Turbulence, Heat and Mass Transfer 10

On a checkerboard-free, conservative method for turbulent flows

J.A. Hopman*,[1], À. Alsalti-Baldellou[1,2], F.X. Trias[1] and J. Rigola[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]*Termo Fluids SL, Sabadell (Barcelona), Spain; www.termofluids.com*

Abstract – In computational fluid dynamics (CFD), the checkerboard problem remains a challenging issue when employing collocated grid arrangements for complex geometries. It manifests as oscillatory and non-physical pressure modes, which can disrupt fluid motion and compromise numerical accuracy. This work focuses on quantifying and mitigating checkerboarding conservatively in numerical simulations of turbulent flows. First, several quantification methods are introduced and tested, leading to a global, normalised, dimensionless checkerboard coefficient C_{cb}, which can be employed even on complex geometries. Through this coefficient, a new self-regulatory solver was developped that monitors the quantity of checkerboarding during runtime and adjusts the pressure predictor accordingly to mitigate checkerboarding at the cost of a higher pressure error. Comparing this solver to existing solvers on a channel flow at $Re_\tau = 180$, and analysing the turbulent kinetic energy budgets, it was shown that this solver is effective in reaching its design purpose. The new approach provides a unified and efficient solution to mitigate checkerboarding, enhancing reliability and accuracy of CFD simulations.

1. Introduction

Industrial CFD codes are often used with complex geometries, for which a collocated grid arrangement is more suitable. Using a central differencing scheme to discretise the divergence and gradient operators of the Navier-Stokes equations will lead to a wide-stencil Laplacian, in which odd and even cells are decoupled. These decoupled cells can form oscillatory, non-physical patterns, which lie on the kernel of the discrete Laplacian and are therefore invisible to the CFD algorithm. This problem is known as the checkerboard problem [3].

Most collocated CFD methods mitigate these oscillations by applying a compact-stencil Laplacian, which couples neighbouring cells. However, this method introduces numerical dissipation [7, 2], which disrupts the motion of fluids, especially at the smallest turbulent length-scales [10]. A different, non-dissipative solution is to filter the spurious pressure modes which lie on the kernel of the discrete wide-stencil Laplacian [6, 4]. This is not suitable for complex geometries however, since calculating the kernel becomes too computationally expensive.

Regardless of the use of the compact-stencil Laplacian, oscillatory patterns can sometimes still be observed, especially when small timesteps are used in unsteady cases [2], or when a predictor pressure is included in the velocity predictor [5]. In this work, the occurrence of checkerboarding despite the usage of a compact-stencil Laplacian is examined. Since the compact-stencil Laplacian does not contain spurious modes on its kernel, a broader definition of checkerboarding also has to be examined. Multiple definitions to do so were designed and tested. The checkerboard problem was quantified for a turbulent channel flow at $Re_\tau = 180$ to examine the different definitions and to compare them to the definition that is restricted to the wide-stencil Laplacian kernel modes.

Finally, when quantification during runtime is made possible, a solver is proposed that uses this information to self-regulate the amount of numerical dissipation through the pressure error. This solver is tested and compared to existing solvers using a channel flow at $Re_\tau = 180$, comparing both qualitative and quantitative checkerboarding. Furthermore, an analysis of the turbulent kinetic energy budgets was made to monitor possible loss of accuracy in the solution, as a result of this self-regulation.

2. Quantification methods

To examine the origin and quantification methods of checkerboarding, the Navier-Stokes equations were discretised and a fractional step method was used to advance the equations in time. This method is given in algorithm 1, in which Forward Euler is used for simplicity as an example for the temporal integration. The notation of algorithm 1 follows the work of [8], where

$$L = MG, \tag{1}$$
$$L_c = M_c G_c = M\Gamma_{cs}\Gamma_{sc}G, \tag{2}$$
$$\Gamma_{sc} = \Omega^{-1}\Gamma_{cs}^T\Omega_s, \tag{3}$$
$$-M^T = \Omega_s G, \tag{4}$$
$$-M_c^T = -(M\Gamma_{cs})^T = \Omega\Gamma_{sc}G = \Omega G_c, \tag{5}$$

with Laplacian, L, gradient, G, divergence, M, collocated operators denoted by subscript c, cell-to-face interpolator, Γ_{cs}, face-to-cell interpolator, Γ_{sc}, cell volumes, Ω and face volumes, Ω_s. Note that the Laplacians are symmetric and that L_c and G_c have kernels that contain spurious modes. In contrast to not using a pressure predictor, i.e. setting $\tilde{\mathbf{p}}_c^p = \mathbf{0}_c$, using a predictor value of $\tilde{\mathbf{p}}_c^p = \tilde{\mathbf{p}}_c^n$ led to more severely pronounced checkerboarding in a turbulent channel flow [4]. This is because, effectively, a larger part of the pressure is treated with a wide-stencil Laplacian, as can be seen in equation A1.3. The difference in checkerboarding can be monitored in many ways, and usually the term checkerboarding is used very loosely and qualitatively. Therefore multiple methods are derived below and tested for their validity and correspondence to qualitative results.

