The 9th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2024
3–7 June 2024, Lisboa, Portugal

COST-ACCURACY ANALYSIS FOR SYMMETRY-PRESERVING METHODS

J.A. Hopman[1*], À. Alsalti-Baldellou[2], J. Rigola[1] and F.X. Trias[1]

[1] Heat and Mass Transfer Technological Center, Technical University of Catalonia, ESEIAAT, c/Colom 11, 08222 Terrassa, Spain,
jannes.hopman@upc.edu, github.com/janneshopman

[2] Department of Information Engineering, University of Padova, Via Giovanni Gradenigo, 6b, 35131 Padova PD, Italy

Key words: FVM, Symmetry-preserving discretisation, Cost-versus-accuracy, OpenFOAM

Summary. Reducing computational cost is of essence to make computational fluid dynamics as a viable tool for industrial design purposes. This work focuses on increasing solution accuracy by applying the symmetry-preserving method, which allows for the same accuracy with lower computational cost. The symmetry-preserving method is derived and an implementation of an algorithm is shown. To establish a cost-versus-accuracy relation, this solver is then tested and compared with two non-symmetry-preserving codes, using opensource software OpenFOAM. A turbulent channel flow case was run with varying number of control volumes in the wall-normal direction to increase computational cost. Accuracy was measured using several lower- and higher-order statistics. Although differencesin accuracy were visible between the methods, alongside its unconditional stability, the computational set-up did not allow establishment of a clear cost-versus-accuracy relation. It is suggested to also increase spatial resolution in streamwise and spanwise directions to obtain better results, as well as using different metrics for accuracy or another test cases set-up. In conclusion, the symmetry-preserving method shows improved accuracy and robustness, but additional work has to be carried out to explore a clear cost-versus-accuracy relation.

1 INTRODUCTION

In recent years, Computational Fluid Dynamics (CFD) is used increasingly as a design tool for industrial applications, such as the medical, automotive and renewable energy industries. The main constraint of the use of CFD in industrial applications nowadays lies in the computational cost and the wall-clock simulation time. This constraint inevitably leads to a cost-versus-accuracy trade-off when simulations are performed. This work is part of a larger project [1] in which the viability of conducting overnight LES simulations on GPU-accelerated supercomputers is evaluated, aiming to combine a highly-portable algebraic framework with a symmetry-preserving discretisation for unstructured collocated grids. The former part aims to cut down on the cost side, which is not the focus of this work. Instead, this work focuses on the potential gain in accuracy obtained by properly applying a symmetry-preserving discretisation.

The symmetry-preserving discretisation aims to conserve energy, momentum and mass of the simulation by mimicking properties of the continuous operators of the Navier-Stokes equations

in their discrete counterparts. This property is deemed essential in accurately depicting the motion of fluids at any scale, which has to be carried out properly in turbulent flow simulations [2]. This method was later extended to collocated grids by [3]. The effect on the accuracy of applying this scheme will be compared to the use of non-symmetry-preserving schemes using the collocated, finite volume, open-source code OpenFOAM, for which the method was previously implemented by [4]. A fully developed turbulent channel flow case at Re_τ is used as a test case, while monitoring lower- and higher- order turbulent statistics, to perform a cost-versus-accuracy analysis for the proposed methods.

2 NUMERICAL METHODS

In this section, the essence of the symmetry-preserving method for collocated grids is explained, as well as the algorithm that follows from it to solve the test cases. Subsequently, the main differences with the readily-available solver algorithms of OpenFOAM are discussed, and two non-symmetry-preserving algorithms are introduced, which are also tested in section 3.

2.1 Symmetry-preserving method

Let the semi-discrete version of the incompressible Navier-Stokes equations be written as:

$$\Omega \partial_t \mathbf{u}_c + C(\mathbf{u}_s)\mathbf{u}_c = -D\mathbf{u}_c - \Omega G_c \mathbf{p}_c, \tag{1}$$

