Approximation of the convective flux in the incompressible Navier-Stokes equations using local boundary-value problems

Citation for published version (APA):

Document license:
TAVERNE

DOI:
10.1016/j.cam.2018.01.030

Document status and date:
Published: 01/10/2018

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Approximation of the convective flux in the incompressible Navier–Stokes equations using local boundary-value problems

N. Kumar, J.H.M. ten Thije Boonkkamp, B. Koren *

Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

1. Introduction

In this paper, we present the approximation of the convective flux in the incompressible Navier–Stokes equations using local boundary-value problems. For the numerical solution of the incompressible Navier–Stokes equations, we use the finite-volume method over staggered grids for the spatial discretization. The resulting semi-discrete system is then integrated in time using Runge–Kutta methods, as described in [1,2]. The semi-discrete system of equations requires velocity components to be approximated at the interfaces of the control volumes. Commonly used methods for the approximation of these cell-face velocity components are the central and the upwind scheme. The central scheme approximates the velocity components by averaging the neighboring grid point velocity components. The upwind scheme gives an approximation of the velocity components depending on the direction of the flow. Both methods have their trade-offs. The central scheme does not add any non-physical dissipation to the system, but may suffer from spurious oscillations, particularly at higher Reynolds numbers [3,4]. The upwind scheme yields good stability properties [5], but adds significant numerical dissipation to the system. An analysis of the energy-conservation properties of the two schemes can be found in [6]. The above two methods for approximation of the cell-face velocities are the limit-case methods for the momentum equations, i.e., the central scheme corresponds to the no-flow situation, whereas the upwind scheme corresponds to inviscid flow. Higher order approximation
of the cell-face velocity components using for example (W)ENO reconstruction provides a more accurate numerical solution that is free of any spurious oscillations. However, these discretization methods are computationally more expensive than the conventional 3-point (in each spatial direction) discretization methods.

An accurate approximation of the cell-face velocities should take into account the balance of the convective and viscous forces acting on the fluid. This issue is tackled by the exponential/hybrid scheme, also known as locally exact schemes described in [7–9], which gives an approximation of the cell-face velocities as a weighted average of the central and the upwind scheme, with the weight function depending on the (local) Péclet number (ratio of the local convective and viscous forces). Apart from the convective and viscous forces, the cell-face velocities are also affected by the pressure gradient and the effect of rotational behavior (quantitatively given by the gradient of the transverse flux) of the fluid flow. In this paper we present methods for the computation of cell-face velocities using reduced momentum equations subject to suitable boundary conditions, giving rise to a local two-point boundary-value problem (BVP). Our goal is to derive approximations of the cell-face velocities, that are accurate, free of numerical oscillations, introduce no or very little numerical dissipation and result in a three-point coupling (per spatial direction).

The paper is organized as follows: In Section 2 the finite-volume method is briefly described, giving a mathematical formulation for the computation of cell-face velocity components. In Section 3, we derive an integral representation for the cell-face velocities by solving local BVPs. Sections 4 and 5 give the details about the close of the model and iterative computation of the cell-face velocities. The numerical results for the local BVP methods are discussed in Section 6 and conclusions are drawn in Section 7.

2. Finite volume discretization

We consider the two-dimensional incompressible Navier–Stokes equations in primitive variable formulation, i.e.,

$$\nabla \cdot \mathbf{u} = 0, \quad \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{uu}) = -\nabla p + \frac{1}{Re} \nabla^2 \mathbf{u},$$  

where \( \mathbf{u} = u(x, y, t) \mathbf{e}_x + v(x, y, t) \mathbf{e}_y, p \) and \( \text{Re} \) represent the velocity field, the pressure and the Reynolds number of the flow, respectively defined on a domain \( \Omega \). Initial and boundary conditions for (1) are specified in Section 6 where we present results for two test cases.

