Journal of Computational Physics 365 (2018) 135–148

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

An oscillation-free flow solver based on flux reconstruction

Horacio J. Aguerre [a,e,*], Cesar I. Pairetti [b,a], Cesar M. Venier [a,b], Santiago Márquez Damián [a,c], Norberto M. Nigro [a,d]

[a] Centro de Investigación de Métodos Computacionales, CONICET-UNL, Santa Fe, Argentina
[b] Escuela de Ingeniería Mecánica, Facultad de Ciencias Exactas, Ingenieria y Agrimensura, Universidad Nacional de Rosario, Rosario, Argentina
[c] Facultad Regional Santa Fe, Universidad Tecnológica Nacional, Santa Fe, Argentina
[d] Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral, Santa Fe, Argentina
[e] Facultad Regional Concepción del Uruguay, Universidad Tecnológica Nacional, Concepción del Uruguay, Argentina

ARTICLE INFO

Article history:
Received 12 January 2018
Accepted 21 March 2018
Available online 27 March 2018

Keywords:
Velocity oscillations
Flux reconstruction
Segregated method
Collocated grids
Finite Volume Method

ABSTRACT

In this paper, a segregated algorithm is proposed to suppress high-frequency oscillations in the velocity field for incompressible flows. In this context, a new velocity formula based on a reconstruction of face fluxes is defined eliminating high-frequency errors. In analogy to the Rhie–Chow interpolation, this approach is equivalent to including a flux-based pressure gradient with a velocity diffusion in the momentum equation. In order to guarantee second-order accuracy of the numerical solver, a set of conditions are defined for the reconstruction operator. To arrive at the final formulation, an outlook over the state of the art regarding velocity reconstruction procedures is presented comparing them through an error analysis. A new operator is then obtained by means of a flux difference minimization satisfying the required spatial accuracy. The accuracy of the new algorithm is analyzed by performing mesh convergence studies for unsteady Navier–Stokes problems with analytical solutions. The stabilization properties of the solver are then tested in a problem where spurious numerical oscillations arise for the velocity field. The results show a remarkable performance of the proposed technique eliminating high-frequency errors without losing accuracy.

1. Introduction

In the context of the Finite Volume Method (FVM), segregated algorithms are among the most popular strategies to couple pressure and velocity in Computational Fluid Dynamics (CFD). In these methodologies, the variables of the problem can be stored in the same or different spatial location, which gives place to grid arrangements known as collocated or staggered respectively. The staggered grids have the advantage of avoiding interpolations and thus prevent the appearance of spurious oscillations on the unknown fields. On the other hand, collocated grid arrangements are more suitable for unstructured meshes which are mandatory to address practical applications involving complex geometries.

One major disadvantage of collocated grids is the appearance of the aforementioned high-frequency oscillations due to a decoupling between pressure and velocity. This is related to the employment of central differencing for the discretization of the spatial derivatives. A common way to tackle these numerical issues is introducing a stabilizing effect on the pressure

[*] Corresponding author.
E-mail address: aguerrehoracio@gmail.com (H.J. Aguerre).

https://doi.org/10.1016/j.jcp.2018.03.033

equation through the so-called Rhie–Chow interpolation [1]. Since its development, this technique has been the most used alternative to mitigate pressure oscillations and it is still subject of several discussions regarding its accuracy and stability properties [2–8]. One of the main issues regarding this technique is that it does not guarantee non-oscillatory fields for all problem conditions. Oxtoby et al. [6] and Mencinger et al. [7] proposed new discretization rules to eliminate unphysical spikes at the interface with porous media in Volume Of Fluid techniques. Other authors proposed solutions involving adjustments on the Rhie–Chow interpolation to mitigate these problems. In particular, Zhang et al. [8] put in evidence the flaws of the standard Rhie–Chow interpolation when the corrective terms on the mass conservation equations become sufficiently large. Such situation arises when dealing with discontinuous momentum sources. They propose a correction consisting on smoothing out the pressure gradient on cell faces by introducing the body forces in the calculation. In the same manner, Nordlund et al. [9] addressed similar problems and proposed two methods to mitigate spurious oscillations: one involving an adjustment of the porous resistivity in the cells near the interface and another one which introduces auxiliary pressure values in the same region. All these works propose particular solutions to eliminate oscillatory fields in different situations. However, for the velocity, a deeper study on the reasons that generate high-frequency oscillations still needs to be addressed, which is the main motivation of this work.

In this context, a new approach to mitigate spurious high-frequency velocity oscillations in pressure-velocity coupling algorithms for incompressible flows is presented; this formulation is inspired by the non-oscillatory nature of the fluxes in a collocated FVM. More precisely, the velocity field is defined by a face-to-cell reconstruction of the fluxes which must satisfy certain conditions to preserve second-order accuracy. Reconstruction operators can be obtained through different procedures, such as flux difference minimization [10–12] or weighted average [13]. Here, a reconstruction operator is designed based on a flux difference minimization to obtain an oscillation-free algorithm.

The paper is organized as follows: On Section 2, the general framework of the standard SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) family algorithms is presented, including a study of the reasons for the velocity oscillations and a solution based on face-to-cell reconstruction. Section 3 presents an overlook of some reconstruction procedures followed by the development of a new reconstruction method. In Section 4, two benchmark cases are solved to demonstrate that the new formulation is second-order accurate. Section 5 shows the stabilization properties solving a porous media flow problem. Finally, the conclusions are presented in Section 6.

2. Theoretical background

2.1. Segregated algorithms on collocated grids

The continuum incompressible Navier–Stokes equations may be presented as:

$$\nabla\cdot \boldsymbol{u} = 0, \tag{2.1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla\cdot(\boldsymbol{u}\boldsymbol{u}) = -\nabla p + \nu\,\nabla^2 \boldsymbol{u} + \boldsymbol{\Phi}, \tag{2.2}$$

where $p = \dfrac{\mathcal{P}}{\rho}$, $\mathcal{P}$ is the pressure field, $\boldsymbol{u} = (u, v, w)$ is the velocity field, ρ is the mass density, ν is the kinematic viscosity and $\boldsymbol{\Phi}$ is a momentum source term (e.g. a body force). Based on the Finite Volume Method [14], Eqs. (2.1) and (2.2) may be discretized as:

$$\sum_f \boldsymbol{u}_f \cdot \mathbf{S}_f = \sum_f F_f = 0, \tag{2.3}$$

$$a_P\,\boldsymbol{u}_P + \sum_N a_N\,\boldsymbol{u}_N = b_P\,\boldsymbol{u}_P^0 + \boldsymbol{\Phi}_P - \nabla p_P, \tag{2.4}$$

where the subscript P indicates the cell-centered value of the current cell, N indicates neighbor cells values and f refers to the face values. The term $b_P\,\boldsymbol{u}_P^0$ is the contribution of the previous time-step and is function of the temporal scheme. The face-normal vector $\mathbf{S}_f$ has the magnitude of the face area and points out of the cell, F_f is the velocity flux at the face f and a_P and a_N are the diagonal and off-diagonal momentum matrix coefficients respectively. The operator ∇p_P is a Gauss-based gradient of the pressure computed using first neighbors values of that field. Isolating $\boldsymbol{u}_P$ from Eq. (2.4),

$$\boldsymbol{u}_P = \boldsymbol{H}_P - \frac{1}{a_P}\nabla p_P, \tag{2.5}$$

where the term $\boldsymbol{H}_P$ is computed as,

$$\boldsymbol{H}_P = \frac{b_P\,\boldsymbol{u}_P^0 + \boldsymbol{\Phi}_P - \sum_N a_N\,\boldsymbol{u}_N}{a_P}. \tag{2.6}$$

