Transport of passive scalars in incompressible flows using a coarse-grid projection method

A. Kashefi

Received: date / Accepted: date

Abstract  Coarse Grid Projection (CGP) methodology is used to accelerate the computations of sets of decoupled nonlinear evolutionary and linear static equations. In CGP, the linear equations are solved on a coarsened mesh compared to the nonlinear equations, leading to a reduction in central processing unit (CPU) time. The accuracy of the CGP scheme has been assessed for the advection-diffusion equation along with the pressure Poisson equation. Here we add another decoupled equation to this set: the conservation equation of a passive scalar field. In this article, we examine the influence of CGP methodology for the first time on the passive scalar field. To this purpose, a semi-implicit-time-integration unstructured-triangular-finite-element CGP version is selected. The temperature variable is used as the passive scalar. The CGP platform is validated with the test case of the flow over a circular cylinder with the condition of constant cylinder temperature. For different Prandtl numbers, we compare the CGP and non-CGP configurations according to the Nusselt number and the spatial structure of the scalar field obtained. CGP is able to maintain excellent to reasonable accuracy of the passive scalar field, while achieves speedup factors ranged from 3.282 to 3.694. The phase lag between the standard and CGP approaches is transmitted from the velocity field into the temperature field, and thus into the local transient Nusselt number. For one and two levels of coarsening, the numerical predictions by CGP for the unsteady local heat transfer coefficients agree well with available data in the literature.

Keywords  Coarse grid projection · Transport of passive scalars · Multiresolution methods · Pressure-correction schemes · Flow past a cylinder

1 Introduction

Pressure projection schemes are widely used for the unsteady incompressible flow computations [1–4]. Taking the advantages of these techniques, the saddle-point issue of the continuity and momentum equations disappears [3, 4]. Hence, one only deals with two decoupled cascading elliptic equations: the advection-diffusion equation and the pressure Poisson one. Different multigrid schemes have been already introduced to lessen the computational times associated with the numerical pressure correction methods (see e.g., Refs. [5–9]). Coarse Grid Projection (CGP) methodology is a recently used multiresolution scheme to accelerate these computations [10–14]. CGP saves a considerable amount of CPU time by reducing the degree of freedom for the discretized Poisson equation, which is the most time consuming subproblem. Accordingly, the nonlinear advection-diffusion equation is solved on a fine grid and the linear pressure Poisson equation is solved on a corresponding coarsened grid. Mapping functions transfer data between the grids. The CGP procedure is described in detail in Sect. 2.2.

In 2010 Lentine et al. [10] first introduced CGP for accelerating inviscid flow computations. In 2013 San and Staples [11] used the CGP technique for the numerical simulation of the incompressible Navier-Stokes equations. In 2014 the CGP algorithm was used in the fast fluid dynamics (FFD) models by Jin and Chen [12]. In 2016 Kashefi and Staples [13] applied the CGP framework to incremental pressure correction...
schemes. In 2017 a finite element version of CGP with a semi-implicit time integration scheme was presented by Kashefi and Staples [14].

In all the literature cited above, the authors [10–14] studied the performance of CGP in terms of the level of accuracy obtained in the velocity or pressure fields and achieved speedup factors. Nonetheless, the influence of the CGP algorithm on a passive scalar field has not yet been investigated.

For small scalar differences, the conservation equation of a passive scalar can be independently solved for a given velocity field [15]. In this case, the study of transporting the passive scalars in a numerical simulation performed by the CGP technique is important in two aspects. First, since the nonlinear convection term in the conservation equation is based on the velocity field obtained by CGP, preserving the accuracy level of the passive scalar field should be investigated. Second, in order to obtain the velocity and passive scalar fields using pressure projection schemes, one has to deal with three decoupled elliptic equations at each time step: a linearized Poisson equation for the pressure field, and a linearized equation for the intermediate velocity field, as well as a linearized equation for the passive scalar field. Hence, the contribution of the CGP scheme to accelerating the computations becomes significant.

An external unsteady flow past a cylinder is a physically meaningful benchmark case and a model for canonical studies of demanding fluid mechanics problems [16]. Thus, this test case is solved for different Prandtl numbers in order to investigate the performance of the CGP strategy for the simulation of the conservation equation.