$$M\mathbf{u}_s = \mathbf{0}_c. \tag{2}$$

The collocated velocity field is given by $\mathbf{u}_c = \left(\mathbf{u}_{c,x}^T, \mathbf{u}_{c,y}^T, \mathbf{u}_{c,z}^T\right)^T \in \mathbb{R}^{3n}$, the staggered velocity field is given by $\mathbf{u}_s \in \mathbb{R}^m$, and finally, the kinematic pressure field is given by $\mathbf{p}_c \in \mathbb{R}^n$. n and m give the number of control volumes and faces, respectively. The discrete operators of equations (1) and (2) can be derived from the geometry of the mesh, after discretising the domain, as seen in figure 1. The two-dimensional figure serves as an illustration that can easily be extended to three dimensions, for which the matrices are constructed. The face-owner and face-neighbour connectivity matrices, $T_{fo}, T_{fn} \in \mathbb{R}^{n\times m}$, contain entry $(i, f) = 1$ if cell i connects to face f as an owner or neighbour respectively. The face-normal matrix, $N_s = (N_{sx}, N_{sy}, N_{sz}) \in \mathbb{R}^{m\times 3m}$, contains the (x, y, z)-components of the face-normals, $\mathbf{n}_f$. The face area matrix, $A_s \in \mathbb{R}^{m\times m}$, contains the areas of the faces, A_f, on its diagonal. The face-normal distance matrix, $\delta_{ns} \in \mathbb{R}^{m\times m}$, contains δ_{nf}, the distance between cell-centers i and j projected onto $\mathbf{n}_f$, on its diagonal. Finally, the cell-volume matrix, $\Omega_c \in \mathbb{R}^{n\times n}$, contains the volumes of each cell, Ω_i, on its diagonal. The operators for collocated volumes, $\Omega \in \mathbb{R}^{3n\times 3n}$, staggered volumes, $\Omega_s \in \mathbb{R}^{m\times m}$, divergence, $M \in \mathbb{R}^{n\times m}$, gradient, $G \in \mathbb{R}^{m\times n}$, Laplacian, $L \in \mathbb{R}^{n\times n}$, cell-to-face dot-interpolator, $\Gamma_{cs} \in \mathbb{R}^{m\times 3n}$, collocated divergence, $M_c \in \mathbb{R}^{n\times 3n}$, collocated gradient, $G_c \in \mathbb{R}^{3n\times n}$, convection $C(\mathbf{u}_s) \in \mathbb{R}^{3n\times 3n}$, and diffusion $D \in \mathbb{R}^{3n\times 3n}$, are then derived as:

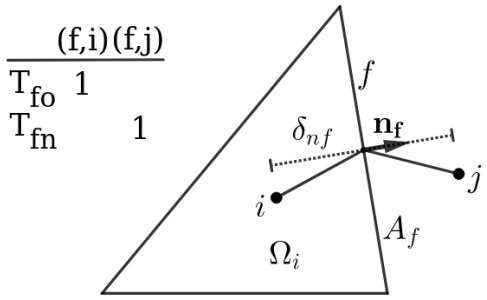


Figure 1: Geometric parameters needed to constitute the discrete symmetry-preserving operators.

$$\Omega = I_3 \otimes \Omega_c, \tag{3}$$
$$\Omega_s = A_s \delta_{ns}, \tag{4}$$
$$M = (T_{fo} - T_{fn})\, A_s, \tag{5}$$
$$G = -\Omega_s M^T, \tag{6}$$
$$L = MG, \tag{7}$$
$$\Gamma_{cs} = N_s \left(I_3 \otimes \Pi_{cs}^V\right), \tag{8}$$
$$M_c = M\Gamma_{cs}, \tag{9}$$
$$G_c = -\Omega M_c^T, \tag{10}$$
$$\Gamma_{sc} = \Omega^{-1}\Gamma_{cs}^T \Omega_s \tag{11}$$
$$C(\mathbf{u}_s) = I_3 \otimes \left(M diag\left(\mathbf{u}_s\right) \Pi_{cs}^M\right), \tag{12}$$
$$D = I_3 \otimes \left(-\nu L\right), \tag{13}$$

in which ν gives the kinematic viscosity and $\Pi_{cs} \in \mathbb{R}^{n\times m}$ gives the face-to-cell interpolator. Superscript V denotes volumetric interpolation, whereas superscript M denotes midpoint interpolation. The importance of using midpoint interpolation in the convective term is given in [2] and [3], whilst the importance of the volumetric interpolator that appears in the collocated divergence and gradient is discussed in [5].

The symmetry-preserving method conserves kinetic energy, $E_k = \frac{1}{2}\mathbf{u}_c^T \Omega \mathbf{u}_c$. The temporal evolution of this term is given by:

$$\frac{d}{dt}E_K = -\frac{1}{2}\begin{pmatrix} \mathbf{u}_c^T \left(C(\mathbf{u}_s) + C^T(\mathbf{u}_s)\right)\mathbf{u}_c \\ +\mathbf{u}_c^T \left(D + D^T\right)\mathbf{u}_c \\ +\mathbf{u}_c^T \Omega G_c \mathbf{p}_c + \mathbf{p}_c^T G_c^T \Omega^T \mathbf{u}_c \end{pmatrix}. \tag{14}$$

The first right-hand side (RHS) term is zero as the convective term is skew-symmetric by construction, i.e. $C(\mathbf{u}_s) = -C^T(\mathbf{u}_s)$. The second RHS term is strictly dissipative as D is (symmetric) negative semi-definite, i.e. $\mathbf{x}^T D \mathbf{x} \leq 0$ for all $\mathbf{x} \in \mathbb{R}^{3n}$. The third RHS term is strictly dissipative and close to zero as $G_c^T \Omega^T \mathbf{u}_c = M_c \mathbf{u}_c \approx \mathbf{0}_c$, which is related to the mass conservation of the collocated velocity field [5].