One way to solve Eqs. (2.3) and (2.4) for $\boldsymbol{u}$ and p consists on addressing one equation at a time, update one of the unknown fields, solve the remaining equation and iterate the sequence. These techniques are known as segregated methods, where the SIMPLE-family techniques [15–19] are among the most popular for incompressible flows. It broadly consists of a Momentum Predictor step and a Corrector step. In the Momentum Predictor, a first approximation for the velocity field is computed using the momentum equation where the pressure term is treated explicitly:

$$a_P\,\boldsymbol{u}_P^{i+1} + \sum_N a_N\,\boldsymbol{u}_N^{i+1} = b_P\,\boldsymbol{u}_P^0 + \boldsymbol{\Phi}_P - \nabla p_P^i, \tag{2.7}$$

where the superscript i indicates fields that are stored from previous iterations and $i+1$ are the fields being solved at the current iteration. This preliminary result does not satisfy the divergence-free condition. In the Corrector step, a pressure equation, based on including Eq. (2.5) in Eq. (2.3), is assembled and solved. In order to do so, the velocity field $\boldsymbol{u}_P$, originally on cell-centers, needs to be interpolated to the faces obtaining:

$$\sum_f \left(\boldsymbol{H}_P - \frac{1}{a_P}\nabla p_P\right)_f \cdot \mathbf{S}_f = 0 \tag{2.8}$$

Therefore, the Corrector step may be summarized as:

1. Assemble and solve the pressure based on Eq. (2.8):

$$\sum_f \left(\frac{1}{a_P}\nabla p_P\right)_f \cdot \mathbf{S}_f = \sum_f \boldsymbol{H}_f \cdot \mathbf{S}_f \tag{2.9}$$

   Here, $\boldsymbol{H}_f$ is obtained by linear interpolation of the $\boldsymbol{H}_P$ values on the cells that shares the same face f. In practice, this operation may lead to high-frequency oscillations for the pressure field. To address this issue, the left-hand-side term on Eq. (2.9) is replaced by $\sum_f \left(\frac{1}{a_P}\right)_f (\nabla p_f \cdot \mathbf{S}_f)$, where ∇p_f is a face gradient based on adjacent cell values [14]. This approximation is an undercover form of the Rhie–Chow correction [1] for collocated grids.
2. Compute the face fluxes by adding the new pressure contribution:

$$F_f = \boldsymbol{H}_f \cdot \mathbf{S}_f - \left(\frac{1}{a_P}\right)_f \nabla p_f \cdot \mathbf{S}_f \tag{2.10}$$

3. Correct the cell-centered velocity field based on the new pressure field, using Eq. (2.11)

$$\boldsymbol{u}_P = \boldsymbol{H}_P - \frac{1}{a_P}\nabla p_P \tag{2.11}$$

Most of the SIMPLE-family algorithms preserve this general structure. In particular, the SIMPLE algorithm [15], originally conceived as a steady state flow solver, adds under-relaxation factors to obtain a stable solution. Another variant is the PISO (Pressure-Implicit with Splitting of Operators) algorithm, which relies on iterating the Corrector step a given number of times to ensure a correct coupling between pressure and velocity for transient flow problems [19,20]. The SIMPLE-PISO combined technique has been largely adopted in several computational codes (e.g. OpenFOAM(R) [21]) to address transient incompressible flow problems due to its beneficial convergence features. It consists on iterating the corrector steps until a certain mass conservation criteria is fulfilled (as in PISO) and iterating the whole sequence (Momentum Predictor and Corrector Step) several times until both mass and momentum equations are verified for each time step. This algorithm will be adopted in the present work.

2.2. Velocity oscillations

The Rhie–Chow correction [1] is a mechanism to mitigate high-frequency oscillations in the pressure field. These oscillations are a consequence of using a central differencing scheme to discretize the pressure gradients. In this manner, high-frequency oscillations on the pressure are filtered in the momentum equation. The correction consists of modifying the pressure equation adding numerical diffusion which results in a smoother field with a non-oscillatory behavior.

For the velocity field, a similar issue arises due to the role of central interpolation schemes. Following Eq. (2.11), the velocity field is a function of $\boldsymbol{H}_P$, a_P and ∇p_P. This indicates that, in spite of any improvement regarding the pressure field oscillations, the corrected velocity field may still be affected by high-frequency oscillations on $\boldsymbol{H}_P$ and a_P. Moreover, the pressure gradient in Eq. (2.11) cannot control these issues since oscillations in $\boldsymbol{H}_P$ and a_P are not present in the pressure equation due to the cell-face interpolation procedure. Thus, there is no feedback possible between velocity oscillations and the pressure gradient. The interpolation operation is crucial since it is responsible for filtering high-frequency oscillations of cell-centered fields on faces. This may be appreciated by considering a generic field defined on cell centers, which can be decomposed in two terms as depicted in Fig. 1: one consisting only of a high-frequency contribution and other which is the oscillation-free field. After the linear interpolation on cell faces, the high-frequency contribution becomes null. Ultimately, the standard SIMPLE-PISO algorithm has no mechanism to damp velocity oscillations in the pressure equation as a consequence of linear interpolation of the velocity fluxes.

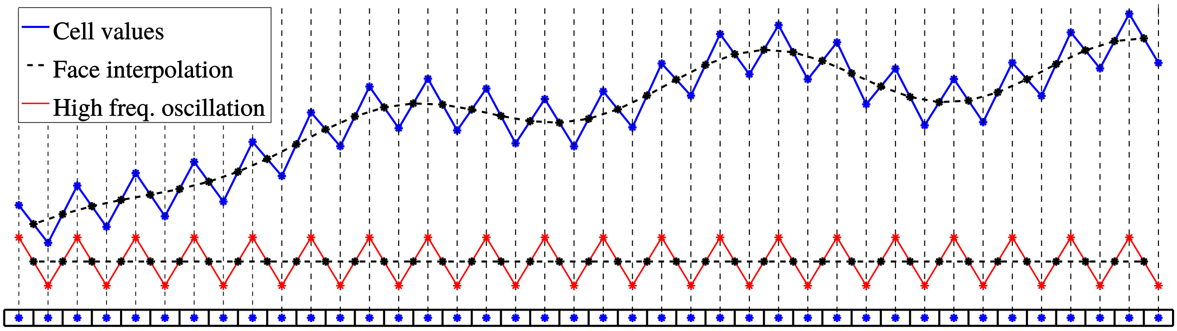


Fig. 1. Cell-centered oscillatory field and the filtering effect of face interpolation.

2.3. An oscillation-free corrector step for the velocity

After the pressure equation (Eq. (2.9)) is solved, a smooth pressure field is obtained and used to update the face fluxes F_f through Eq. (2.10). Therefore, the updated fluxes F_f are free from oscillations which encourages its use to eliminate such behavior on the velocity field. In this sense, this work proposes to compute the velocity at cell-centers through a face-to-cell reconstruction of the fluxes,

$$\tilde{\boldsymbol{u}}_P = \mathcal{R}(F_f), \tag{2.12}$$

where $\mathcal{R}$ is a linear operator that reconstructs face fluxes into a cell-centered velocity value with a given accuracy and preserves its smoothness. In order to be consistent with the momentum equation, the pressure gradient must be modified by including Eq. (2.12) in Eq. (2.11),

$$\tilde{\nabla} p_P = a_P \left[\boldsymbol{H}_P - \mathcal{R}(F_f) \right]. \tag{2.13}$$

In fact, the original algorithm can be preserved by only modifying the pressure gradient according to Eq. (2.13) for the algebraic momentum expression given by Eq. (2.11). Introducing Eq. (2.10) in Eq. (2.13) and reordering using the linear property of the operator $\mathcal{R}$,

$$\tilde{\nabla} p_P = a_P \mathcal{R} \left[\left(\frac{1}{a_P} \right)_f (\nabla p)_f \cdot \boldsymbol{S}_f \right] + a_P \left[\boldsymbol{H}_P - \mathcal{R}(\boldsymbol{H}_f \cdot \mathbf{S}_f) \right]. \tag{2.14}$$

It can be observed that this new pressure gradient has two contributions. The first term uses the pressure field obtained from Eq. (2.9) and the second one is a function of $\boldsymbol{H}_P$ and a_P. Then, the new Corrector step is,

$$\tilde{\boldsymbol{u}}_P = \boldsymbol{H}_P - \frac{1}{a_P} \tilde{\nabla} p_P. \tag{2.15}$$