The above system of equations is discretized using a second-order accurate finite-volume method on a uniform staggered grid [10], as shown in Fig. 1. In this paper, we consider rectangular grids, but the discretization methods can be extended to orthogonal curvilinear grids. Introducing the vectors \( \mathbf{u} \) and \( \mathbf{p} \) for the numerical approximation of the velocity field and pressure, respectively, we can formulate the semi-discrete incompressible Navier–Stokes equations as follows

$$D\mathbf{u} = \mathbf{r}_1(t),$$  

$$\frac{\Omega}{dt}(\nabla \mathbf{u} + \nabla \mathbf{u}^T) = -\mathbf{N}(\mathbf{u}) + \frac{1}{\text{Re}} \Delta \mathbf{u} + \mathbf{G}(\mathbf{p}) + \mathbf{r}_2(t),$$

where \( \mathbf{D}, \mathbf{N}, \mathbf{L} \) and \( \mathbf{G} \) represent the discrete divergence, the non-linear convection, the viscous term and the gradient operator, respectively. The vector \( \mathbf{r}_1 \) gives the boundary conditions for the continuity equation (1a) and \( \mathbf{r}_2 \) contains the boundary conditions for the momentum equations (1b). The diagonal matrix \( \Omega = \text{diag}(\Omega_{xx}, \Omega_{yy}, \Omega_{zz}) \) contains the areas of the \( u \)- and \( v \)-control volumes shown in Fig. 1. For the control volume \( \Omega_{i+1/2,j} \), the \( u \)-component \( N^u \) of the discretized convective operator \( \mathbf{N}(\mathbf{u}) \) is given by

$$N^u_{i+1/2,j} = \Delta y (u_{i+1,j}^2 - u_{i,j}^2) + \Delta x ((uv)_{i+1/2,j+1/2} - (uv)_{i+1/2,j-1/2}) + \Delta y ((uv)_{i+1,j+1/2} - (uv)_{i+1,j-1/2}),$$

where \((uv)_{i+1,j+1/2} = u_{i+1,j+1/2}v_{i+1,j+1/2} \) and \( u_{ij} \) the numerical approximation of the velocity component \( u(x_i, y_j) \), etc.

Similarly, for the \( v \)-control volume \( \Omega^v_{i+1/2,j} \), the discrete convective operator \( N^v(\mathbf{u}) \) is given by

$$N^v_{i+1/2,j} = \Delta y (u_{i+1,j}^2 - u_{i,j}^2) + \Delta x ((uv)_{i+1/2,j+1/2} - (uv)_{i+1/2,j-1/2}) + \Delta x ((uv)_{i+1,j+1/2} - (uv)_{i+1,j-1/2}),$$

The details for the discrete operators \( \mathbf{D}, \mathbf{L} \) and \( \mathbf{G} \) can be found in the Appendix.

In order to compute \( N^u(\mathbf{u})_{i+1/2,j} \) we need the velocity components \( u_{i+1,j}, u_{i+1/2,j+1/2} \) and \( v_{i+1/2,j+1/2} \), the rest can be computed in a similar manner. These velocity components are not defined on the grid structure used for the discretization, and hence need to be approximated using neighboring grid velocities.

We begin with the approximation of the velocity component \( u_{i+1,j} \), the approximation of the other cell-face velocity components is described in Section 4. The central approximation of the velocity component \( u_{i+1,j} \) is given by

$$u_{i+1,j} = \frac{1}{2}(u_{i+1/2,j} + u_{i+3/2,j}).$$

The upwind approximation depends on the direction of the flow and is defined as

$$u_{i+1,j} = \begin{cases} u_{i+1/2,j}, & \text{if } u_{i+1/2,j} > 0 \land u_{i+3/2,j} > 0, \\ u_{i+3/2,j}, & \text{if } u_{i+1/2,j} < 0 \land u_{i+3/2,j} < 0, \\ \frac{1}{2}(u_{i+1/2,j} + u_{i+3/2,j}), & \text{if } u_{i+1/2,j}u_{i+3/2,j} \leq 0. \end{cases}$$
Alternatively, we compute the velocity component $u_{i+1,j}$ by solving the momentum equation locally, such that the computed cell-face velocity component depends on the local Péclet number and source terms like the pressure gradient and the cross-flux gradient. Consider the steady $u$-momentum equation, i.e.,