The symmetry-preserving operators are implemented in a time-stepping algorithm using the projection method:

$$\mathbf{u}_c^p = \mathcal{F}\left(\mathbf{u}_c, \mathbf{u}_s\right), \tag{15}$$
$$\mathbf{u}_c^{p*} = \mathbf{u}_c^p - G_c\tilde{\mathbf{p}}_c^p, \tag{16}$$
$$L\tilde{\mathbf{p}}_c' = M_c\mathbf{u}_c^{p*}, \tag{17}$$
$$\tilde{\mathbf{p}}_c^{n+1} = \tilde{\mathbf{p}}_c^p + \tilde{\mathbf{p}}_c', \tag{18}$$
$$\mathbf{u}_s^{n+1} = \Gamma_{cs}\mathbf{u}_c^p - G\tilde{\mathbf{p}}_c^{n+1}, \tag{19}$$
$$\mathbf{u}_c^{n+1} = \mathbf{u}_c^p - G_c\tilde{\mathbf{p}}_c^{n+1}, \tag{20}$$

where $\tilde{\mathbf{p}} = \Delta t\mathbf{p}$ denotes the pseudo-pressure. $\mathcal{F}\left(\mathbf{u}_c, \mathbf{u}_s\right)$ denotes the temporal integration method. In this work, the projection method was implemented within a Runge-Kutta framework which uses the PISO method, which was implemented in the OpenFOAM solver *RKSymFoam* [4][6].

2.2 Non-symmetry-preserving methods

The readily available OpenFOAM solver *pimpleFoam* was extended to include the Runge-Kutta time-stepping framework to serve as a non-symmetry-preserving solver. There are two main differences between *pimpleFoam* and the symmetry-preserving solver.

Firstly, as a default, OpenFOAM does not use symmetry-preserving spatial discretisations. It uses linear interpolations from cells to faces, for example in the collocated divergence, $M_c = M\Gamma_{cs}^L$. Furthermore, the collocated gradient is performed by a linear interpolation followed by a Gauss-gradient operation, $G_G\Pi_{cs}^L$. This operator is described and rewritten to $\Gamma_{sc}^V G$ in [7]. When combining equations (9), (10) and (11), it can be observed that these operators are inconsistent, as $\Gamma_{sc}^V \neq \Omega^{-1}\Gamma_{cs}^{L^T}\Omega_s$. Moreover, linear interpolation is used in the convective term, allowing it to become non-skew-symmetric. Finally, a correction term is added to the gradient, G, such that equation (6) no longer holds, i.e. $G \neq -\Omega_s M^T$. This correction term of the gradient of field ϕ_c is given by:

$$\nabla_{corr} = \left(\mathbf{n}_f - \mathbf{d}_f/\cos\theta_f\right) \cdot \left[\Gamma_{cs}^L G_c\phi_c\right]_f, \tag{21}$$

where $\mathbf{d}_f$ gives the unit vector of the vector connecting cells i and j, and θ_f denotes the angle between $\mathbf{n}_f$ and $\mathbf{d}_f$.

Secondly, a flux-correction term, ϕ_{corr}, is added to the term $\Gamma_{cs}\mathbf{u}_c^{p*}$, which appears implicitly on the RHS of equation (17). This flux-correction term depends on the chosen temporal integration method, as an example, for Forward Euler time-stepping, it is given by:

$$\phi_{coor} = -C_{u_s}\mathbf{u}_{s,corr}, \tag{22}$$
$$\mathbf{u}_{s,corr} = \mathbf{u}_s^n - \Gamma_{cs}^L\mathbf{u}_c^n, \tag{23}$$
$$\left[C_{u_s}\right]_{f,f} = 1 - min\left(\frac{||\left[\mathbf{u}_{s,corr}\right]_f||}{||\left[\mathbf{u}_s^n\right]||_f - \varepsilon}\right), \tag{24}$$

in which ε is a very small number to avoid division by zero. It was shown by [8] that this term stabilises the results at the cost of introducing a sizable amount of numerical dissipation.

Three different solver algorithms where tested in total for this work, of which an overview is given in table 1. Firstly, *pimpleFoam*, which contains the two major issues described in section 2.2. Secondly, ϕ-Foam, which is the same as *pimpleFoam*, except it has been stripped of the flux-correction term given in equation (22). And finally, the symmetry-preserving algorithm, as described in section 2.1, denoted as Sym-Pres.

	pimpleFoam	ϕ-Foam	Sym-Pres
Conservative operators	×	×	✓
Conservative fluxes	×	✓	✓

Table 1: Comparison of different tested solvers

3 RESULTS

3.1 Case set-up

To test the cost-versus-accuracy between the different solvers, a turbulent channel flow case at $Re_\tau = 180$ was used. The dimensions of the channel domain were $L_x = 4\pi h$, $L_y = 2h$ and $L_z = \frac{4}{3}\pi h$, in streamwise (x), wall-normal (y) and spanwise (z) direction respectively, with channel half-height h, as used in [9], see figure 2. Periodic boundaries were used in x- and z-directions and no-slip and zero-gradient boundary conditions at the walls for the velocity and pressure, respectively. A pressure gradient of -1 was imposed to drive the flow, leading to $u_\tau = 1$. The kinematic viscosity was set so that $Re_\tau = \frac{u_\tau h}{\nu} = 180$.