2.4. Accuracy requirements for the new formulation

The redefinition of the pressure gradient given in Eq. (2.14) introduces a difference into the velocity corrector step with respect to the standard formulation. Subtracting Eq. (2.11) from Eq. (2.15),

$$\tilde{\boldsymbol{u}}_P - \boldsymbol{u}_P = -\frac{1}{a_P} \left[\left(\tilde{\nabla} p_P - \nabla p_P \right) \right], \tag{2.16}$$

and introducing Eq. (2.14) in the last expression,

$$\tilde{\boldsymbol{u}}_P - \boldsymbol{u}_P = \left[\mathcal{R}(\boldsymbol{H}_f \cdot \mathbf{S}_f) - \boldsymbol{H}_P \right] - \mathcal{R} \left[\left(\frac{1}{a_P} \right)_f (\nabla p)_f \cdot \boldsymbol{S}_f \right] + \frac{\nabla p_P}{a_P}. \tag{2.17}$$

This expression is the difference between the new and standard methodologies for the first time step where $\boldsymbol{H}_P$, a_P and the pressure field p are the same in both formulations. The difference at a given time, $\tilde{\boldsymbol{u}}_P(t) - \boldsymbol{u}_P(t)$, is the accumulation of the errors generated at each time-step:

$$\tilde{\boldsymbol{u}}_P(t) - \boldsymbol{u}_P(t) = \sum_{n=1}^{N} \left(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n \right) \quad (N = t/\Delta t), \tag{2.18}$$

where n is the time-step index.

The quantity defined in the last equation is an extra term included in the governing equations to suppress velocity oscillations. To obtain second-order convergence of the numerical method, the spatial and temporal terms must be discretized with second-order schemes as a necessary condition. However, when an extra term is incorporated in the governing equations, this condition may not be sufficient since these terms may alter the convergence behavior. Any additional term incorporated into the equations must converge to zero when the mesh and time steps tend to zero. For the term of Eq. (2.18) this means that:

$$\lim_{\Delta x,\, \Delta t \to 0} \left[\tilde{\boldsymbol{u}}_P(t) - \boldsymbol{u}_P(t) \right] = 0. \tag{2.19}$$

Otherwise, the numerical formulation will not be consistent having a zero-order term. Moreover, the magnitude of this term must decrease to zero with a second-order rate to guarantee this order of convergence for the solver. In this context, the following condition is imposed for Eq. (2.18), considering a mesh refinement analysis with a constant Courant number and a uniform time-step,

$$\mathcal{O}(\tilde{\boldsymbol{u}}_P(t) - \boldsymbol{u}_P(t)) = \mathcal{O}\left[\sum_{n=1}^{N} \left(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n \right) \right] = \mathcal{O}(h^k) \qquad k \geq 2, \quad (N = t/\Delta t), \tag{2.20}$$

where h is the cell size. Since each of the terms of the summation have the same order,

$$\mathcal{O}\left[\frac{t}{\Delta t} \left(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n \right) \right] = \mathcal{O}\left[\frac{t}{h} \left(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n \right) \right] = \mathcal{O}(h^k) \qquad k \geq 2, \tag{2.21}$$

and then,

$$(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n) \propto \mathcal{O}(h^k) \qquad k \geq 3. \tag{2.22}$$

Considering that the term $\left(\tilde{\boldsymbol{u}}_P^n - \boldsymbol{u}_P^n \right)$ shares the same order of the term given by Eq. (2.17), the condition over the reconstruction operator can be expressed as,

$$\mathcal{R}\left[\left(\frac{1}{a_P} \right)_f (\nabla p)_f \cdot \boldsymbol{S}_f \right] - \frac{\nabla p_P}{a_P} + \left[\boldsymbol{H}_P - \mathcal{R}(\boldsymbol{H}_f \cdot \boldsymbol{S}_f) \right] \propto \mathcal{O}(h^k) \qquad k \geq 3, \tag{2.23}$$

which implies,

$$\left[\boldsymbol{H}_P - \mathcal{R}(\boldsymbol{H}_f \cdot \boldsymbol{S}_f) \right] \propto \mathcal{O}(h^k) \qquad k \geq 3, \tag{2.24}$$

$$\mathcal{R}\left[\left(\frac{1}{a_P} \right)_f (\nabla p)_f \cdot \boldsymbol{S}_f \right] - \frac{\nabla p_P}{a_P} \propto \mathcal{O}(h^k) \qquad k \geq 3. \tag{2.25}$$

In Eq. (2.24), a third-order convergence is imposed over the $\mathcal{R}$ operator. An analysis over the second requirement shows that the term a_P varies with the spatial and temporal refinement. Given a standard discretization scheme for the momentum equation, a_P can be expressed as:

$$a_P = \left(\frac{U}{\mathrm{Co}\, h} + \sum_f \frac{\alpha_f}{h} + \sum_f \frac{\nu \beta_f}{h^2} \right), \tag{2.26}$$

where α_f and β_f are the convective and diffusive coefficients given by the discretization schemes and U is the characteristic velocity of the problem. Considering that α_f, β_f, ν, U and the Courant number $(Co = U\Delta t/h)$ are constant, the most restrictive condition arises when $\nu \to 0$, which implies,

$$\lim_{h \to 0} a_P \propto \frac{1}{h}. \tag{2.27}$$

This expression corresponds to a convective-dominated flow. Introducing Eq. (2.27) in Eq. (2.25), the following requirement is obtained for the reconstruct operator $\mathcal{R}$,

$$\nabla p_P - \mathcal{R}\left(\nabla p_f \cdot \boldsymbol{S}_f \right) \propto \mathcal{O}(h^k) \qquad k \geq 2. \tag{2.28}$$

Adopting an $\mathcal{R}$ operator that satisfies the conditions given in Eq. (2.24) and Eq. (2.28) for the computation of the pressure gradient (Eq. 2.14), an oscillation-free and second-order accurate solution will be obtained.

3. Face-to-cell reconstruction of vector fields

In the context of the FVM on collocated grids, computing face values based on cell interpolations is a common practice. The reverse operation, getting cell values from face fields, is known as reconstruction [10,13]. This procedure can be performed in several ways.

Shashkov et al. [10] propose a least square scheme to minimize the difference between the original fluxes at faces $F_f(\boldsymbol{u}_f)$ and a new set of fluxes F_f^* which are expressed as function of $\boldsymbol{u}_P$. In this sense, the problem consists in finding the value of $\boldsymbol{u}_P$ which minimize the following function:

$$g(\boldsymbol{u}_P) = \sum_f \left[F_f(\boldsymbol{u}_f) - F_f^*(\boldsymbol{u}_P) \right]^2, \tag{3.1}$$

where $\boldsymbol{u}_f$ are the faces values of the velocity field to be reconstructed and $F_f = \boldsymbol{u}_f \cdot \boldsymbol{S}_f$ are the corresponding face fluxes. Shashkov et al. describe two approximations for the fluxes F_f^* using a zero and first-order Taylor series,

$$\begin{aligned} F_f^*(\boldsymbol{u}_P) &= \boldsymbol{u}_P \cdot \boldsymbol{S}_f \\ F_f^*(\boldsymbol{u}_P) &= (\boldsymbol{u}_P + \nabla \boldsymbol{u}_P \cdot \boldsymbol{d}_{Pf}) \cdot \boldsymbol{S}_f, \end{aligned} \tag{3.2}$$

where $\boldsymbol{d}_{Pf}$ is the vector connecting the cell and face centroids (pointing to the face centroid) and $\nabla \boldsymbol{u}_P$ is the cell-centered Gauss gradient. Solving the minimization problem in Eq. (3.1), the following cell-centered velocity expressions are obtained,

$$\begin{aligned} \boldsymbol{u}_P &= \left(\sum_f \boldsymbol{S}_f \boldsymbol{S}_f \right)^{-1} \cdot \left(\sum_f F_f \boldsymbol{S}_f \right), \\ \boldsymbol{u}_P &= \left(\sum_f \boldsymbol{S}_f \boldsymbol{S}_f \right)^{-1} \cdot \left[\sum_f \left(F_f - \nabla \boldsymbol{u}_P \cdot \boldsymbol{d}_{Pf} \cdot \boldsymbol{S}_f \right) \boldsymbol{S}_f \right]. \end{aligned} \tag{3.3}$$

On the other hand, Weller et al. [11,12] describes a cell-centered velocity reconstruction,

$$\boldsymbol{u}_P = \left(\sum_f \frac{\boldsymbol{S}_f \boldsymbol{S}_f}{\|\boldsymbol{S}_f\|} \right)^{-1} \cdot \left[\sum_f F_f \frac{\boldsymbol{S}_f}{\|\boldsymbol{S}_f\|} \right]. \tag{3.4}$$

This form is adopted in some CFD codes (*e.g.* OpenFOAM(R) [21]). It can be demonstrated that this formula results from solving a minimization problem using a functional similar to that presented by Shashkov et al., but employing the inverse of the face area as a weighting factor for the face error,

$$g(\boldsymbol{u}_P) = \sum_f \frac{1}{\|\boldsymbol{S}_f\|} \left[F_f(\boldsymbol{u}_f) - F_f^*(\boldsymbol{u}_P) \right]^2. \tag{3.5}$$

In this case, the zero order approximation of Eq. (3.2) for the flux $F_f^*(\boldsymbol{u}_P)$ is considered.