The rest of this article is structured as follows. Section 2.1 gives the governing equations for incompressible flows and conservation of passive scalar fields. We discuss coarse-grid projection methodology in Sect. 2.2. Computational aspects of the problem are described in Sect. 2.3. Numerical results and their relevant discussions are presented in Sect. 3. Conclusions and notes for extensions of the work are provided in Sect. 4.

\section*{2 Problem formulation}

\subsection*{2.1 Governing equations}

The equations of conservation of momentum, mass, and passive scalars for an incompressible flow of a Newtonian fluid are given by

\begin{equation}
\nabla \cdot \mathbf{u} = 0 \text{ in } V,
\end{equation}

\begin{equation}
\mathbf{u} = \mathbf{u}_{\Gamma_D} \text{ on } \Gamma_D,
\end{equation}

\begin{equation}
-p \mathbf{n} + \mu \nabla \mathbf{u} \cdot \mathbf{n} = \mathbf{t}_{\Gamma_N} \text{ on } \Gamma_N,
\end{equation}

\begin{equation}
\rho \left[ \frac{\partial \theta}{\partial t} + (\mathbf{u} \cdot \nabla) \theta \right] = \frac{k}{c_p} \Delta \theta \text{ in } V,
\end{equation}

\begin{equation}
\theta = \theta_{\Omega_D} \text{ on } \Omega_D,
\end{equation}

\begin{equation}
\nabla \theta \cdot \mathbf{n} = \mathbf{b}_{\Omega_N} \text{ on } \Omega_N,
\end{equation}

where \( \mathbf{u} \) is the velocity vector, \( p \) stands for the pressure, and \( \theta \) represents the temperature of the fluid in domain \( V \). \( \mathbf{f} \) is the vector of external force. \( \mathbf{t}_{\Gamma_N} \) and \( \mathbf{b}_{\Omega_N} \) denote the stress vectors applied to the velocity and temperature fields, respectively. \( \rho \) is the fluid density and \( \mu \) is the dynamic viscosity. \( k \) is the conductivity of the fluid and \( c_p \) is the specific heat at a constant pressure. \( \Gamma_D \) and \( \Gamma_N \) respectively represent the velocity Dirichlet and Neumann boundaries, while \( \Omega_D \) and \( \Omega_N \) respectively denote the temperature Dirichlet and Neumann boundaries of the domain \( V \). \( \mathbf{n} \) is the outward unit vector normal to the boundaries. There is no overlapping between \( \Gamma_D \) and \( \Gamma_N \) subdomains. Similarly no overlap exists between \( \Omega_D \) and \( \Omega_N \) subdomains as well.

We discretize the system of equations using a first-order semi-implicit time integration formula [17]. Then, we apply a non-incremental pressure correction scheme [4] to the time-discretized system, yielding to

\begin{equation}
\rho \left[ \frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\delta t} + (\mathbf{u}^n \cdot \nabla) \tilde{\mathbf{u}}^{n+1} \right] - \mu \Delta \tilde{\mathbf{u}}^{n+1} = \mathbf{f}^{n+1} \text{ in } V,
\end{equation}

\begin{equation}
\tilde{\mathbf{u}}^{n+1} = \mathbf{u}_{\Gamma_D}^{n+1} \text{ on } \Gamma_D,
\end{equation}

\begin{equation}
\mu \nabla \tilde{u}^{n+1} \cdot \mathbf{n} = \mathbf{t}_{\Gamma_N}^{n+1} \text{ on } \Gamma_N,
\end{equation}

\begin{equation}
\Delta p^{n+1} = \frac{\rho}{\delta t} \nabla \cdot \tilde{\mathbf{u}}^{n+1} \text{ in } V,
\end{equation}
\[ \nabla p^{n+1} \cdot n = 0 \quad \text{on } \Gamma_D, \]  
\[ p^{n+1} = 0 \quad \text{on } \Gamma_N, \]  
\[ u^{n+1} = u^{n+1} - \frac{\delta t}{\rho} \nabla p^{n+1} \quad \text{in } V, \]  
\[ \rho \left[ \frac{\theta^{n+1} - \theta^n}{\delta t} + (u^{n+1} \cdot \nabla) \theta^{n+1} \right] = k \Delta \theta^{n+1} \quad \text{in } V, \]  
\[ \theta^{n+1} = \theta^{n+1} \quad \text{on } \Omega_D, \]  
\[ \nabla \theta^{n+1} \cdot n = b^{n+1} \quad \text{on } \Omega_N, \]  

where \( \delta t \) represents the time step and \( \tilde{u} \) is the intermediate velocity vector. For a more detailed description of the pressure projection scheme implemented here, one may refer to Refs. [1–4].