Partial velocity predictor	$R_c^n = \mathbf{u}_c^n - \Delta t\Omega^{-1}\left(C\left(\mathbf{u}_s^n\right) + D\right)\mathbf{u}_c^n$	(A1.1)
Velocity predictor	$\mathbf{u}_c^p = R_c^n - G_c\tilde{\mathbf{p}}_c^p$	(A1.2)
Poisson equation	$L\tilde{\mathbf{p}}_c' = M_c\mathbf{u}_c^p = M_c R^n - L_c\tilde{\mathbf{p}}_c^p$	(A1.3)
Update cell-centered velocities	$\mathbf{u}_c^{n+1} = \mathbf{u}_c^p - G_c\tilde{\mathbf{p}}_c'$	(A1.4)
Update face-centered velocities	$\mathbf{u}_s^{n+1} = \Gamma_{cs}\mathbf{u}_c^p - G\tilde{\mathbf{p}}_c'$	(A1.5)
Update pressure	$\tilde{\mathbf{p}}_c^{n+1} = \tilde{\mathbf{p}}_c^p + \tilde{\mathbf{p}}_c'$	(A1.6)

Algorithm 1: Fractional step method including pressure predictor

Strict kernel definition – "$||\mathbf{p}_{CB}||$ method"
This definition only considers modes of pressure that lie on the kernel of the wide-stencil Laplacian, L_c. If L_c is constructed using volumetric interpolation, i.e. $\phi_f = (\Delta_- \phi_- + \Delta_+ \phi_+) / (\Delta_- + \Delta_+)$, then a set of pressure modes that span the kernel for 3D Cartesian meshes, $\mathbf{p}_c^-$, can be constructed as follows:

$$Ker(L_c) = span(\mathbf{p}^-) = span\Big(1, \quad (-1)^i \Delta_i, \quad (-1)^{i+j} \Delta_i \Delta_j, \quad (-1)^{i+j+k} \Delta_i \Delta_j \Delta_k\Big) \tag{6}$$

$$||\mathbf{p}_{CB}|| = ||proj_{Ker(L_c)}(\mathbf{p}_c^{n+1})|| \tag{7}$$

where i, j and k are the indices of the cells in each Cartesian direction. This results in 8 modes if permutations of Cartesian directions are accounted for. The quantification relies on taking the magnitude of the pressure field projected onto the kernel modes, given by equation 7. This definition only holds on Cartesian meshes.

Norm of the Rhie-Chow correction term – "$||RC||$ method"
The Rhie and Chow correction term is the difference between the face-centered velocites, $\mathbf{u}_s^{n+1}$, and the interpolated cell-centered velocities, $\Gamma_{sc}\mathbf{u}_c^{n+1}$, which causes the damping of the spurious pressure modes. It can be expressed from equations A1.5 and A1.4 as:

$$\begin{aligned} ||RC|| &= ||\mathbf{u}_s^{n+1} - \Gamma_{cs}\mathbf{u}_c^{n+1}|| \\ &= ||\Gamma_{cs}\mathbf{u}_c^{p} - G\tilde{\mathbf{p}}_c' - \Gamma_{cs}\left(\mathbf{u}_c^{p} - G_c\tilde{\mathbf{p}}_c'\right)|| \\ &= ||\left(\Gamma_{cs}G_c - G\right)\tilde{\mathbf{p}}_c'|| \end{aligned} \tag{8}$$

Pressure diffusion energy budget term – "$\mathbf{P_{diff}}$ method"
Another quantification method, which makes more physical sense, is the pressure diffusion budget term, found by taking the inner product of velocity and the momentum equation, given in equation 9. In a continuous framework this term equals zero, however, not necessarily when discretised. In turn, this budget term can be rewritten to include the term found in equation 8. Here, $\Omega G_c = -M_c^T$ and $M\mathbf{u}_s^{n+1} = \mathbf{0}_c$ are used to rewrite in terms of equation 8:

$$\begin{aligned} P_{diff} &= {\mathbf{u}_c^{n+1}}^T \Omega G_c \mathbf{p}_c^{n+1} \\ &= -{\mathbf{p}_c^{n+1}}^T M_c \mathbf{u}_c^{n+1} \\ &= {\mathbf{p}_c^{n+1}}^T M\left(\mathbf{u}_s^{n+1} - \Gamma_{cs}\mathbf{u}_c^{n+1}\right) \\ &= {\mathbf{p}_c^{n+1}}^T M\left(\Gamma_{cs}G_c - G\right)\tilde{\mathbf{p}}_c'. \end{aligned} \tag{9}$$

Difference in the norms of the gradients – "$\Delta||\nabla p||$ method"
Next, a method similar to equation 9 was considered, where $\tilde{\mathbf{p}}_c'$ is replaced by $\mathbf{p}_c^{n+1^T}$ to make it independent of time-step size and pressure predictor. It then follows that this term is a simple difference in the norms of the compact- and wide-stencil gradients of pressure:

$$\begin{aligned}\Delta||\nabla p|| &= \mathbf{p}_c^{n+1^T} M\left(\Gamma_{cs} G_c - G\right)\mathbf{p}_c^{n+1} \\ &= \mathbf{p}_c^{n+1^T}\left(L_c - L\right)\mathbf{p}_c^{n+1} \\ &= ||G\mathbf{p}_c^{n+1}|| - ||G_c\mathbf{p}_c^{n+1}||,\end{aligned} \tag{10}$$

where the norm of a field is expressed as $||\phi_c|| = \phi_c^T \Omega \phi_c$ and $||\phi_s|| = \phi_s^T \Omega_s \phi_s$. This term has the nice properties that highly oscillatory pressure modes, such as the ones in $Ker(L_c)$, are only filtered by G_c and will therefore be represented in this term.