A different approach, by Perot et al. [13], consists in computing a volume weighted average of the given face values $\boldsymbol{u}_f$,

$$\boldsymbol{u}_P = \frac{1}{V_P} \sum_f (\boldsymbol{S}_f \cdot \boldsymbol{d}_{Pf}) \boldsymbol{u}_f. \tag{3.6}$$

3.1. A new reconstruction formula

Linear interpolation of cell values to faces consists in applying a second order Taylor approximation based on a face gradient $\nabla \boldsymbol{u}_f$ computed from adjacent cells,

$$\boldsymbol{u}_f^* = \boldsymbol{u}_P + \nabla \boldsymbol{u}_f \cdot \boldsymbol{d}_{Pf}. \tag{3.7}$$

The reconstruction operator should act as the inverse of cell-to-face interpolation, as suggested by Eq. (2.24). To accomplish this, the approximation of fluxes in the reconstruction procedures should be based on Eq. (3.7) which is not the case of the presented reconstructions.

If the original velocity field is oscillatory, adopting the same face gradient used in Eq. (3.7) for the reconstruction will give the exact inverse function of the linear interpolation recovering the oscillatory behavior. For this reason, the face gradient on Eq. (3.7) is replaced here by an approximation based on a linear interpolation of cell gradients, $(\overline{\nabla \boldsymbol{u}_P^i})_f$. Thus, the following expression for the approximated face values $\boldsymbol{u}_f^*$ is proposed,

$$\boldsymbol{u}_f^* = \boldsymbol{u}_P^{i+1} + \overline{(\nabla \boldsymbol{u}_P^i)}_f \cdot \boldsymbol{d}_{Pf}. \tag{3.8}$$

The fluxes resulting from this approximation take the following form,

$$F_f^*(\boldsymbol{u}_P) = \left[\boldsymbol{u}_P^{i+1} + \overline{(\nabla \boldsymbol{u}_P^i)}_f \cdot \boldsymbol{d}_{Pf}\right] \cdot \boldsymbol{S}_f. \tag{3.9}$$

Adopting the minimization of the function given by Eq. (3.5), the $\boldsymbol{u}_P^{i+1}$ is,

$$\boldsymbol{u}_P^{i+1} = \mathcal{R}(F_f) = \left(\sum_f \frac{\boldsymbol{S}_f \boldsymbol{S}_f}{\|\boldsymbol{S}_f\|}\right)^{-1} \cdot \left\{\sum_f \left[F_f - \overline{(\nabla \boldsymbol{u}_P^i)}_f \cdot \boldsymbol{d}_{Pf} \cdot \boldsymbol{S}_f\right] \frac{\boldsymbol{S}_f}{\|\boldsymbol{S}_f\|}\right\}. \tag{3.10}$$

where $\overline{(\nabla \boldsymbol{u}_P^i)}_f$ is computed based on stored velocity values from the previous iteration of the Corrector Step.

The difference between the velocity given by Eq. (3.7) and the spatial approximation of the new reconstruction formula given by Eq. (3.8) is,

$$\boldsymbol{u}_f^* - \boldsymbol{u}_f = \left[\overline{(\nabla \boldsymbol{u}_P^i)}_f - \nabla \boldsymbol{u}_f\right] \cdot \boldsymbol{d}_{pf}. \tag{3.11}$$

The numerical error introduced to the final reconstructed velocity field is proportional to Eq. (3.11), which is equivalent to a discrete form of a third order derivative [22]. The Rhie–Chow interpolation adds a similar term based on pressure gradients in the pressure equation. In the same manner, the proposed operator adds a diffusive effect over the velocity filtering high-frequency oscillations in the final reconstructed fields.

3.2. Accuracy of the reconstruction operators

To evaluate the accuracy of the reconstruction methodologies, the following test is performed: a vector field located at cells $\boldsymbol{u}_P$ is linearly interpolated at faces obtaining the fluxes F_f. Subsequently, these fluxes are reconstructed into cell values to define the field $\tilde{\boldsymbol{u}}_P$. The error of this procedure is computed as the root mean square (RMS) difference between the initial and final fields $\boldsymbol{u}_P$ and $\tilde{\boldsymbol{u}}_P$ given by,

$$E_2(\boldsymbol{u}_P) = \sqrt{\frac{\sum_i^n \left(\boldsymbol{u}_P - \tilde{\boldsymbol{u}}_P\right)^2}{n}}, \tag{3.12}$$

where n represents the total number of cells. The field used in the test is the initial condition of the two-dimensional Taylor–Green problem:

$$\begin{aligned} u_x(x,y) &= -\sin(x)\cos(y) \\ u_y(x,y) &= \cos(x)\sin(y). \end{aligned} \tag{3.13}$$

The procedure is applied over a square domain with a side length of 2π. To analyze the convergence of the error, the mesh resolution is increased by a factor of two ranging from 10 to 640 grid points per side. The results are presented in Fig. 2 showing the error as a function of the mesh resolution. The methods proposed by Shashkov et al., Weller et al. and Perot et al. follow a second order trend which, according to the conditions given in Eq. (2.24), will reduce the accuracy of the flow solver below second order. In contrast, the reconstruction formula defined by Eq. (3.10) follows a fourth order trend, fulfilling the first requirement.

A second test is conceived to prove if the new reconstruction formula satisfy the second requirement given by Eq. (2.28). Here, the RMS error is computed as,

$$E_2(\nabla\phi_P) = \sqrt{\frac{\sum_1^n (\nabla\phi_P - \nabla\phi_P^{\mathcal{R}})^2}{n}}, \tag{3.14}$$

where ϕ is a generic field and $\nabla\phi_P$ is the cell-centered Gauss gradient and $\nabla\phi_P^{\mathcal{R}}$ is calculated by the following formula:

$$\nabla\phi_P^{\mathcal{R}} = \mathcal{R}(\nabla\phi_f \cdot \boldsymbol{S}_f). \tag{3.15}$$

The field for this test is a Gaussian function,

$$\phi(x,y) = \exp\left[\frac{(x^2+y^2)}{2}\right]. \tag{3.16}$$

The domain is a square with a side length of 20 m. As in the first test, the computation of E_2 uses different mesh sizes varying between 20 and 2560 divisions per side. The results are presented in Fig. 3 where the required second-order convergence for the term given by Eq. (2.28) is obtained.

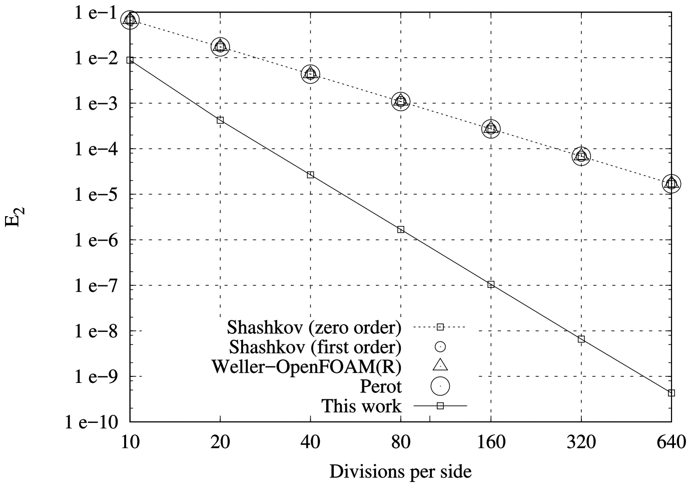


Fig. 2. Error between final and initial fields applying several reconstruction operators.