To examine a relation between the cost and the accuracy, the number of cells in the y-direction was varied as $Ny \in \{40, 60, 80, 100, 120, 140\}$, while N_x and N_z where kept constant at 40, which leads to different computational costs and accuracies of the solutions. The cell spacing in the y-direction was done with a hyperbolic tangent function, as was done in [10]. A diagonally-implicit Runge-Kutta 2 (DIRK2) temporal integration scheme was used with time-step size $\Delta t = 0.001$ to meet the Courant-Friedrichs-Lewy (CFL) condition for all meshes.

The statistical quantities that were measured were the mean stream-wise velocity, $\langle u_x \rangle$, the Reynolds shear stress $\langle u'_x u'_y \rangle$, the Root-Mean-Square (RMS) velocities, $RMS(u_i)$, the turbulent kinetic energy (TKE), $k = u'_i u'_i$. Furthermore, the higher-order statistical quantities of the kinetic energy budget terms were calculated, consisting of production, C_k^p, transport, C_k^T, viscous diffusion, D_k^v, dissipation, D_k^ϵ and pressure diffusion, P_k [4]. All statistical quantities are averaged in the periodic spatial directions. Spatial averaging was started after developing turbulence from an initial state, and was performed over 30 time units, which showed statistical convergence.

To express the results of each mesh-solver combination as a single data point, the accuracy was expressed as an error value, compared to reference solution of [9]. For this comparison, three statistical quantities were used, $\langle u_x \rangle$, $\langle u'_x u'_y \rangle$ and k, and the sum of the budget terms, Σ_{buds}. Two different methods of expressing the error were used. Firstly, the weighted RMS of the difference between the obtained values and the reference values, using the cell size as weights for the averaging. The second method was by simply looking at the difference in the peak values (maximum absolute values) of each statistical quantity. The error in Σ_{buds} was only expressed with the weighted RMS, as analytically it should be zero everywhere. With $\langle u_x \rangle$ as an example,

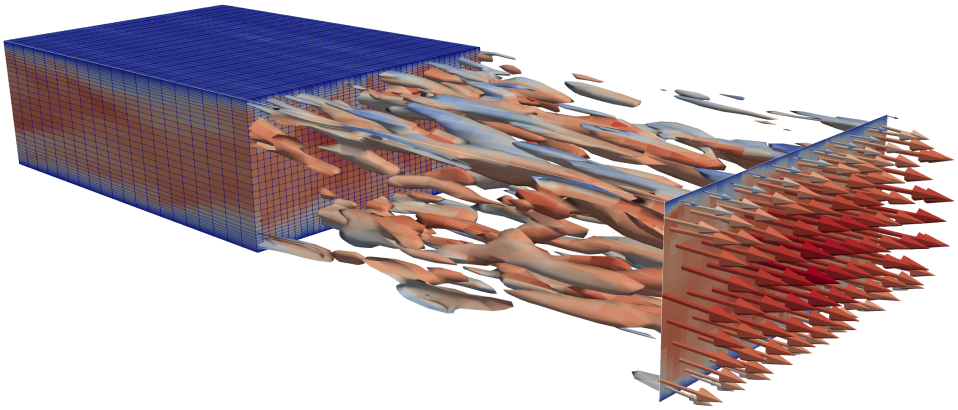


Figure 2: Snapshot of the turbulent channel flow, showing spatial discretisation (left), contours of vorticity at an arbitrary level to show turbulence (middle) and instantaneous velocities (right).

these values are then calculated as:

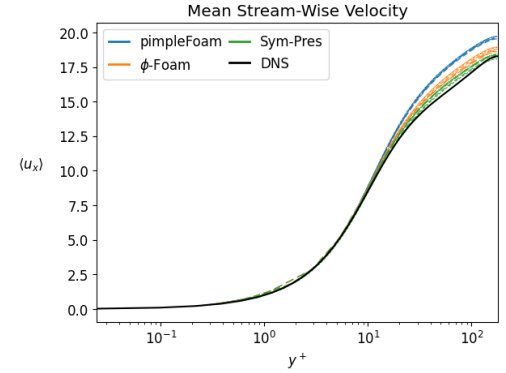
$$max\Delta\left(\langle u_x\rangle\right) = \left|max\langle|\langle u_x\rangle|\rangle - max\langle|\langle u_x\rangle_{ref}|\rangle\right|, \tag{25}$$

$$wRMS\Delta\left(\langle u_x\rangle\right) = \left(\frac{1}{h}\sum_{j=1}^{N_y}\left[dy\right]_j\left(\langle u_x\rangle_j - \langle u_x\rangle_{j,ref}\right)^2\right)^{\frac{1}{2}}. \tag{26}$$