Ratio of connectivity norms – "C_{cb} method"
Finally, a method was derived from equation 10, where the gradient is replaced by a non-dimensional gradient, G^*, which is a cell-to-face incidence matrix, so that:

$$[G^*\mathbf{p}_c]_f = [\delta_{n,f} G\mathbf{p}_c]_f = \mathbf{p}_{c,n} - \mathbf{p}_{c,o}, \tag{11}$$

$$M^* = -G^{*^T}\Omega_s, \tag{12}$$

$$L^* = M^* G^* = -G^{*^T}\Omega_s G^*, \tag{13}$$

$$L_c^* = M_c^* G_c^* = -G_c^* \Omega G_c^*. \tag{14}$$

Then, the norm of $||G^*\mathbf{p}_c||$ is divided out to normalise the term. Leading to the checkerboard coefficient, C_{cb}:

$$\begin{aligned}C_{cb} &= -\frac{\mathbf{p}_c^T\left(L_c^* - L^*\right)\mathbf{p}_c}{\mathbf{p}_c^T L^* \mathbf{p}_c} \\ &= \left(||G^*\mathbf{p}_c|| - ||G_c^*\mathbf{p}_c||\right)/||G^*\mathbf{p}_c||\end{aligned} \tag{15}$$

Choice in quantification method
A turbulent channel flow at $Re_\tau = 180$ was run on a $4\pi \times 2 \times \frac{4}{3}\pi$ domain with $38 \times 108 \times 38$ cells in x, y, z respectively. The mesh was stretched in the y-direction from $\Delta y_w^+ = 0.66$ to $\Delta y_B^+ = 9.4$. The symmetry-preserving Runge-Kutta OpenFOAM solver *RKSymFoam* was used, which was developped for [5] and can be found in the GitHub repository given in the author's affiliations. The Runge-Kutta 3 scheme was used with $\Delta t = 0.001$. The case was run with $\tilde{\mathbf{p}}_c^p = \mathbf{0}_c$ and $\tilde{\mathbf{p}}_c^p = \tilde{\mathbf{p}}_c^n$, to obtain different levels of checkerboarding. Also, the effect of filtering the $\mathbf{p}^-$-modes was examined, obtaining a total number of 4 different cases.

From figure 1 it can clearly be seen that the choice in pressure predictor has a sizable effect on the severity of the oscillations that can be observed. Though much smaller, some oscillations can also be observed in the channel flows with absence of a pressure predictor. Finally, the effect of filtering the $\mathbf{p}^-$ modes does not remove the visible oscillations in either case.

Table 1: Quantification of checkerboarding using different methods

	$\tilde{\mathbf{p}}_c^p = \mathbf{0}_c$		$\tilde{\mathbf{p}}_c^p = \tilde{\mathbf{p}}_c^n$	
	unfiltered	filtered	unfiltered	filtered
$\|\|\mathbf{p}_{CB}\|\|$	$3.4 \cdot 10^{-5}$	$2.8 \cdot 10^{-22}$	$1.1 \cdot 10^{-5}$	$1.1 \cdot 10^{-22}$
$\|\|RC\|\|$	$4.3 \cdot 10^{-3}$	$4.3 \cdot 10^{-3}$	$3.7 \cdot 10^{-5}$	$3.7 \cdot 10^{-5}$
P_{diff}	-5.5	-5.5	$-5.8 \cdot 10^{-3}$	$-5.8 \cdot 10^{-3}$
$\Delta\|\|\nabla p\|\|$	$6.0 \cdot 10^{3}$	$6.0 \cdot 10^{3}$	$1.6 \cdot 10^{4}$	$1.6 \cdot 10^{4}$
C_{cb}	0.62	0.62	0.76	0.76

The quantification results in table 1 firstly show that the $\mathbf{p}^-$-filtering only affects the value of $||\mathbf{p}_{CB}||$, however, the problem qualitatively still exists, therefore this method may give a false impression that the problem was solved. Furthermore, neither $||\mathbf{p}_{CB}||$, nor $||RC||$, nor P_{diff} were able to portray the difference in severity of the oscillations caused by the pressure predictor, leaving $\Delta||\nabla p||$ and C_{cb} as the only suitable quantification methods. Both $||RC||$ and P_{diff} may suffer from the fact that only $\tilde{\mathbf{p}}_c'$ is included in these terms instead of the full pressure. This may lead to compensation between higher oscillations and a lower value of $\tilde{\mathbf{p}}_c'$ between the cases with and without pressure predictors. Between $\Delta||\nabla p||$ and C_{cb}, C_{cb} is favorable because its value is normalised and dimensionless, making it easier to compare the quantity of checkerboarding between different cases.