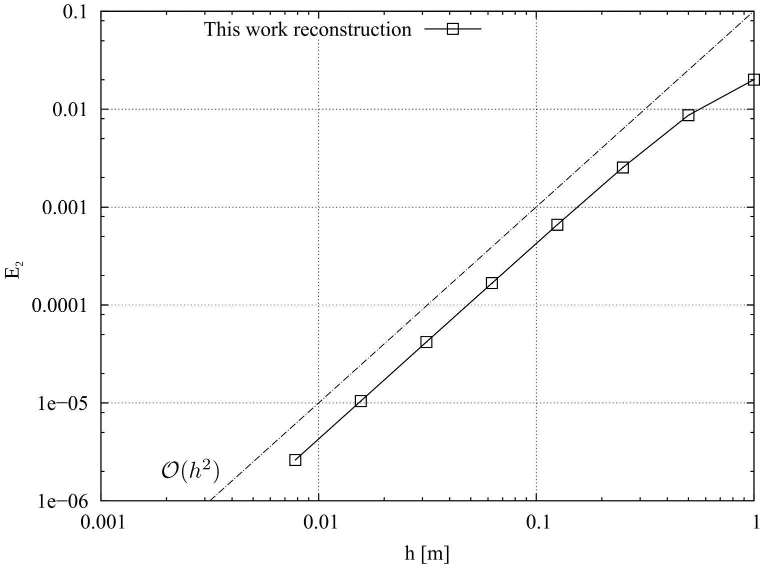


Fig. 3. E_2 convergence of error using the new reconstruction formula to compute cell centered gradient.

The new reconstruction formula satisfy the requirements given in Eqs. (2.24) and (2.28) which enables its use in Eq. (2.14) ensuring a second-order accuracy of the flow solver when using second-order discretization schemes for the spatial and temporal terms.

4. Accuracy of the flow solver

This section is devoted to evaluate the accuracy of the flow solver which uses the new reconstruction technique (Eq. (3.10)) for the velocity computation. In this sense, two unsteady flow problems are solved. To compare, two additional numerical formulations are included in the analysis: the standard SIMPLE-PISO algorithm and a solver which computes the velocity via Eq. (2.12) but using the reconstruction formula described by Weller et al. (Eq. (3.4)) which is the standard face-cell reconstruction operator implemented in the OpenFOAM(R) suite. In all simulations, second-order discretization schemes are set up for the temporal and spatial terms. For each problem, the numerical results are compared with its respective analytical solution and an error convergence study is performed varying the mesh resolution and the time-step size.

4.1. Taylor–Green vortex problem

The two-dimensional Taylor–Green case is an unsteady flow problem in which initial vortices decay due to viscosity dissipation. The analytic solution of the problem is presented as follows:

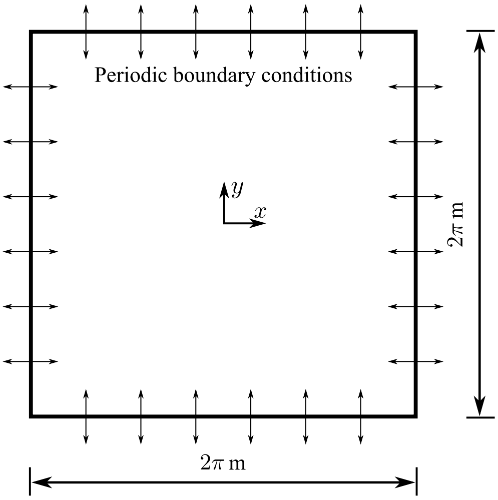


Fig. 4. Square domain with periodic boundary conditions.

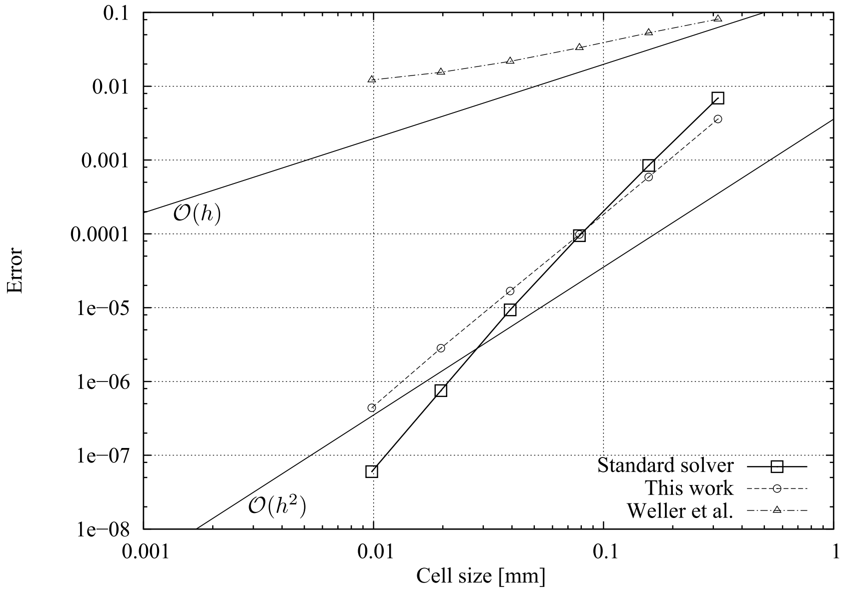


Fig. 5. Error in velocity for the Taylor–Green vortex problem.

$$\begin{aligned} u\,(x,y,t) &= -\sin(x)\cos(y)\exp(-2\nu t), \\ v\,(x,y,t) &= \cos(x)\sin(y)\exp(-2\nu t), \\ p(x,y,t) &= \frac{1}{4}\left[\cos(2x)+\sin(2y)\right]\exp(-4\nu t), \end{aligned} \tag{4.1}$$

where ν is the kinematic viscosity. The domain of the problem is a square with sides of 2π where periodic conditions are set up at the boundaries as depicted in Fig. 4.

A convergence study is performed evaluating the numerical error of the velocity for several mesh resolutions, where the coarsest grid has 20 divisions per side and the finest one is refined with 640 divisions per side. In all simulations, the time step is adjusted accordingly to define a maximal Courant number of 2 and the kinematic viscosity is 0.01 m^2/s corresponding to a Reynolds number of 157. The simulations end at $t = 1.28$ s.

The error curves of the velocity components are plotted in Fig. 5. The solvers with the standard formulation and with the new reconstruction formula exhibit a convergence rate higher than second order. On the contrary, the solver that uses the reconstruction of Weller et al. is below first-order accurate. The numerical dissipation of this last technique can be seen in Fig. 6 where the velocity magnitude along the domain is depicted for the simulations with the coarsest mesh resolution. In the same figure, no significant difference between the results of the standard solver and the current technique is recognized.