The finite-element Galerkin scheme [3, 18] with the piecewise linear basis function \( P_1 \) is used to spatially discretize the space of the velocity, pressure, and temperature fields. The finite-element form of Eqs. (8)–(17) is expressed as

\[ \frac{1}{\delta t} \left( M_v \tilde{U}^{n+1} - M_v U^n \right) + \left[ N^n + L_v \right] \tilde{U}^{n+1} = M_v F^{n+1}, \]  
\[ L_p P^{n+1} = \frac{\rho}{\delta t} D \tilde{U}^{n+1}, \]  
\[ M_v U^{n+1} = M_v \tilde{U}^{n+1} - \delta t G P^{n+1}, \]  
\[ \frac{1}{\delta t} \left( M_\theta \Theta^{n+1} - M_\theta \Theta^n \right) + \left[ N^n + L_\theta \right] \Theta^{n+1} = Q^{n+1}, \]

where \( M_v, M_\theta, L_v, L_\theta, D, \) and \( G \) denote the matrices associated, respectively, to the velocity mass, temperature mass, velocity laplacian, pressure laplacian, temperature laplacian, divergence, and gradient operators. \( N^n \) and \( N^{n+1} \) indicate the nonlinear convection operators at time \( t^n \) and \( t^{n+1} \), respectively. The vectors \( \tilde{U}^{n+1}, U^{n+1}, \Theta^{n+1}, P^{n+1}, F^{n+1}, \) and \( Q^{n+1} \) represent the nodal values of the intermediate velocity, the end-of-step velocity, the temperature, the pressure, the forcing term on the velocity domain, and the stress term on the temperature domain at time \( t^{n+1} \), respectively.

2.2 Coarse grid projection methodology

In the CGP scheme, first we balance the advection-diffusion equation on a fine grid and obtain the intermediate velocity field data \( \tilde{U}_f^{n+1} \). Then, we restrict \( \tilde{U}_f^{n+1} \) to a corresponding coarsened grid and set \( \tilde{U}_c^{n+1} \). We take the divergence of the restricted intermediate velocity \( \tilde{U}_c^{n+1} \) in order to set the source term of the pressure Poisson equation. We solve the Poisson equation on the coarsened grid and obtain \( P_c^{n+1} \). In the next stage, we prolong the resulting pressure data \( P_c^{n+1} \) from the coarse grid to the fine grid and set \( P_f^{n+1} \). We correct the velocity domain and obtain \( U_c^{n+1} \) on the fine grid.

Now, we create the nonlinear convection matrix \( N^{n+1} \) based to the obtained velocity field data \( U_f^{n+1} \). Finally, we solve the last conservation equation for the passive scalar field \( \Theta_f^{n+1} \) on the fine grid.

In practice we consider four nested spaces: \( V_1 \subseteq V_2 \subseteq V_3 \subseteq V_4 = V \). We uniformly subdivide each triangular element of the discretized space of \( V_l \) (for \( 1 \leq l \leq 3 \)) into four triangles. This procedure provides the discretized space of \( V_{l+1} \). Hence, for each CGP simulation we have a fine mesh and a corresponding coarsened mesh respectively with \( M \) and \( N \) elements such that \( N = 4^{-k} M \), where \( k \) indicates the coarsening level. The restriction \( R : V_4 \rightarrow V_{4-l} \) and prolongation \( P : V_l \rightarrow V_{l+1} \) operators and their matrix representations, respectively \( R_{l+1}^{4} \) and \( P_{l+1}^{4} \), are constructed using Geometric Multigrid (GMG) tools. \( R_{l+1}^{4} \) injects the intermediate velocity data from a fine grid \( (V_4) \) to the corresponding coarse grid \( (V_{4-l}) \). \( P_{l+1}^{4} \) corresponds to the finite-element shape functions. Since we implement in this study, \( P_{l+1}^{4} \) prolongs the pressure data from the coarse grid \( (V_l) \) to the next nested space \( (V_{l+1}) \) using a linear interpolation. Finally, we derive the pressure laplacian \( L_p \) and divergence \( D \) operators on a relatively coarse mesh \( (V_{4-l}) \) by taking the inner products of the coarse grid finite-element shape functions. One may see Sect. 2.3 of Ref. [14] for further details.