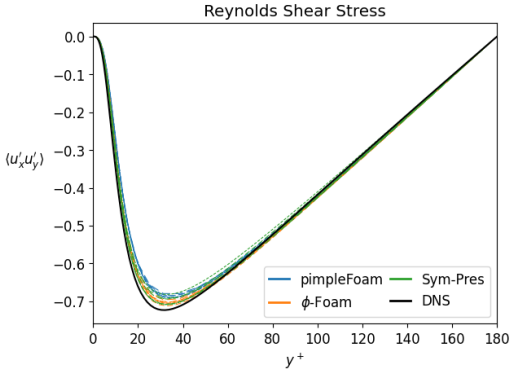
3.2 Cost-versus-accuracy results

Firstly, the results of the lower-order statistics are given in figure 3. The results for different values of N_y are lumped together in a single colour for each solver, so that a comparison of the range of the accuracy can be seen quickly from the image. In general, the results converged towards the reference result, therefore the partially overlapping lines form a range of spatial convergence for each solver. In figure 3a, it can be seen that in the wall region, the solutions mostly overlap. In the bulk region, however, *pimpleFoam* lies furthest away from the reference solution, with a large improvement when using conservative fluxes, as the ϕ-Foam results demonstrate. The Sym-Pres results lie closest to the reference solution and are best at reproducing the mean velocity profile that is expected. Similar results are observed for the RMS velocities, figure 3c and the TKE, figure 3d. The difference is less distinct in the Reynolds shear stress, figure 3b, where a distinction between *pimpleFoam* and ϕ-Foam is somewhat clear, but the effect of the spatial discritisation in Sym-Pres seems to have neither a positive nor a negative effect on accuracy for this set-up.

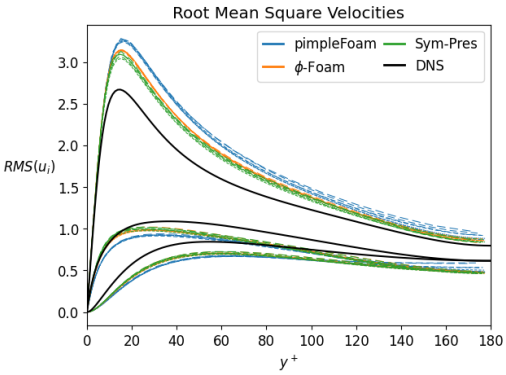
In order to see the relation between cost and accuracy, these lower-order statistical results were distilled into single data points using equations (25) and (26). The different RMS velocities were not considered, as k is dependent on these terms and could give a better single-point value of the accuracy of these sub-terms. The resulting relations are given in figure 4. The weighted errors given by $wRMS\left(\langle u_x\rangle\right)$ are in general highest for *pimpleFoam*, lowest for Sym-Pres, with ϕ-Foam in the middle, with similar results for $max\Delta\left(\langle u_x\rangle\right)$. However, mesh refinement does



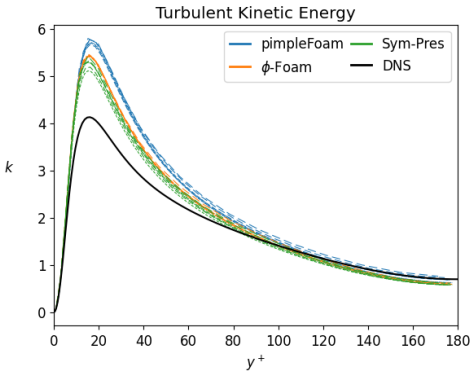
(a) Mean stream-wise velocity



(b) Reynolds shear stress



(c) RMS velocities



(d) Turbulent kinetic energy

Figure 3: Turbulent channel flow lower-order statistics compared for different solvers and values of N_y. Results for the different meshes are lumped together in one colour to provide an insight into relative solver performance at a quick glance, compared to each other and the reference solution of [9].

not show a big improvement for this variable for any of the solvers, and it is hard to make a direct cost-versus-accuracy analysis based on this data. This might be due to the fact that the mesh remained very coarse in x- and z-directions for all meshes, suggesting that these spatial directions should be refined first, before spending more resources on refinement in y. The results for the TKE again show the same general difference in accuracy of the solvers, with a slightly more visible trend of increased accuracy at higher cost. Although the results do not give a strong relation between cost and accuracy, and the conclusion that Sym-Pres with $N_y = 40$ is as accurate as ϕ-Foam with $N_y = 140$ seems far-fetched. Finally, the results for the Reynolds shear stress also show the improvement in the solver by removing the flux-correction, however, the results of ϕ-Foam and Sym-Pres largely overlap and no clear trends are visible. The main conclusions from the results so far are that removing the flux-correction term improves the accuracy of the results, while the spatial discretisation improves it only a little. While cost-versus-accuracy trends and comparisons are not possible using this method. Improvements here could be two-fold. Firstly, the mesh could be refined in all directions, to allow convergence to the reference solution, and secondly, different methods of measuring accuracy could be tested.