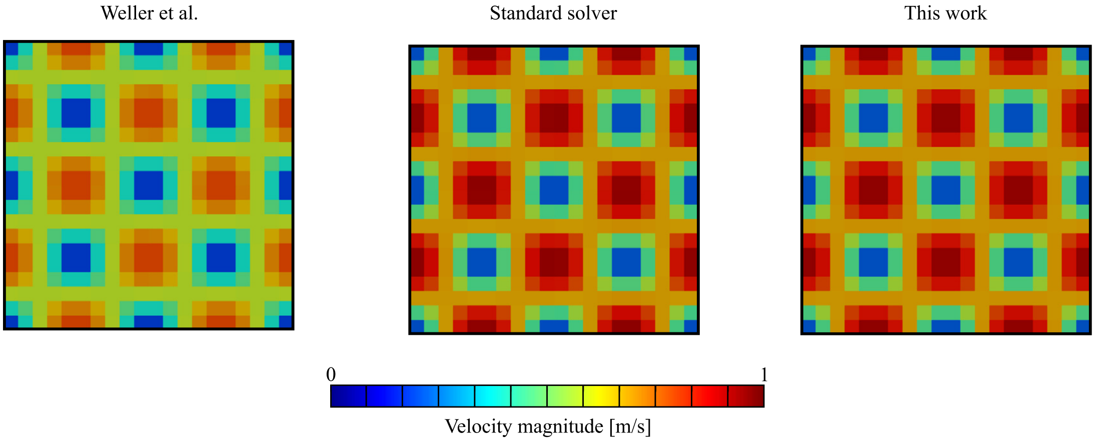


Fig. 6. Taylor–Green vortex problem: domains colored according to velocity magnitude at time 1.28 s. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

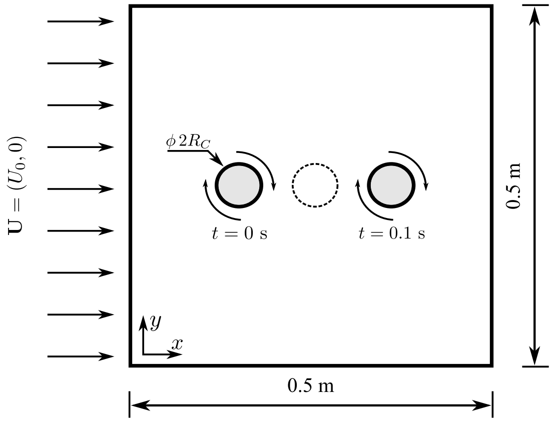


Fig. 7. Description of the vortex advection problem.

4.2. Viscous advection of a vortex

This test is graphically explained in Fig. 7: a vortex structure of diameter $2\,R_c$ which is initially located at position $(-3R_c, 0)$ is transported through an uniform flow $\mathbf{U} = (U_0,\, 0)$ during a total time of $t = 0.1$ s. The problem is solved using a square domain with sides of 0.5 m.

The initial conditions for the velocity field are,

$$u(x, y, 0) = U_0 - \frac{\Gamma}{R_c^2}\,(y - y_c)\exp\left(\frac{-r^2}{2R_c^2}\right), \qquad v(x, y, 0) = \frac{\Gamma}{R_c^2}\,(x - x_c)\exp\left(\frac{-r^2}{2R_c^2}\right), \tag{4.2}$$

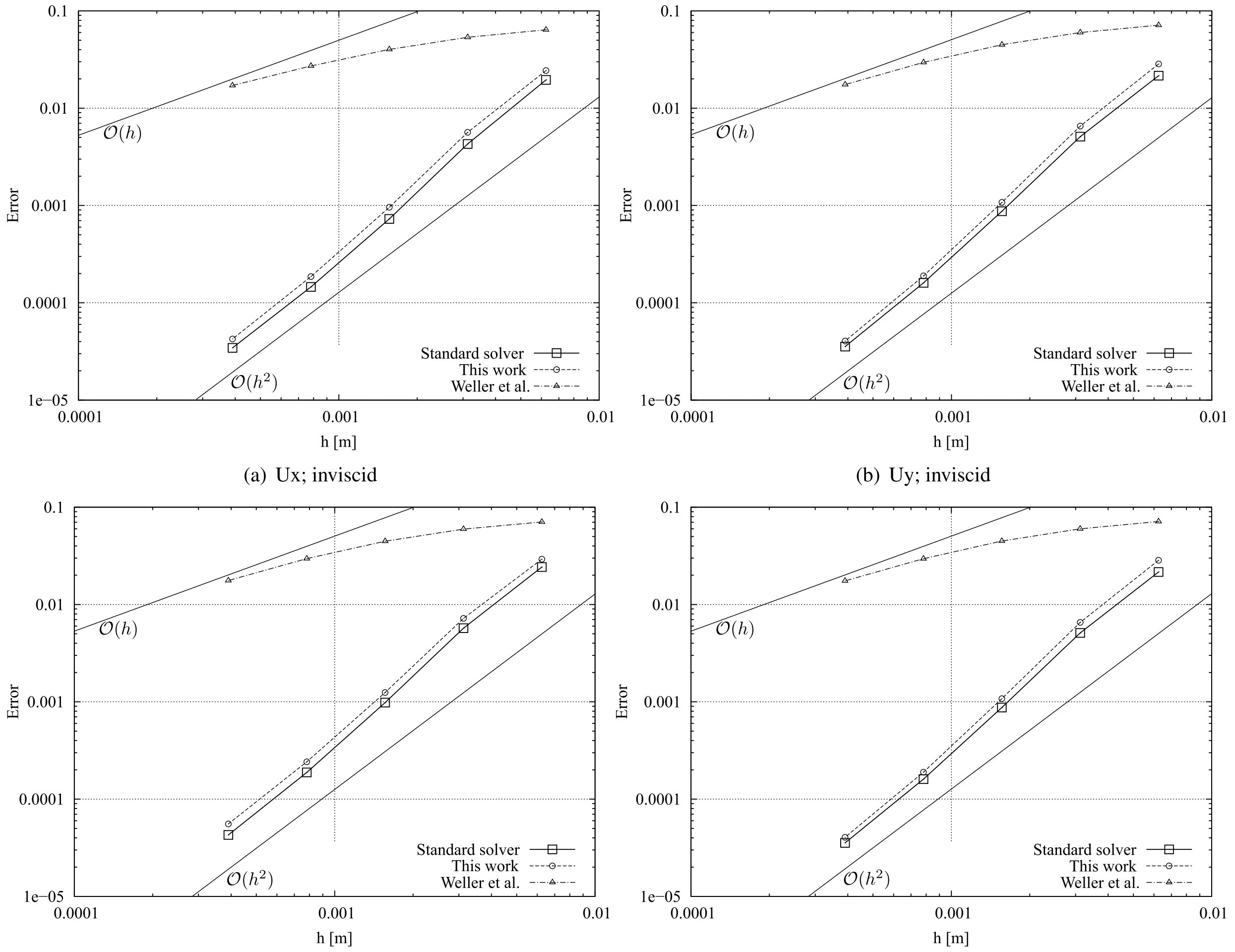
where x_c and y_c are the vortex center coordinates, $r = \sqrt[2]{(x - x_c)^2 + (y - y_c)^2}$ is the distance to the vortex center, $\Gamma = 0.036$ is the vortex strength, R_c the radius of the vortex and U_0 is the mean velocity magnitude. The analytical solution for this problem is,

$$u(x, y, t) = U_0 - \frac{\Gamma}{R_c^2}\,\frac{(y - y_c)}{\alpha^2}\exp\left(\frac{-r^2}{2\,\alpha\, R_c^2}\right), \qquad v(x, y, t) = \frac{\Gamma}{R_c^2}\,\frac{(x - x_c)}{\alpha^2}\exp\left(\frac{-r^2}{2\,\alpha\, R_c^2}\right), \tag{4.3}$$

where α is,

$$\alpha = 1 + 2\,\nu\, t, \tag{4.4}$$

and ν is the kinematic viscosity. A convergence study is performed using several mesh resolutions and modifying the time-step in order to preserve a constant Courant number of 0.9. The convergence of the error is presented in Fig. 8 which shows a second-order convergence of the solver with the new reconstruction approach. For this solver, there is a relatively small deviation of the error against the standard SIMPLE-PISO formulation proving that the accuracy is not affected by the



(a) Ux; inviscid

(b) Uy; inviscid

(c) Ux; Re = 155.6

(d) Uy; Re = 155.6

Fig. 8. Error of velocity components for two different viscosities using three formulations: standard solver, this work reconstruction and OpenFOAM(R) reconstruction.

new reconstruction step. On the other hand, the solver based on the reconstruction described by Weller et al. reduces the solution accuracy to first order. In Fig. 9, results showing the velocity magnitude are shown for the inviscid simulations with a mesh resolution of 80 × 80.

The vortex structure is similar in the standard and new solver solutions. On the other hand, the reconstruction formula of Weller et al. produces high levels of dissipation.