In conclusion, a normalised, non-dimensional method that detects local oscillations and includes full pressure, regardless of time-integration method, is preferred, all of which can be found in C_{cb}.

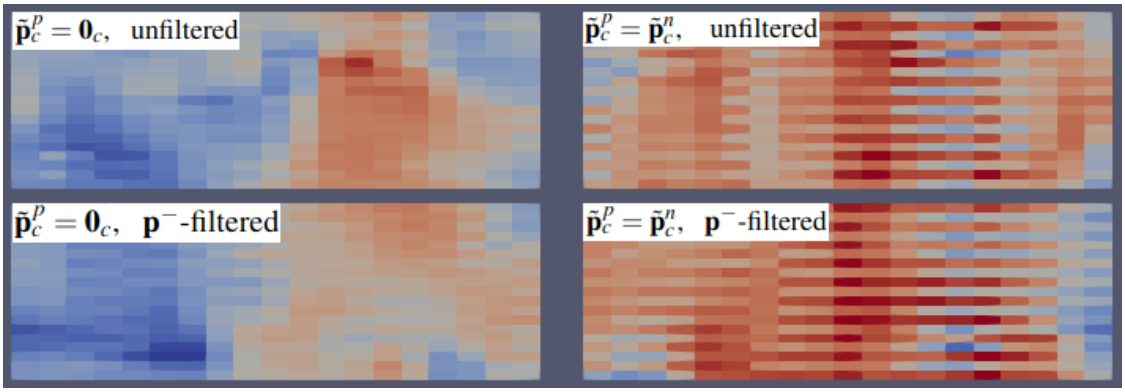


Figure 1: Part of y-plane (halved in x and z) of turbulent channel flows, at center height

3. Filtering checkerboarding conservatively

The choice of pressure predictor is of great influence on the level of checkerboarding [5]. The choice is usually made between $\tilde{\mathbf{p}}_c^p = \mathbf{0}_c$, or $\tilde{\mathbf{p}}_c^p = \tilde{\mathbf{p}}_c^n$, corresponding to the classical fractional step method and the Van Kan method [9]. Since using C_{cb} enables quantifying checkerboarding during runtime, and since the value is bounded between 1 (pressure consists of only kernel modes) and 0 (perfectly smooth field), this value can be used to set an interactive, self-regulating pressure predictor. To do so, the pressure predictor is rewritten as:

$$\mathbf{p}_c^p = \theta_p \mathbf{p}_c^n \tag{16}$$

Then, using $\theta_p = 1 - C_{cb}$, the pressure predictor will be self-regulated to diminish checkerboarding. If there is no checkerboarding present, this solver will use a full pressure predictor, reducing the pressure error from the order of $\mathscr{O}(\Delta t h^2)$ to $\mathscr{O}(\Delta t^2 h^2)$, whereas in the presence of checkerboarding, it will favour less oscillatory pressure fields at the cost of a higher pressure error. This new solver scheme was compared to $\theta_p = 0$ and $\theta_p = 1$, leading to the following solvers:

Solver	θ_0	θ_1	θ_{cb}
θ_p	0	1	$1 - C_{cb}$

These solvers were all tested with a channel flow at $Re_\tau = 180$ with the exact same domain, discretisations and solvers as for the results shown in figure 1.

4. Kinetic energy budgets

To analyse the turbulent channel flow, the turbulent kinetic energy budgets were monitored, aside from some of the lower order statistical quantities: $\overline{u_x}$, $u_{x,RMS}$, $u_{y,RMS}$, $u_{z,RMS}$ and $k = \overline{u_i' u_i'}$. $\bar{\cdot}$ is used to indicate temporal averaging and averaging in x and z directions, leading to quantities as a function of the channel height, y^+. The transport equation for turbulent kinetic energy is given by:

$$\frac{Dk}{Dt} = \partial_t k + \overline{u_j}\,\partial_j k = -\overline{u_i' u_j'}\partial_j \overline{u_i} - \overline{u_i' u_j' \partial_j u_i'} + 2\nu\partial_j\left(\overline{u_i' S_{ij}'}\right) - 2\nu\overline{S_{ij}' S_{ij}'} - \overline{u_i' \partial_i p'} \tag{17}$$

where $S_{ij} = \frac{1}{2}\left(\partial_j u_i + \partial_i u_j\right)$ gives the symmetric rate-of-strain tensor and Einstein summation convention was used to indicate summation over variables with the same indices. The terms on the RHS of equation (17) give the turbulent kinetic energy budget terms, production, P_k, transport, T_k, viscous diffusion, D_k^v, dissipation, ε_k and pressure diffusion, D_k^p, respectively. To accurately calculate these terms, it is important to use the same discrete operators as the ones found in the code of the solver, therefore the gradients are taken before field averaging in time and space, and not during post-processing. Moreover, the fluctuating terms in equation (17) can be rewritten to a combination of instantaneous fields and average fields, by using: $\phi' = \phi - \overline{\phi}$. By doing so, no fluctuating fields have to be calculated in the post-process nor do any of these fields have to be stored in memory. In conclusion, the budget terms were calculated using the following averaged components:

$$\begin{aligned} P_k &= -\left(\overline{u_i u_j} - \overline{u_i}\,\overline{u_j}\right)\overline{\partial_j u_i} \\ &= \overline{u_x u_y}\overline{\partial_y u_x} \end{aligned} \tag{18}$$

$$\begin{aligned} T_k &= \overline{u_i u_j \partial_j u_i} - \overline{u_i}\,\overline{u_j \partial_j u_i} - \overline{u_j}\,\overline{u_i \partial_j u_i} - \overline{u_i u_j}\,\overline{\partial_j u_i} + 2\overline{u_i}\,\overline{u_j}\,\overline{\partial_j u_i} \\ &= \overline{u_i u_j \partial_j u_i} - \overline{u_x}\,\overline{u_j \partial_j u_x} - \overline{u_x}\,\overline{u_i \partial_x u_i} - \overline{u_x u_y}\,\overline{\partial_y u_x} \end{aligned} \tag{19}$$

$$\begin{aligned} D_k^{\nu} &= \nu\overline{u_i \partial_j \partial_j u_i} - \nu\overline{u_i}\,\overline{\partial_j \partial_j u_i} - \varepsilon_k \\ &= \nu\overline{u_i \partial_j \partial_j u_i} - \nu\overline{u_x}\,\overline{\partial_j \partial_j u_x} - \varepsilon_k \end{aligned} \tag{20}$$

$$\begin{aligned} \varepsilon_k &= -\nu\overline{\partial_j u_i \partial_j u_i} + \nu\overline{\partial_j u_i}\,\overline{\partial_j u_i} \\ &= -\nu\overline{\partial_j u_i \partial_j u_i} + \nu\overline{\partial_y u_x}\,\overline{\partial_y u_x} \end{aligned} \tag{21}$$

$$\begin{aligned} D_k^{p} &= \overline{u_i \partial_i p} - \overline{u_i}\,\overline{\partial_i p} \\ &= \overline{u_i \partial_i p} \end{aligned} \tag{22}$$

where the second line of each of equation (18) to (22) gives the simplified version specific to the channel flow configuration where: $\overline{u_y} = 0$, $\overline{u_z} = 0$, $\partial_x \phi = 0$, $\partial_z \phi = 0$ and $\overline{\partial_i \phi} = \partial_i \overline{\phi}$. The remaining terms are all scalar values that can easily be averaged over time during runtime by the solver, greatly reducing the required memory.

5. Results

First impressions of the qualitative levels of checkerboarding for each case are given in figure 2. The domain was clipped at half the height of the channel, showing oscillatory patterns for all three solvers. In the center part of the image, corresponding to θ_1, the intensity can be seen to significantly higher than in the other cases. This suggests that the high C_{cb} led to a low θ_{cb} which self-regulated the occurrence of checkerboarding.

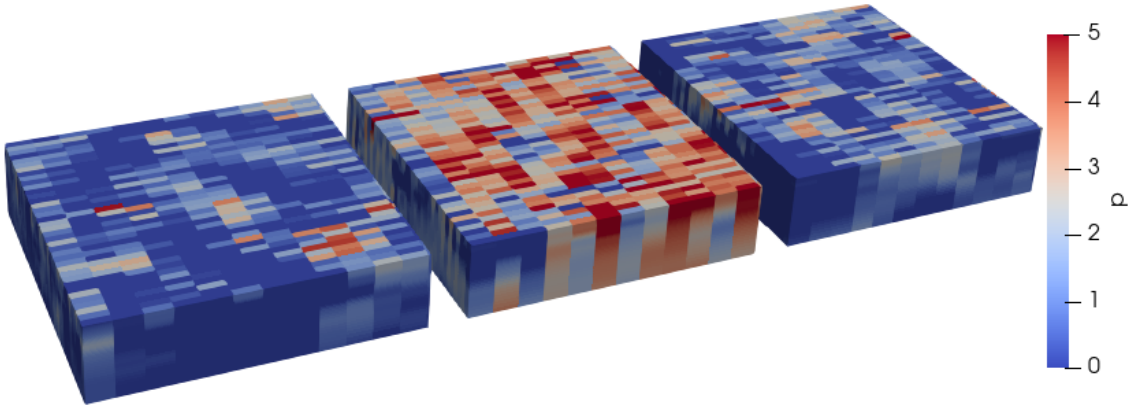


Figure 2: Pressure field at half-height of channel for θ_0 (left), θ_1 (center) and θ_{cb} (right). Visibly more checkerboarding present in $x-$ and $z-$directions in θ_1