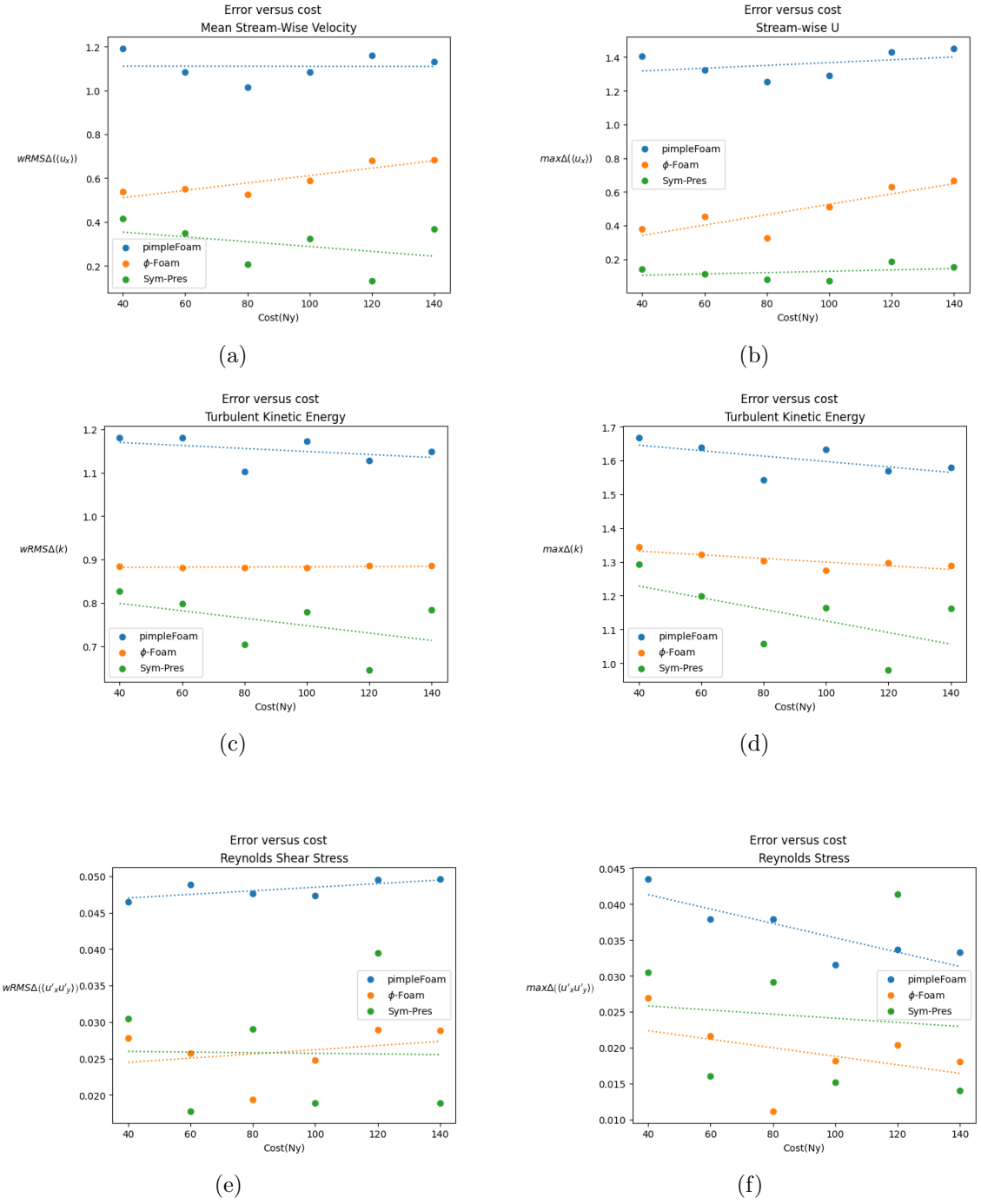


Figure 4: Error-versus-cost plots for the relevant lower-order statistics of the turbulent channel flow. Left/Right: weighted RMS of the deviation from the reference solution (Left), equation (26), peak value difference (Right), equation 25. Top/Center/Bottom: mean stream-wise velocity (Top), turbulent kinetic energy (Middle), Reynolds shear stress (Bottom). A least squares trend line is added for the set of solutions of each solver.

The results of the higher-order statistics are depicted in figure 4. In general, the budget terms overlapped mostly for all solvers. They are not analysed in detail, but an overview of all terms is given in figure 5a. The sum of the budget terms, figure 5b, show that the problems mainly reside at the wall, with some eratic results for *pimpleFoam*. The trend of the improvement in accuracy for this term, figure 5c does show a nice trend of reduction in error with increasing computational cost. However, the individual data points show some zig-zag and the trend lines overlap, making it difficult to take conclusions in terms of comparing the different solvers from these figures.