$$(u^2)_x + (uv)_y = -p_x + \frac{1}{Re}(u_{ex} + u_{yy}).$$

Restricting the solution of the above equation to the domain $x \in (x_{i+1/2}, x_{i+3/2}), y = y_j$, gives us the local BVP

$$(u^2 - \epsilon u_x)_x = -p_x - (uv - \epsilon u_y)_y, \quad \epsilon = 1/Re, \quad u(x_{i+1/2}, y_j) = u_{i+1/2,j}, \quad u(x_{i+3/2}, y_j) = u_{i+3/2,j}. \quad (5a)$$

$$(\delta_x p)_{i+1/2,j} = (\delta_y p)_{i+3/2,j}.$$

Fig. 2 shows the domain and the boundary conditions involved in the computation of $u_{i+1,j}$, the details of which are given in the following section.

3. Integral representation of cell-face velocities

We aim to compute the cell-face velocities from the corresponding local momentum equation, as an extension of the exponential scheme [7]. In the exponential scheme, the cell-face velocity is computed from a local, one-dimensional and constant-coefficient convection–diffusion equation. In our method, we consider the entire steady-state momentum equation, defined locally between two grid points, and derive the cell-face velocity from this equation. This allows us to derive several approximations of the interface velocity. In the absence of any source terms, the method reduces to the exponential scheme, which we refer to as the homogeneous local BVP method. As a first extension, we include the discrete pressure gradient as a source term in the governing equation, resulting in the inhomogeneous 1-D local BVP method. Second, to incorporate the two-dimensionality of the flow, we also include the gradient of the cross flux, giving the 2-D local BVP method for the cell-face velocity approximation.
To account for the nonlinearity of the convective terms in Eq. (1), it is desirable to solve a nonlinear local BVP, as demonstrated in [11] for the viscous Burgers equation. However solving Eq. (5a) using methods presented in [11] is difficult due to the presence of the source terms. To simplify the problem, the non-linear term \( u^2 \) is linearized as \( \dot{u}u \), where \( U \) is an approximation of the velocity component \( u_{i+1,j} \). Initially we take for \( U \) the (second order) central approximation of \( u_{i+1,j} \), and update it iteratively as explained in Section 5.

To complete the BVP, we have to specify the pressure gradient \(-p\) and the cross flux term \((uv - \epsilon v_y)\), in the right-hand side of (5a). We assume that the pressure \( p \) has a piecewise linear profile over \((x_{i+1/2}, x_{i+3/2})\), \( y = y_i \) with a jump at \( x = x_{i+1} \), as shown in Fig. 3, making the (discrete) pressure gradient \( \delta p \) piecewise constant. Consequently, the pressure gradient is given by

\[
\delta_x p(x) = \begin{cases} 
(\delta_x p)_{i+1/2,j} = \frac{1}{\Delta x} (p_{i+1,j} - p_{i,j}), & x_{i+1/2} \leq x \leq x_{i+1}, \\
(\delta_x p)_{i+3/2,j} = \frac{1}{\Delta x} (p_{i+2,j} - p_{i+1,j}), & x_{i+1} < x \leq x_{i+3/2}.
\end{cases}
\]

Moreover, we assume that the gradient of the cross-flux \((uv - \epsilon v_y)\) is constant over the interval \((x_{i+1/2}, x_{i+3/2})\) and replace it with its numerical approximation \( C_{i+1,j}^u \approx (uv - \epsilon v_y)_{x_{i+1}, y_i} \). With the above assumptions, Eq. (5a) reduces to a linear, inhomogeneous, quasi-one-dimensional ordinary differential equation, given by

\[
(Uu - \epsilon v_y)\Delta x = -\dot{u}p - C_{i+1,j}^u, \quad x_{i+1/2} < x < x_{i+3/2}, \quad y = y_i.
\] (6)