Eqs. (22)–(27) summarize the CGP algorithm at each time step, \( \delta t \), of the simulation.

1. Calculate \( \tilde{U}_f^{n+1} \) on \( V \) by solving

\[ (M_v + \delta t N^n + \delta t L_v) \tilde{U}_f^{n+1} = \delta t M_v F^{n+1} + M_v U_f^n. \]  

2. Map \( \tilde{U}_f^{n+1} \) onto \( V_{4-l} \) and obtain \( \tilde{U}_c^{n+1} \) via

\[ \tilde{U}_c^{n+1} = R_{4-l}^{4} \tilde{U}_f^{n+1}. \]  

3. Calculate \( P_c^{n+1} \) on \( V_{4-l} \) by solving

\[ L_p P_c^{n+1} = \frac{\rho}{\delta t} D \tilde{U}_c^{n+1}. \]
4. Remap $P^{n+1}_c$ onto $V$ and obtain $P^{n+1}_f$ via

$$P^{n+1}_f = P^{n+1}_l P^{n+1}_c.$$  \hfill (25)

5. Calculate $U^{n+1}_f$ via

$$M_u U^{n+1}_f = M_u U^{n+1}_f - \delta t G P^{n+1}_f.$$  \hfill (26)

6. Build up the nonlinear convection operator $N^{n+1}$ using the obtained velocity field $U^{n+1}_f$.

7. Calculate $\Theta^{n+1}_f$ on $V$ by solving

$$(M_\theta + \delta t N^{n+1} + \delta t L_\theta) \Theta^{n+1}_f = \delta t Q^{n+1} + M_\theta \Theta^{n+1}_f.$$  \hfill (27)

### 2.3 Computational consideration

We employ an in-house C++ object oriented code. We use the $ILU(0)$ preconditioned GMRES($m$) algorithm \cite{19, 20}. The Gmsh application \cite{21} is used for generating unstructured finite element meshes. All simulations are performed on a single Intel(R) Xeon(R) processor with 2.66 GHz clock rate and 64 Gigabytes of RAM.

### 3 Results and discussion

We consider a rectangular computational domain $V := [0, 38] \times [0, 32]$. The cylinder is represented by a circle with diameter $d$ in two dimensions. The center of the circle lies at the point $(8, 16)$. At the inflow boundary, we impose a free stream velocity $u_\infty$ perpendicular to the vertical axis, while the outflow boundary is described with a natural Neumann condition

$$\mu \nabla u \cdot n = 0.$$  \hfill (28)

The velocity at the top and bottom of the field is perfectly slipped with the magnitude and direction of $u_\infty$. The circle is considered as a rigid body with no-slip conditions. For the temperature we take boundary conditions from Ref. \cite{16} such that $\theta = \theta_w$ is enforced at the circle, while $\theta = \theta_\infty$ is imposed at the remaining boundaries. The conditions correspond to a problem with constant cylinder temperature. The Reynolds number is expressed as

$$Re = \frac{\rho d u_\infty}{\mu},$$  \hfill (29)

and the Prandtl number is determined as

$$Pr = \frac{c_p \mu k}{k}.$$  \hfill (30)

We determine each point on the circular cylinder surface by the angle $\alpha$ from the negative $x$-axis. Thus, the local Nusselt number on the cylinder surface is formulated as

$$Nu_\alpha = -d \frac{\partial \theta}{\partial n}|_\alpha.$$  \hfill (31)

The time-averaged Nusselt number per time cycle, $t_p$, is expressed by

$$Nu = \frac{1}{t_p} \int_0^{t_p} N u_\alpha dt.$$  \hfill (32)

And the time- and space-averaged Nusselt number is calculated by

$$\overline{Nu} = \frac{1}{t_p} \int_0^{t_p} \frac{1}{2\pi} \int_0^{2\pi} N u_\alpha d\alpha dt.$$  \hfill (33)

The density ($\rho$), free stream velocity ($u_\infty$), specific heat ($c_p$), cylinder temperature ($\theta_w$), and cylinder diameter ($d$) are set to 1.00; and the temperature at infinity ($\theta_\infty$) is set to 0.00 in the International Unit System. The viscosity ($\mu$) and conductivity ($k$) of the fluid vary to set the Reynolds and Prandtl numbers. A fixed time step of $\delta t = 0.05$ s is chosen and we execute the numerical simulations until time $t = 150$ s.

