, “Turbulence statistics in fully developed channel flow at low Reynolds number,” *Journal of Fluid Mechanics*, vol. 177, pp. 133–166, 1987.