In the following derivation we suppress the \( y \)-dependence of the local BVP, \( u(x_i) := u(x_i, y_i) \), and introduce the following variables,

\[
a := U/\epsilon, \quad P := a \Delta x = URe \Delta x,
\]

where \( \Delta x = x_{i+3/2} - x_{i+1/2} \) and \( P \) is the grid Péclet number. Thus Eq. (6) can be rewritten as

\[
(a - u)\Delta x = \frac{1}{\epsilon} (\delta_x p + C_{i+1,j}^u).
\] (7)

First, we derive from the BVP (7)–(5b) an integral representation for \( u = u(x) \). Once we have determined \( u(x) \), we simply set \( u_{i+1} = u(x_{i+1}) \). Integrating Eq. (7) we get

\[
u_x - au = \frac{1}{\epsilon} \left( I_1(x) + C_{i+1,j}^u (x - x_{i+1}) + k_1 \right), \quad I_1(x) = \int_{x_{i+1}}^{x} \delta_x p(\xi) \, d\xi,
\] (8)

where \( k_1 \) is an integration constant. Note that we integrated Eq. (7) from \( x_{i+1} \) to \( x \in (x_{i+1/2}, x_{i+3/2}) \), as the pressure gradient \( \delta_x p \) has a jump at \( x = x_{i+1} \). Integrating Eq. (8) using the integrating factor formulation, \( u_x - au = e^{ax}(e^{-ax}u) \), and invoking the left boundary condition \( u(x_{i+1/2}) = u_{i+1/2,j} \), we get

\[
u(x) = e^{a(x-x_{i+1/2})}u_{i+1/2,j} + \frac{1}{\epsilon} \int_{x_{i+1/2}}^{x} e^{a(\xi-x)} l_1(\xi) \, d\xi + \frac{1}{\epsilon} C_{i+1,j}^u \int_{x_{i+1/2}}^{x} e^{a(\xi-x)} (\xi - x_{i+1}) \, d\xi + \frac{k_1}{\epsilon} \int_{x_{i+1/2}}^{x} e^{a(\xi-x)} \, d\xi.
\] (9)

It is convenient to introduce the normalized coordinate \( \sigma \) (0 \( \leq \sigma \leq 1 \)) on the interval \([x_{i+1/2}, x_{i+3/2}]\), given by

\[
\sigma := \sigma(x) = \frac{x - x_{i+1/2}}{\Delta x}.
\]

Expressed in terms of \( \sigma \), the formula for \( u \) reads

\[
u(\sigma) = e^{a\sigma} u_{i+1/2,j} + \frac{1}{\epsilon} \Delta x l_1(\sigma; P) + \frac{1}{\epsilon} C_{i+1,j}^u \Delta x^2 I_3(\sigma; P) + \frac{k_1}{\epsilon} \Delta x \varphi(P \sigma).
\]
where $\varphi(z) = (e^z - 1)/z$ and the integral functions $I_2$ and $I_3$ are defined as

$$I_2(\sigma; P) := \int_0^\sigma e^{\rho(\sigma - \eta)}I_1(x_{i+1/2} + \eta \Delta x) \, d\eta,$$

$$I_3(\sigma; P) := \int_0^\sigma e^{\rho(\sigma - \eta)}\left(\rho - \frac{1}{2}\right) \, d\eta.$$

Evaluating the integral $I_3(\sigma; P)$ gives

$$I_3(\sigma; P) = \frac{1}{P^2} \left( (1 - \frac{1}{2}P)(e^{\rho\sigma} - 1) - \sigma P \right).$$