Notations in the form of $M : N$ demonstrate the grid resolutions of the advection-diffusion and passive scalar equation solvers, $M$ elements, and the pressure Poisson equation solver, $N$ elements.

The Poisson solver uses the meshes with 108352 nodes and 215680 elements, 27216 nodes and 53920 elements, 6868 nodes and 13480 elements, and 1749 nodes and 3370 elements, respectively, for $k = 0$, $k = 1$, $k = 2$, and $k = 3$. Figure 1 shows those grids for $k = 2$ and $k = 3$.

The detailed results related to the velocity field were presented in one of our previous works (see Sect. 3.3 of Ref. \cite{14}) and are not repeated here again.

Table 1 lists the CPU times devoted to each subproblem and speedup factors achieved for the simulations with several spatial resolutions. The most time-consuming component of the simulations with standard resolutions (215680:215680, 53920:53920, 13480:13480, and 3370:3370) is the Poisson equation. Taking the advantages of the CGP method into account, the price of the Poisson equation portion becomes less than 1.2% only for two levels ($k = 2$) of coarsening. The maximum achieved speedup is a factor of 3.694. In practice, one must solve the linear system of Eq. (22) and Eq. (27) to compute respectively the intermediate velocity field ($U^{n+1}_f$) and the temperature field ($\Theta^{n+1}_f$). From a numerical linear algebra point of view, the ($M_u + \delta t N^{n+1} + \delta t L_u$) and ($M_\theta + \delta t N^{n+1} + \delta t L_\theta$) matrices are similar to each other. Hence, the value of the ratio of the computational cost to the number of unknowns...
is the same for the both systems. The nodal value of the velocity field is twice the nodal value of the temperature field. And this is why for all the simulations with and without the CGP technique, the computational cost of the advection-diffusion equation is roughly twice as much as the cost of the conservation equation of the temperature field. As discussed earlier, the prolongation and restriction operators are constructed based on the idea proposed in Sect. 2.3 of Ref. [14]. Following the data structure introduced in Ref. [14], the numerical expense of the mapping part becomes insignificant, as can be seen from Table 1.

Figure 2 visually compares the temperature fields obtained for $Re = 100$ and $Pr = 2$ with and without CGP for different grid resolutions at time $t = 150$ s. The temperature fields obtained by the CGP procedure for one level ($k = 1$) and two levels ($k = 2$) of the Poisson grid coarsening are close to that simulated with the standard full grid resolution (215680: 215680). For three ($k = 3$) levels of coarsening; however, a considerable reduction in the fidelity of the temperature field is observed. Nonetheless, the resulting field of CGP with the 215680:3370 spatial resolution is still better than those that are performed on the standard full coarse grid resolution (3370:3370). Similar observation is reported by Kashefi and Staples [14] for the velocity field (see e.g., Figs. 9-10 of Ref. [14]).

From a general point of view, the spatial discretization of the advection-diffusion domain acts as a low-pass filter on the grid, and the Poisson solver also acts as a pre-filtering process [11]. The CGP procedure specifically uses the belief in order to increase saving in computational time without negatively affecting the properly-resolved advection-diffusion field, and consequently the temperature field. A visual demonstration of these effects is displayed in Fig. 3. Figure 3 depicts the temperature distributions along the horizontal centerline in the wake region behind the cylinder for $Re = 100$ and $Pr = 2$ at time $t = 150$ s. While the outputs of the pure coarse grid are contaminated by spurious fluctuations at the end of the fluid domain, these fluctuations are filtered in the temperature field obtained by the CGP framework.

There could be a phase lag between the velocity outputs of the standard and CGP approaches, depending on the time increment ($\delta t$) [14]. As can be seen in Fig. 2 and Fig. 3, these phase lags are transmitted from the velocity field into the temperature one.