These findings are confirmed by figure 3, in which C_{cb} is plotted over time. It is shown that solver θ_{cb} results in significantly less checkerboarding. To see if this mitigation of checkerboarding came at the cost of less accurate results, a quantification of the flow variables and budget terms has been performed and their results can be seen in figures 4 and 5, with reference data of [11]. All terms were non-dimensionalised by u_τ and Re_τ. With $u_\tau = \sqrt{\nu \left(\partial_y \overline{u_x}\right)_{wall}}$, which were found for each case to be:

Solver	θ_0	θ_1	θ_{cb}
u_τ	1.004	0.982	1.004

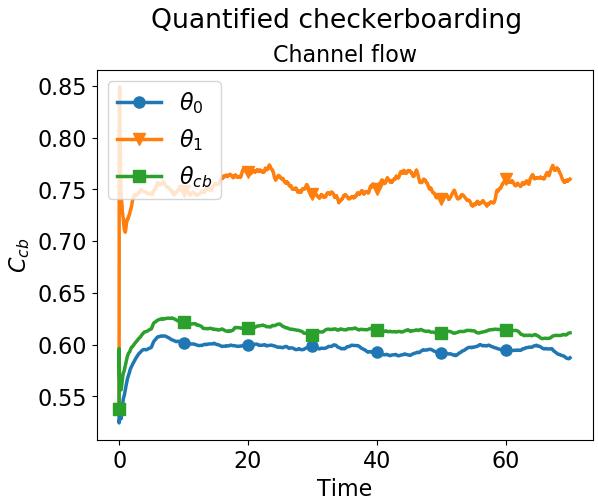
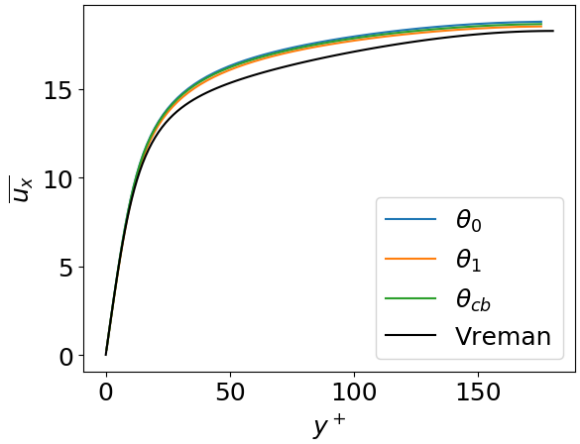
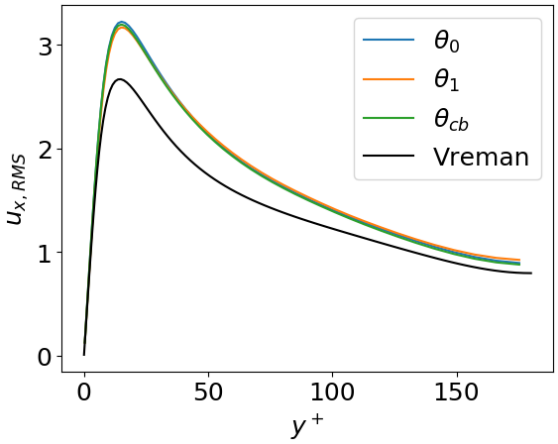


Figure 3: Checkerboard coefficient monitored over time, using a rolling average of C_{cb} over $\Delta T = 5$.

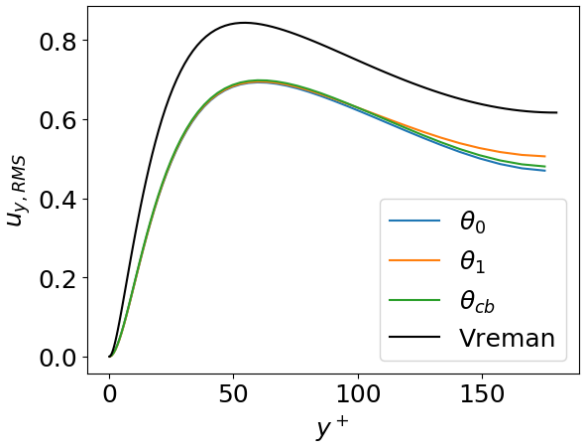
As can be seen from figure 4 the mean velocity closely resembles the reference value for all cases. The main difference with the reference data is that $u_{x,RMS}$ is slightly too high in each case, which in turn leads to a higher turbulent kinetic energy, k. All solvers show similar results, suggesting that the energy loss in θ_0 and θ_{cb} through the pressure error is not significantly higher.