Finally, invoking the right boundary condition $u(x_{i+3/2}) = u_{i+3/2,j}$ we can determine the integration constant $k_1$ and find

$$u(\sigma) = H_1(1 - \sigma; -P)u_{i+1/2,j} + H_1(\sigma; P)u_{i+3/2,j} + \frac{1}{\epsilon} \Delta x \left( I_2(\sigma; P) - H_1(\sigma; P)I_2(1; P) \right) +$$

$$\frac{1}{\epsilon} C_{i+1,j}^u \Delta x^2 \left( I_3(\sigma; P) - H_1(\sigma; P)I_3(1; P) \right),$$

where the function $H_1$ is defined as

$$H_1(\sigma; P) := \frac{e^{\rho\sigma} - 1}{e^{\rho\sigma} - 1}.$$

Summarizing, the integral representation $u(\sigma)$ can be expressed as the sum of three parts, i.e.,

$$u(\sigma) = u^h(\sigma) + u^p(\sigma) + u^c(\sigma),$$

$$u^h(\sigma) := H_1(1 - \sigma; -P)u_{i+1/2,j} + H_1(\sigma; P)u_{i+3/2,j},$$

$$u^p(\sigma) := \frac{1}{\epsilon} \Delta x \left( I_2(\sigma; P) - H_1(\sigma; P)I_2(1; P) \right),$$

$$u^c(\sigma) := \frac{1}{\epsilon} C_{i+1,j}^u \Delta x^2 \left( I_3(\sigma; P) - H_1(\sigma; P)I_3(1; P) \right),$$

with $u^h(\sigma)$ corresponding to the homogeneous solution of the local BVP (7) and $u^p(\sigma), u^c(\sigma)$ to the discrete pressure gradient and the gradient of the cross-flux, respectively. Note that $u^h, u^p$ and $u^c$ satisfy the boundary conditions, with

$$u^h(0) = u_{i+1/2,j}, \quad u^h(1) = u_{i+3/2,j},$$

$$u^p(0) = u^p(1) = 0,$$

$$u^c(0) = u^c(1) = 0.$$

To compute the interface velocity $u_{i+1,j}$ we substitute $\sigma = \frac{1}{2}$, i.e., $u_{i+1,j} = u(\frac{1}{2})$. The homogeneous component $u^h(\frac{1}{2})$ can be written as

$$u^h(\frac{1}{2}) = W(-P/2)u_{i+1/2,j} + W(P/2)u_{i+3/2,j},$$

where the function $W$ is defined as

$$W(z) := (e^z + 1)^{-1},$$

see also Fig. 4. Note that $W$ satisfies $0 \leq W(z) \leq 1$ and $W(z) + W(-z) = 1$ and consequently $u^h(\frac{1}{2})$ is a weighted average of $u_{i+1/2,j}$ and $u_{i+3/2,j}$, the weights determined by the local convection–diffusion balance. This approximation is in fact the

![Fig. 4. Function W.](image-url)
expansive scheme for the interface velocity. Evaluating the integral \( I_2(\sigma; P) \) for \( \sigma = 1/2 \) and \( \sigma = 1 \), we get

\[
I_2(1/2; P) = \Delta x (p_{i+1,j} - p_{i,j}) \frac{1}{P^2} \left((1 - \frac{1}{2}P)e^{P/2} - 1\right),
\]

\[
I_2(1; P) = e^{P/2}I_2(1/2; P) + \Delta x (p_{i+2,j} - p_{i+1,j}) \left(e^{P/2} - 1 - \frac{1}{2}P\right).
\]

Introducing the function

\[
H_2(P) := \frac{e^P - 1 - P}{P^2(e^P + 1)},
\]

and using the expressions for \( I_2(1/2; P) \) and \( I_2(1; P) \), the pressure component \( u^p(1/2) \) can be written as

\[
u^p(1/2) = -\frac{1}{4\epsilon} \Delta x^2 \left(H_2\left(-\frac{P}{2}\right)\delta_x p_{i+1/2,j} + H_2\left(P/2\right)\delta_x p_{i+3/2,j}\right).
\]

Finally, the cross flux component \( u^c(1/2) \) can be expressed as

\[
u^c(1/2) = -\frac{1}{8\epsilon} \Delta x^2 C^u_{i+1,j}\tanh(c/P/4),
\]

where \( \tanh(c(z)) = \tanh(z)/z \). Observe that the functions \( H_2(z) \) and \( \tanh(c(z)) \) satisfy

\[
H_2(-z) + H_2(z) = \frac{1}{2} \tanh(c(z)/2),
\]

implying that the expressions for \( u^p(\sigma) \) in (15) and \( u^c(\sigma) \) in (16) are equivalent for the special case \( \delta_x p_{i+1/2,j} = \delta_x p_{i+3/2,j} \).