Figure 4 compares the local Nusselt number ($Nu_\alpha$) computed for $Re = 100$ and $Pr = 0.5$ by the standard and CGP algorithms at two different angles of $\alpha = \pi$ and $\alpha = 5\pi/3$. For one ($k = 1$, 215680:53920) and two ($k = 2$, 215680:13480) levels of the Poisson grid coarsening, the CGP results are close to the outputs of the full fine scale simulation ($k = 0$, 215680:215680). More importantly, they are significantly more accurate than the local Nusselt number ($Nu_\alpha$) computed on the full coarse grids ($k = 0$, 53920:53920 and $k = 0$, 13480:13480). For one level ($k = 1$) of mesh coarsening, there is a phase lag between the local Nusselt number.
Fig. 2 Temperature fields for the flow over a circular cylinder for $Re = 100$ and $Pr = 2$ at $t = 150$ s. Labels in the form of $M : N$ illustrate the grid resolutions of the advection-diffusion and passive scalar fields, $M$ elements, and the pressure field, $N$ elements.

Fig. 3 Comparison of the temperature in the wake region of the flow past a cylinder for different grid resolutions for $Re = 100$ and $Pr = 2$ at $t = 150$ s for a $k = 0, 1$, and 0; b $k = 0, 2$, and 0. Labels in the form of $M : N$ illustrate the grid resolutions of the advection-diffusion and passive scalar fields, $M$ elements, and the pressure field, $N$ elements.
Transport of passive scalars in incompressible flows using a coarse-grid projection method

4 Conclusions and future directions

In this article, we applied the novel CGP multigrid scheme to the pressure Poisson equation in order to obtain a numerical solution to the velocity and temperature fields. We solved a two-dimensional flow past a

Table 1 Total CPU times and their component percentages: the advection-diffusion equation, the pressure Poisson equation, the conservative equation of the passive scalar field, and the mapping function, for the simulation with the Reynolds number of \( Re = 100 \) and the Prandtl number of \( Pr = 2 \). \( M : N \) represents the grid resolution of the advection-diffusion and the passive scalar solvers (\( M \) elements), and Poisson’s equation (\( N \) elements).

| \( k \) | Resolution | \%Adv-Diff | \%Poisson | \%Passive | \%Map | CPU (s) | Speedup |
|---|---|---|---|---|---|---|---|
| 0 | 215680:215680 | 18.25 | 72.61 | 9.14 | 0.00 | 810363.0 | 1.000 |
| 1 | 215680:53920 | 56.513 | 15.185 | 28.301 | 0.01 | 246887.0 | 3.282 |
| 2 | 215680:13480 | 65.845 | 1.171 | 32.983 | 0.01 | 221521.0 | 3.658 |
| 3 | 215680:3370 | 66.524 | 0.102 | 33.373 | 0.01 | 219351.0 | 3.694 |
| 0 | 53920:53920 | 19.976 | 69.962 | 10.062 | 0.00 | 45032.3 | 1.000 |
| 0 | 13480:13480 | 18.248 | 72.486 | 9.266 | 0.00 | 3848.9 | 1.000 |
| 0 | 3370:3370 | 8.889 | 86.922 | 4.189 | 0.00 | 51.1 | 1.000 |

\((Nu_a)\) obtained by the standard and CGP simulations at both angles of \( \alpha = \pi \) and \( \alpha = 5\pi/3 \) (see Fig. 4a and Fig. 4c). On the other hand, there is no phase lag between the standard and CGP outputs for two levels \((k=2)\) of grid coarsening neither at angle of \( \alpha = \pi \) nor at \( \alpha = 5\pi/3 \) (see Fig. 4b and Fig. 4d). Comparing these results with the temperature fields presented in Fig. 3, we experience the same observation. From a mathematical point of view, this phenomena is expected since the local Nusselt number \((Nu_a)\) is proportional to the normal derivative of the temperature variable, and the phase of a continuous bounded oscillatory function gets transmitted to its derivative.