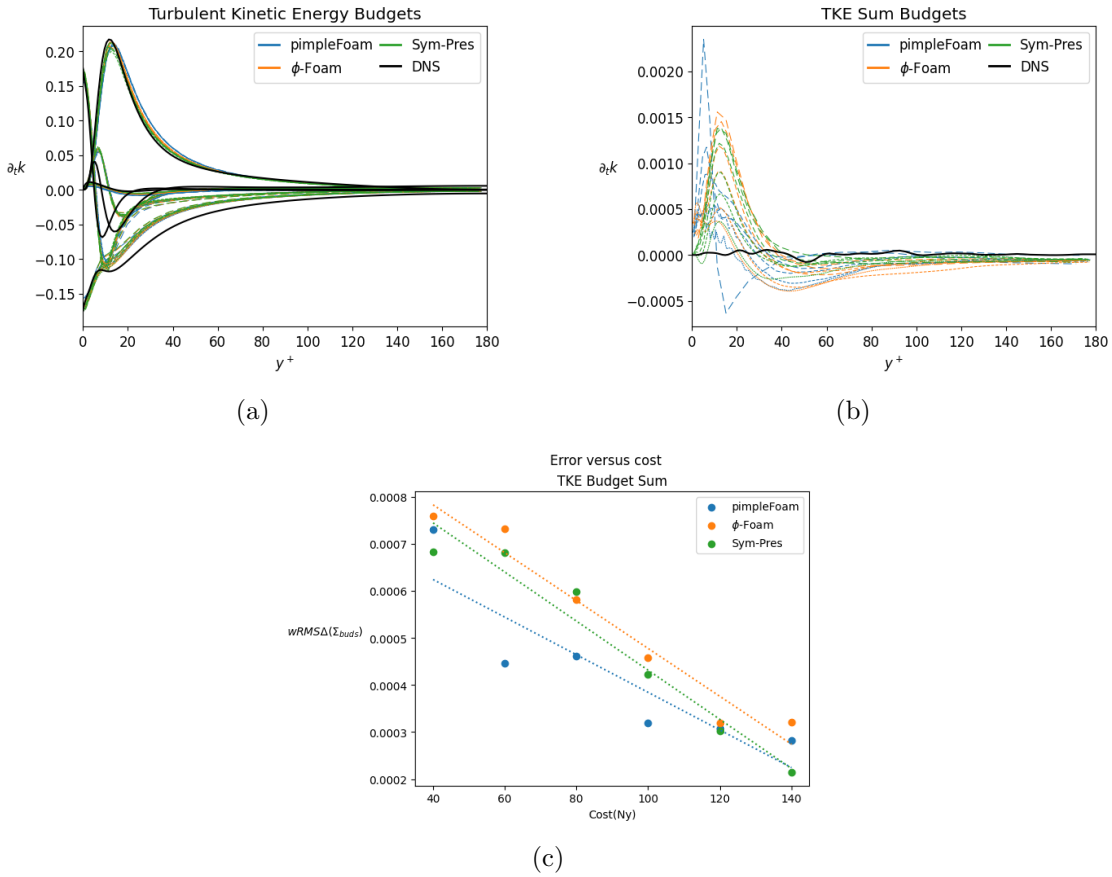


Figure 5: Sum budgets. (a) Combined budget, (b) Sum budgets, (c) Integrated sum budgets.

3.3 Robustness and stability

So far only Cartesian meshes have been considered, where the gradient correction of *pimpleFoam* and ϕ-Foam are zero and therefore do not influence the results. Moreover, one of the strengths of the symmetry-preserving method, is that it provides unconditional stability, even on highly distorted meshes. For these reasons, additional tests were performed for the $N_y = 100$ mesh with highly distorted control volumes. These control volumes were formed by stretching four out of eight diagonally placed vertices away from the cell-center, i.e. the (000), (011), (101) and (110) vertices, see figure 6 for an example of such a control volume. The channel flow on

this type of mesh run with *pimpleFoam* and ϕ-Foam quickly diverged due to accumulation of kinetic energy, whereas the Sym-Pres solver converged to give a result. The accuracy of these results is of course very low, and do not need to be examined in detail, see fore example the Reynolds shear stress in figure 7. Although on this mesh, any accuracy is hard to guarantee, the symmetry-preserving method is at least able to provide some result. This stability and robustness of the solver is of very great value in industrial applications, where challenging geometry and meshes might be used, and end-users might not be interested in figuring out why a simulation did not converge.