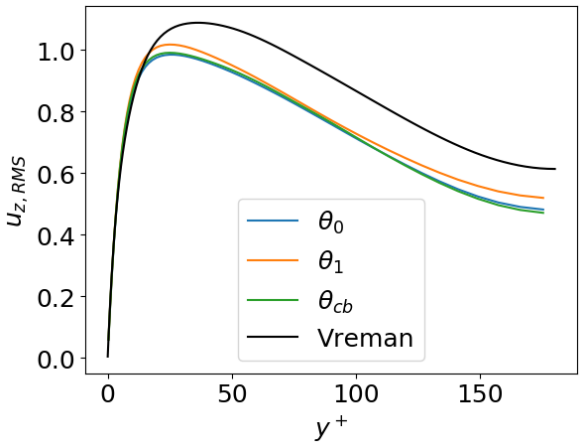
(a) Mean stream-wise velocity



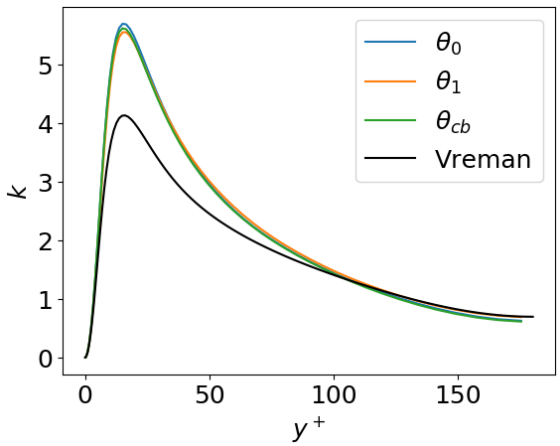
(b) Root mean square u_x



(c) Root mean square u_y



(d) Root mean square u_z



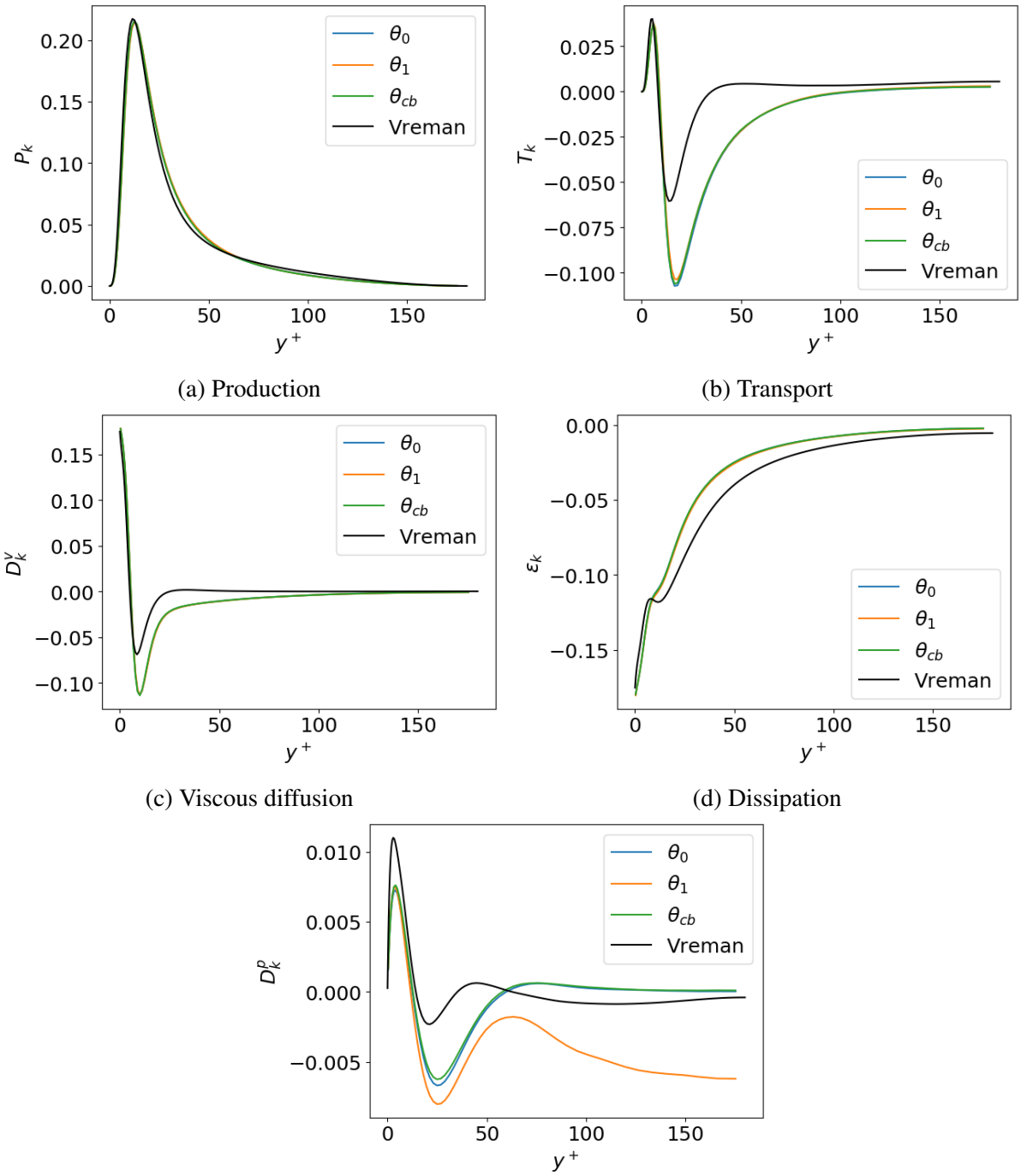
(e) Turbulent kinetic energy

Figure 4: Flow characteristics in channel flow at $Re_\tau = 180$ for each solver, compared to reference data of Vreman *et al.* [11]