5. Non-oscillatory property of the new solver: discontinuous body-force stabilization

In this section, the stabilization effect of the new solver is studied comparing its numerical results with those of the standard SIMPLE-PISO formulation. With this aim, an incompressible flow through a two-dimensional channel with a porous material in its middle region is simulated. A schematic description for this problem is presented in Fig. 10.

The porous plug is modeled by the Darcy–Forchheimer model, as described in [23], resulting in the following momentum balance:

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{1}{\varepsilon}\nabla\cdot(\boldsymbol{u}\boldsymbol{u}) = -\varepsilon\nabla p + \nu\,\nabla^2\boldsymbol{u} - \left(\frac{\varepsilon\nu}{\kappa} + \frac{\varepsilon}{\sqrt{\kappa}}\frac{c_E\,|\boldsymbol{u}|}{2}\right)\boldsymbol{u}, \tag{5.1}$$

where ε is the porosity of the material, κ is its permeability (given by $\mathrm{Da} = \kappa/H^2$) and c_E is the inertial coefficient, which is computed using the expression $c_E = 1.75\,\varepsilon/\sqrt{150\,\varepsilon^5}$. The boundary conditions of the problem are the following: the inlet of the channel has a fixed inlet velocity profile $\boldsymbol{u}_{\mathrm{in}} = 6\,(y/H)\,[1-(y/H)]$, the outlet has a fixed pressure $p_{\mathrm{out}} = 0$ and a non-slip condition is imposed at the top and bottom walls.

The problem is addressed using different Reynolds numbers (Re = 25, 250 and 2500) and evaluating three Darcy numbers ($\mathrm{Da} = 10^{-3}$, 10^{-4} and 10^{-5}). Regarding the porosity properties, $\varepsilon = 1$ in the non-porous region Ω_A and at $\varepsilon = 0.7$ in the

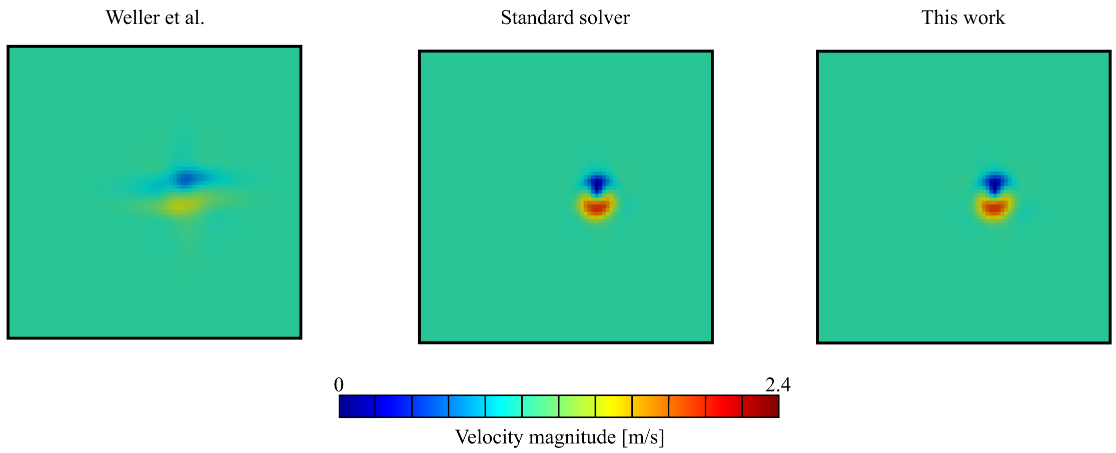


Fig. 9. Vortex advection problem: domains colored according to velocity magnitude at time 0.1 s.

porous region Ω_B. All simulations are solved using a structured mesh of $h = 2$ mm and a time-step of $\Delta t = 0.025$ s resulting in a mean Courant number of 1. As in the previous tests, second order temporal and spatial discretization schemes are applied.

The model of Eq. (5.1) introduces a discontinuous body force acting only in the porous region. This discontinuity is prone to generate velocity oscillations for the standard SIMPLE-PISO algorithm when the body force is large enough. This behavior can be recognized in the steady-state solution of the problem as shown in Fig. 11, where both standard and new approaches are presented. It can be observed that the standard algorithm solution presents unphysical velocity peaks at the interface between regions. In contrast, the new formulation produces a smooth solution.

A more detailed insight into the numerical results is presented in Fig. 12 which shows the velocity along the x axis for several Re and Da values. For the standard formulation, the results show that the amplitude of the velocity oscillation increases as the Da and Re numbers decrease (producing an increment on the Darcy force term). However, this numerical anomaly is completely eliminated with the new formulation which exhibits a physical-compatible variation of the velocity profiles for all cases.

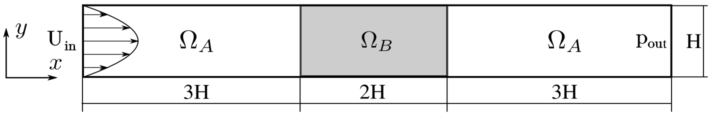


Fig. 10. Domain for the porous plug test. $H = 50$ mm.

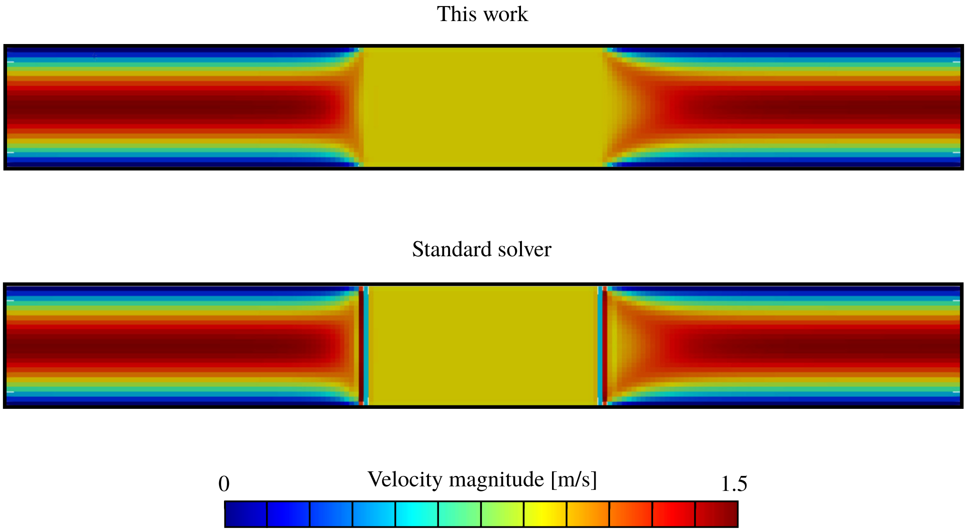


Fig. 11. Velocity field solved with (top) and without (bottom) applying velocity reconstruction ($Da = 10^{-5}$, $Re = 25$).