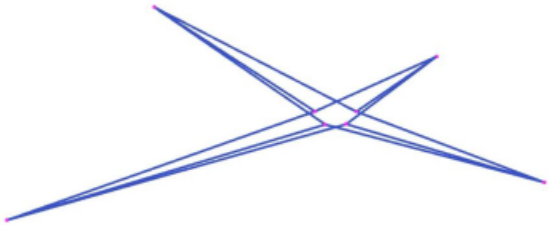


Figure 6: Control Volume

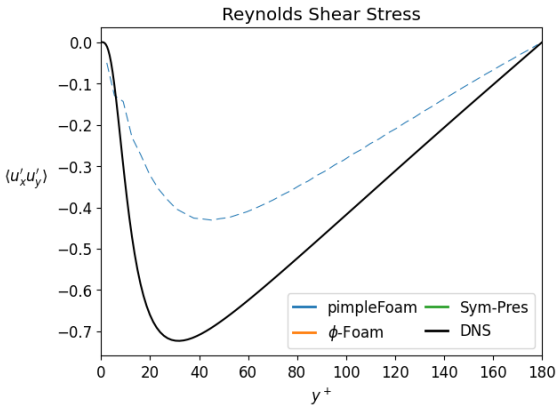


Figure 7: Reynolds Shear Stress

4 CONCLUSIONS AND DISCUSSION

In this work, it is shown that the conservative properties of the symmetry-preserving method can increase the accuracy simulation results. The derivation and implementation of the method are shown in this work and then compared to a readily-available solver in the opensource software OpenFOAM. The main differences that were identified were the non-symmetry-preserving spatial discretisation and the flux correction term, of which the latter had the biggest negative impact on the resulting accuracy. In theory, an increase in accuracy could equal a decrease in computational cost, if the targeted accuracy is kept constant. The relation between computational cost and accuracy in this work was shown using a turbulent channel flow case, using lower- and higher-order statistics, distilled to single data points of accuracy. The results of this approach, however, are not very convincing, and additional tests and an improved methodology are suggested. Firstly, the grid should be refined in all directions, or should already be of DNS quality in x- and z-directions, to allow convergence to the reference solution. Secondly, other measures of accuracy should be sought after, for instance by using a different test case. For example, the re-attachment point of the flow over a periodic hill could be a good measure. Finally, one of the main advantages of the symmetry-preserving method, its stability and robustness, was demonstrated by showing converged results even on highly distorted meshes. In conclusion, the symmetry-preserving method shows improved accuracy and robustness, but additional work has to be carried out to explore a clear cost-versus-accuracy relation.

ACKNOWLEDGEMENTS This work is supported by the SIMEX project (PID2022-142174OB-I00) of *Ministerio de Ciencia e Innovación*, Spain and the RETOtwin project (PDC2021-120970-I00) of *Ministerio de Economía y Competitividad*, Spain. J.A.H. is supported by the predoctoral grant FI 2024 (2024 FI_B1 00204) of the *Catalan Agency for Management of University and Research Grants (AGAUR)*.

REFERENCES

[1] Alsalti Baldellou, À., Colomer Rey, G., Hopman, J. A., Álvarez Farré, X., Gorobets, A., Trias Miquel, F. X., Pérez-Segarra, C.D. & Oliva Llena, A. (2023). Reliable overnight industrial LES: challenges and limitations. Application to CSP technologies. In *14th International ERCOFTAC Symposium on Engineering, Turbulence, Modelling and Measurements: 6th-8th September 2023, Barcelona, Spain: proceedings. European Research Community on Flow, Turbulence, and Conbustion (ERCOFTAC)*.

[2] Verstappen, R. W. C. P. & Veldman, A. E. P. (2003). Symmetry-preserving discretization of turbulent flow. *Journal of Computational Physics, 187(1), 343-368*.

[3] Trias, F. X., Lehmkuhl, O., Oliva, A., Pérez-Segarra, C. D. & Verstappen, R. W. C. P. (2014). Symmetry-preserving discretization of Navier–Stokes equations on collocated unstructured grids. *Journal of Computational Physics, 258, 246-267*.

[4] Komen, E. M. J., Hopman, J. A., Frederix, E. M. A., Trias, F. X. & Verstappen, R. W. C. P. (2021). A symmetry-preserving second-order time-accurate PISO-based method. *Computers & Fluids, 225, 104979*.

[5] Santos, D., Trias, F. X., Colomer, G. & Pérez-Segarra, C. D. (2022). An energy-preserving unconditionally stable fractional step method on collocated grids. In *Collection of papers presented at the 8th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS Congress 2022)*.

[6] Hopman, J. A. & Frederix, E. M. A. (2023). RKSymFoam GitHub page. *URL https://github.com/janneshopman/RKSymFoam*

[7] Hopman, J. A., Trias, F. X. & Rigola, J. (2023). On a conservative solution to checkerboarding: Examining the discrete Laplacian kernel using mesh connectivity. In *ERCOFTAC Workshop Direct and Large Eddy Simulation (pp. 306-311). Cham: Springer Nature Switzerland*.

[8] Vuorinen, V., Keskinen, J. P., Duwig, C. & Boersma, B. J. (2014). On the implementation of low-dissipative Runge–Kutta projection methods for time dependent flows using OpenFOAM. *Computers & Fluids, 93, 153-163*.

[9] Vreman, A. W. & Kuerten, J. G. (2014). Comparison of direct numerical simulation databases of turbulent channel flow at $Re_\tau = 180$. *Physics of Fluids, 26(1)*.

[10] Zhang, H., Trias, F. X., Gorobets, A., Tan, Y. & Oliva, A. (2015). Direct numerical simulation of a fully developed turbulent square duct flow up to $Re_\tau = 1200$. *International Journal of Heat and Fluid Flow, 54, 258-267*.