From figure 5 it can be seen that the budgets show a close resemblance to the reference data of [11]. The production budget term is nearly identical and shows very little error for each case. The transport term shows very similar behaviour close to the wall with an underestimation for each case further away from the wall. This behaviour is also seen in the viscous diffusion. The dissipation budget shows that this underestimation of the previous terms is caused by an error in the dissipation term. The dissipation term is very sensitive to mesh resolution and especially on coarser grids, it is very difficult to capture the characteristic bump close to the wall. Finally, the pressure diffusion term shows resemblance for θ_0 and θ_{cb}, but θ_1 shows a large difference in the bulk. In conclusion, it can be seen that all solvers behave very similarly, and price for diminishing checkerboarding is not felt significantly. However, it is important to note that in cases with lower checkerboarding, θ_{cb} also benefits from the lower pressure error that θ_1 has, whereas in the results shown here, it is able to mitigate oscillatory pressure modes through self-regulation.

6. Conclusions

In this work, the problem of quantifying checkerboard properly was addressed. It was shown that inclusion of only pressure modes that lie on the kernel of the wide-stencil Laplacian is not enough to quantify local oscillations nor does it allow quantification of the checkerboard problem on complex geometries. To overcome these issues, the checkerboard coefficient, C_{cb}, was introduced and tested, showing results that coincide with qualitative judgements of oscillation patterns in a channel flow. Moreover, this coefficient has the advantage of being dimensionless and normalised. This last quality makes it perfectly suitable as a self-regulatory pressure predictor coefficient, which is able to increase or decrease depending on the level of quantified checkerboarding. Through this, one unified solver was developed that is able to mitigate checkerboarding in a turbulent channel flow case at $Re_\tau = 180$, while at the same time retaining its pressure error order of $\mathcal{O}(\Delta t^2 h^2)$ on cases with no occurrence of the problem. It was shown through turbulent kinetic energy budget analysis that this solver was effective in its desig whithout significant loss in accuracy. Its C_{cb} and θ_{cb} were found to nicely balance between the values for θ_0 and θ_1. In future work, the order of the pressure error will be examined closely through mesh and temporal refinement studies.



(a) Production

(b) Transport

(c) Viscous diffusion

(d) Dissipation

(e) Pressure diffusion

Figure 5: Turbulent kinetic energy budget terms in channel flow at $Re_\tau = 180$ for each solver, compared to reference data of Vreman *et al.* [11]

Acknowledgements

This work is supported by the FusionCAT project (001-P-001722) co-financed by the European Union Regional Development Fund within the framework of the ERDF Operational Program of Catalonia 2014-2020 with a grant of 50% of total cost eligible and the RETOtwin project (PDC2021-120970-I00) of *Ministerio de Economía y Competitividad*, Spain. J.A.H. is supported by the predoctoral grant FI 2022 (2022 FI_B1 00204) of the *Catalan Agency for Management of University and Research Grants (AGAUR)*. A.A.B. has also been supported by the predoctoral grants DIN2018-010061 and 2019-DI-90, given by MCIN/AEI/10.13039/501100011033 and the Catalan Agency for Management of University and Research Grants (AGAUR), respectively.

References

1. Durbin, P. & Reif, B. (2011). Statistical theory and modeling for turbulent flows. Wiley & Sons, Ltd, 2 ed.
2. Felten, F. N., & Lund, T. S. (2006). Kinetic energy conservation issues associated with the collocated mesh scheme for incompressible flow. Journal of Computational Physics, 215(2), 465-484.
3. Ferziger, J. H., Perić, M., & Street, R. L. (2002). Computational methods for fluid dynamics (Vol. 3, pp. 196-200). Berlin: springer.
4. Hopman, J. A., Trias, F. X., Rigola, J. (2022). On a conservative solution to checkerboarding: Examining the discrete Laplacian kernel using mesh connectivity. Proceedings of the 13th International ERCOFTAC Workshop on Direct and Large-Eddy Simulation (DLES13), Held at the University of Udine, October 2022.
5. Komen, E. M., Hopman, J. A., Frederix, E. M. A., Trias, F. X., & Verstappen, R. W. (2021). A symmetry-preserving second-order time-accurate PISO-based method. Computers & Fluids, 225, 104979.
6. Shashank, Larsson, J., & Iaccarino, G. (2010). A co-located incompressible Navier-Stokes solver with exact mass, momentum and kinetic energy conservation in the inviscid limit. Journal of Computational Physics, 229(12), 4425-4430.
7. Rhie, C. M., & Chow, W. L. (1983). Numerical study of the turbulent flow past an airfoil with trailing edge separation. AIAA journal, 21(11), 1525-1532.
8. 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.
9. Van Kan, J. J. I. M. (1986). A second-order accurate pressure-correction scheme for viscous incompressible flow. SIAM journal on scientific and statistical computing 7(3), pp. 870-891.
10. Verstappen, R. W. C. P., & Veldman, A. E. P. (2003). Symmetry-preserving discretization of turbulent flow. Journal of Computational Physics, 187(1), 343-368.
11. 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).