As can be seen in Fig. 4, for two levels \((k=2)\) of the Poisson grid coarsening, the local Nusselt number \((Nu_a)\) predicted by the CGP approach is slightly underestimated at angle of \( \alpha = \pi \) in comparison with the outcomes of the full fine scale simulation (215680:215680). We observe, in contrast, an over prediction at angle of \( \alpha = 5\pi/3 \). It is because the architecture of the triangular unstructured grids established in this zone. Note that we use the simplest strategy for generating the unstructured grids, while using advanced techniques for mesh generation can significantly affect the performance of GMG tools like CGP.

The data collected in Table 2 demonstrates that the prediction of the time- and space-averaged Nusselt number \((\overline{Nu})\) by the CGP technique is generally reliable and has an excellent agreement with the correlations reported in the literature [23, 24]. By increasing the Prandtl number \((Pr)\), a deviation from the full fine scale computation occurs. In fact at low Prandtl numbers \((Pr)\) the linear term \((\Delta\theta^{n+1})\) dominates the nonlinear term \((u^{n+1} \cdot \nabla \theta^{n+1})\) in the conservation equation (Eq. (15)) of the temperature. Since a linear mapping function is used here, a higher level of accuracy is obtained at a lower Prandtl number \((Pr)\). Taking the advantages of more advanced data interpolation schemes (see e.g., Ref. [25]) can be a solution to this issue. However, even using the simple extrapolation technique, the outcomes of the CGP configuration are still more accurate than the resulting data captured from the full coarse scale simulations.
Fig. 4 The local Nusselt number around the cylinder for $Re = 100$ and $Pr = 0.5$ with and without the CGP algorithm for (a) one level coarsening ($k = 1$) at angle $\alpha = \pi$; (b) two levels coarsening ($k = 2$) at angle $\alpha = \pi$; (c) one level coarsening ($k = 1$) at angle $\alpha = 5\pi/3$; (d) two levels coarsening ($k = 2$) at angle $\alpha = 5\pi/3$. Legends in the form of $M : N$ indicate the spatial resolutions of the advection-diffusion and passive scalar fields, $M$ elements, and the pressure field, $N$ elements.

Table 2 Prediction of the time- and space-averaged Nusslet number $\overline{Nu}$ for the Reynolds number of $Re = 100$ for different spatial resolutions. The error percentages are measured with reference to the finest grid resolution. $M : N$ demonstrates the grid resolution of the advection-diffusion and the passive scalar solvers ($M$ elements), and Poisson’s equation ($N$ elements).

| $k$ | Resolution     | $Pr = 0.5$ | %Error | $Pr = 3.0$ | %Error |
|-----|----------------|------------|--------|------------|--------|
| 0   | 215680:215680  | 4.4162     | –      | 9.5254     | –      |
| 1   | 215680:53920   | 4.3723     | 0.994  | 9.1850     | 3.574  |
| 2   | 215680:13480   | 4.2350     | 4.103  | 7.5469     | 20.770 |
| 0   | 53920:53920    | 4.3102     | 2.400  | 8.7447     | 8.195  |
| 0   | 13480:13480    | 4.0242     | 8.876  | 7.3741     | 22.585 |
|     | Fand [23]      | 4.8070     | –      | 8.2285     | –      |
|     | Churchill and Bernstein [24] | 3.5908 | – | 8.7382 | – |
circular cylinder with constant temperature. Concerning the performance of CGP, the speedup factors ranged from 3.282 to 3.694. For one and two levels of the Poisson grid coarsening, the structure of the von Karman street and heat transfer coefficients were in excellent agreement with those simulated using pure fine grid computations. However, only a reasonable level of accuracy was obtained for three levels of the Poisson mesh coarsening.

The objective of our future research is to perform a comparison between the CGP approach with one level of coarsening \((k = 1)\) and the standard finite element algorithm with Taylor-Hood mixed finite elements \(P_2/P_1\) (see e.g., Ref. [18]). From a grid resolution point of view, for an assumed number of grid points of the velocity component, the Poisson solver utilizes a space with an equal pressure node numbers, discretized using either the CGP \((k = 1)\) method or Taylor-Hood elements. In this sense, a detailed investigation of the similarity/difference between these two concepts may introduce novel mapping functions for the CGP tool.

Acknowledgements AK would like to thank Dr. Peter Minev and Dr. Omer San for helpful discussions.