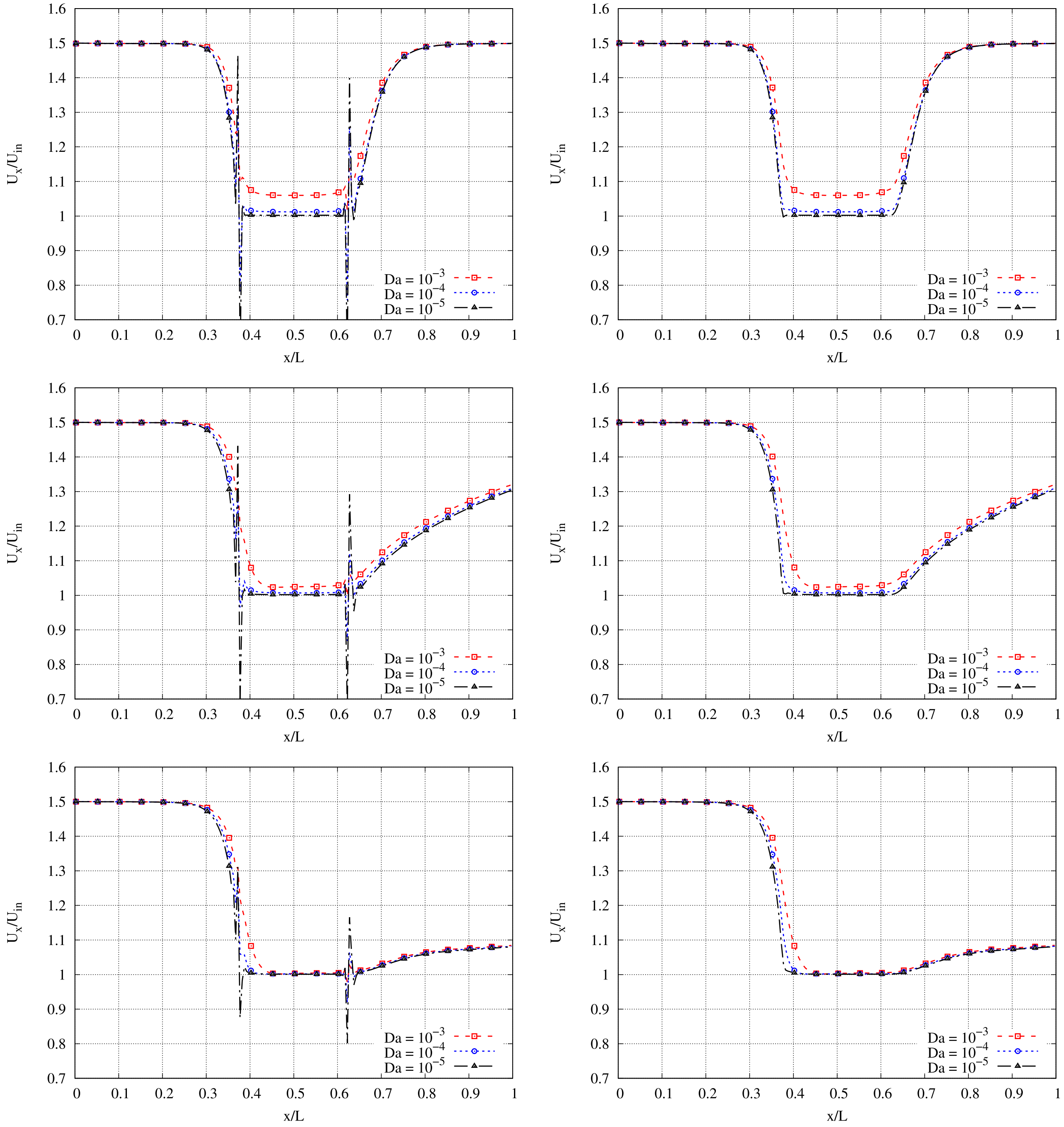


Fig. 12. Porous plug test solved by the standard (left) and new (right) solver. Re numbers are, from top to bottom, 25, 250 and 2500.

6. Conclusions

A new incompressible flow solver that eliminates spurious velocity oscillations has been developed. The procedure relies on a reconstruction of face fluxes to compute the velocity values. To guarantee second-order accuracy of the numerical formulation, two requirements have been obtained defining lower bounds for the spatial accuracy of the reconstruction operator. In order to fulfill these conditions, a new high-order formula has been designed.

An analysis of the SIMPLE-PISO algorithm has shown that oscillations on $\boldsymbol{H}_P$ and a_P remain after a whole iteration and are introduced directly into the velocity correction. On the other hand, the fluxes corrected by the pressure equation are free from this spurious behavior due to the filtering effect of the linear interpolation. Hence, reconstructing these fluxes results in an oscillation-free cell-centered velocity. The new approach is equivalent to using a different pressure gradient in the momentum equation. This introduces numerical diffusion proportional to a high-order derivative of $\boldsymbol{H}_P$ damping velocity oscillations in a similar manner as the Rhie–Chow correction over the pressure.

The proposed solver has been tested for two benchmark cases observing that the second-order accuracy given by the discretization schemes is preserved throughout the iterative procedure in the same manner as the standard formulation. After that, the non-oscillatory feature of this new technique is put into contrast with the original algorithm in a practical problem with a porous material, which is modeled as a discontinuous body force. The results show that the new methodology completely removes the oscillations of the velocity near the interface for different problem conditions.

Finally, it is worth to emphasize that this new formulation does not affect the global structure of the algorithm which makes it suitable for a direct implementation in pre-developed computational codes.

Acknowledgements

The authors wish to thank CONICET, Universidad Nacional del Litoral, ANPCyT and Universidad Tecnologica Nacional for their financial support through grants PIP-2012 GI 11220110100331, CAI+D 2011 501 201101 00435 LI, PICT-2013 0830 and PID-UTN 4364.

References

[1] C. Rhie, W. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA J. 21 (1983) 1525–1532.
[2] S.K. Choi, Note on the use of momentum interpolation method for unsteady flows, Numer. Heat Transf., Part A, Appl. 36 (1999) 545–550.
[3] Y. Kawaguchi, W.-Q. Tao, H. Ozoe, Checkerboard pressure predictions due to the underrelaxation factor and time step size for a nonstaggered grid with momentum interpolation method, Numer. Heat Transf., Part B, Fundam. 41 (2002) 85–94.
[4] S. Majumdar, Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids, Numer. Heat Transf. 13 (1988) 125–132.
[5] T. Miller, F. Schmidt, Use of a pressure-weighted interpolation method for the solution of the incompressible Navier–Stokes equations on a nonstaggered grid system, Numer. Heat Transf., Part A, Appl. 14 (1988) 213–233.
[6] O. Oxtoby, J. Heyns, R. Suliman, A Finite-Volume Solver for Two-Fluid Flow in Heterogeneous Porous Media Based on OpenFOAM R G, 2013.
[7] J. Mencinger, I. Žun, On the finite volume discretization of discontinuous body force field on collocated grid: application to VOF method, J. Comput. Phys. 221 (2007) 524–538.
[8] S. Zhang, X. Zhao, S. Bayyuk, Generalized formulations for the Rhie–Chow interpolation, J. Comput. Phys. 258 (2014) 880–914.
[9] M. Nordlund, M. Stanic, A. Kuczaj, E. Frederix, B. Geurts, Improved PISO algorithms for modeling density varying flow in conjugate fluid–porous domains, J. Comput. Phys. 306 (2016) 199–215.
[10] M. Shashkov, B. Swartz, B. Wendroff, Local reconstruction of a vector field from its normal components on the faces of grid cells, J. Comput. Phys. 139 (1998) 406–409.
[11] H. Weller, Non-orthogonal version of the arbitrary polygonal C-grid and a new diamond grid, Geosci. Model Dev. 7 (2014) 779–797.
[12] H. Weller, A. Shahrokhi, Curl-free pressure gradients over orography in a solution of the fully compressible Euler equations with implicit treatment of acoustic and gravity waves, Mon. Weather Rev. 142 (2014) 4439–4457.
[13] J. Perot, D. Vidovic, P. Wesseling, Mimetic reconstruction of vectors, Compat. Spat. Discretizations (2006) 173–188.
[14] H. Jasak, Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows, Ph.D. thesis, Imperial College of Science, Technology and Medicine, 1996.
[15] S.V. Patankar, D.B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transf. 15 (1972) 1787–1806.
[16] S. Patankar, Numerical Heat Transfer and Fluid Flow, CRC Press, 1980.
[17] S.V. Patankar, A calculation procedure for two-dimensional elliptic situations, Numer. Heat Transf. 4 (1981) 409–425.
[18] J. Van Doormaal, G. Raithby, Enhancements of the SIMPLE method for predicting incompressible fluid flows, Numer. Heat Transf. 7 (1984) 147–163.
[19] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, J. Comput. Phys. 62 (1986) 40–65.
[20] R.I. Issa, A. Gosman, A. Watkins, The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme, J. Comput. Phys. 62 (1986) 66–82.
[21] H.G. Weller, G. Tabor, H. Jasak, C. Fureby, A tensorial approach to computational continuum mechanics using object-oriented techniques, Comput. Phys. 12 (1998) 620–631.
[22] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, Springer Science & Business Media, 2012.
[23] L. Betchen, A.G. Straatman, B.E. Thompson, A nonequilibrium finite-volume model for conjugate fluid/porous/solid domains, Numer. Heat Transf., Part A, Appl. 49 (2006) 543–565.