References

1. Chorin AJ (1968) Numerical solution of the Navier-Stokes equations. Mathematics of computation 22 (104):745-762
2. Temam R (1969) Sur l’approximation de la solution des quations de Navier-Stokes par la mthode des pas fractionnaires (II). Archive for Rational Mechanics and Analysis 33 (5):377-385
3. Timmermans L, Minev P, Van De Vosse F (1996) An approximate projection scheme for incompressible flow using spectral elements. International journal for numerical methods in fluids 22 (7):673-688
4. Guermond J, Minev P, Shen J (2006) An overview of projection methods for incompressible flows. Computer methods in applied mechanics and engineering 195 (44):6011-6045
5. Reusken A (1995) Fourier analysis of a robust multigrid method for convection-diffusion equations. Numerische Mathematik 71 (3):365-397
6. Filelis-Papadopoulos CK, Gravvanis GA, Lipitakis EA (2014) On the numerical modeling of convection-diffusion problems by finite element multigrid preconditioning methods. Advances in Engineering Software 68:56-69
7. Gupta MM, Kouatchou J, Zhang J (1997) A compact multigrid solver for convection-diffusion equations. Journal of Computational Physics 132 (1):123-129
8. Gupta MM, Kouatchou J, Zhang J (1997) Comparison of second-and fourth-order discretizations for multigrid Poisson solvers. Journal of Computational Physics 132 (2):226-232
9. Zhang J (1998) Fast and high accuracy multigrid solution of the three dimensional Poisson equation. Journal of Computational Physics 143 (2):449-461
10. Lentine M, Zheng W, Fedkiw R A novel algorithm for incompressible flow using only a coarse grid projection. In: ACM Transactions on Graphics (TOG), 2010. vol 4. ACM, p 114
11. San O, Staples AE (2013) A coarse-grid projection method for accelerating incompressible flow computations. Journal of Computational Physics 233:489-508
12. Jin M, Liu W, Chen Q (2014) Accelerating fast fluid dynamics with a coarse-grid projection scheme. HVAC&R Research 20 (8):932-943
13. Kashefi A, Staples AE (2016) Acceleration of incremental-pressure-correction incompressible flow computations using a coarse-grid projection method. Bulletin of the American Physical Society 61

http://meetings.aps.org/Meeting/DFD16/Session/L29.1
14. Kashefi A, Staples AE (2017) A finite-element coarse-grid projection method for incompressible flow simulations. Advances in Computational Mathematics https://doi.org/10.1007/s10444-017-9573-5
15. Kim J, Moin P (1987) Transport of passive scalars in a turbulent channel flow. Proc. 6th Symp. on Turbulent Shear Flows, Toulouse, France, 79 September, pp. 5-2-15-2-6.
16. Karniadakis G (1988) Numerical simulation of forced convection heat transfer from a cylinder in crossflow. International Journal of Heat and Mass Transfer 31:107-118
17. Wanner G, Hairer E (1991) Solving ordinary differential equations II, vol 1. Springer-Verlag, Berlin
18. Reddy JN (1993) An introduction to the finite element method, vol 2. vol 2.2. McGraw-Hill New York
19. Saad Y, Schultz MH (1986) GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal on scientific and statistical computing 7 (3):856-869
20. Van der Vorst HA (2003) Iterative Krylov methods for large linear systems, vol 13. Cambridge University Press
21. Geuzaine C, Remacle JF (2009) Gmsh: A 3D finite element mesh generator with builtin preand postprocessing facilities. International Journal for Numerical Methods in Engineering 79 (11):1309-1331
22. Patnaik V, Narayana A, Seetharamu K (1999) Numerical simulation of vortex shedding past a circular cylinder under the influence of buoyancy. International Journal of Heat and Mass Transfer 42:3495-3507
23. Fand RM (1965) Heat transfer by forced convection from a cylinder to water in crossflow. International Journal of Heat and Mass Transfer 8:995-1010
24. Churchill SW, Bernstein M (1977) A correlating equation for forced convection from gases and liquids to a circular cylinder in crossflow. Trans. ASME, Ser. C, J. Heat Transfer 99:300-306
25. Amsallem D, Farhat C (2008) Interpolation method for adapting reduced-order models and application to aeroelasticity. AIAA Jounal 46 (7):1803-1813