The interface velocity \( u_{i+1,j} \) is now given by

\[
u_{i+1,j} = u^h(1/2) + u^p(1/2) + u^c(1/2).
\]

Depending on the terms in the right-hand side of Eq. (7) we get three possible methods for computing \( u_{i+1,j} \), viz., the homogeneous 1-D local BVP method \( u^h \), the inhomogeneous 1-D local BVP method \( u^h + u^p \) and the 2-D local BVP method \( u^h + u^p + u^c \), specified in the following.

3.1. Homogeneous 1-D local BVP method

If we ignore the pressure gradient and the gradient of the cross-flux, then solving the homogeneous equation

\[
(u_x - au)_x = 0
\]

gives the cell-face velocity component \( u_{i+1,j} \) as a weighted average of \( u_{i+1/2,j} \) and \( u_{i+3/2,j} \) as in Eq. (14). This is the homogeneous 1-D local BVP method. Fig. 4 shows the dependence of the function \( W \) on the Péclet number \( P \). Note that for \( P = 0 \), \( W(P/2) = W(-P/2) = 1/2 \), thus we get that \( u_{i+1,j} \) is the arithmetic average of \( u_{i+1/2,j} \) and \( u_{i+3/2,j} \) as in the central scheme. For \( P > 10 \) (or \( P < -10 \)) we have \( W(-P/2) \approx 1 \) (or \( W(P/2) \approx -1 \)), resulting in the upwind scheme. In fact, the homogeneous part can be expressed as a weighted average of the upwind and the central scheme:

\[
u^h(1/2) = (1 - 2W(|P|/2))u^\text{upwind} + 2W(|P|/2)u^\text{central}.
\]

3.2. Inhomogeneous 1-D local BVP method

The inclusion of the pressure gradient results in the inhomogeneous differential equation

\[
(u_x - au)_x = \frac{1}{\epsilon} \delta_x p,
\]

in which case the cell-face velocity component is expressed as \( u_{i+1,j} = u^h(1/2) + u^p(1/2) \). The contribution of the discrete pressure gradients \( \delta_x p_{i+1/2,j} \) and \( \delta_x p_{i+3/2,j} \) to the component \( u^p \) is controlled by the function \( H_2 \). Fig. 5 gives the dependence of \( H_2(P/2) \) on the Péclet number. It can be seen that for \( P = 0 \), \( \delta_x p_{i+1/2,j} \) and \( \delta_x p_{i+3/2,j} \) have equal contribution. However as the Péclet number \( P > 0 \) increases, we get that \( H_2(-P/2) \rightarrow 2/P \) and \( H_2(P/2) \rightarrow 4/P^2 \), as \( P \rightarrow \infty \), making the upwind pressure gradient \( \delta_x p_{i+1/2,j} \) dominant.

3.3. 2-D local BVP method

Finally, including both the pressure gradient and cross-flux gradient and solving differential equation (7), results in the cell-face velocity component given by Eq. (18). The term \( u^c(1/2) \) is proportional to the cross-flux gradient \( C^u_{i+1,j} \) and contains the factor \( \tanh(c(P/4)) \) depending on the Péclet number. Fig. 6 shows the variation in the values of \( \tanh(c(P/4)) \) with the change of Péclet number. It can be observed that \( \tanh(c(P/4)) \) attains its maximum for \( P = 0 \) and then asymptotically goes to 0 as \( |P| \rightarrow \infty \).
Approximation of the gradient of the cross-flux

In order to compute the term $u^i(1/2)$ given by Eq. (16), we need the cross-flux gradient $C_{i+1,j}^u$, $C_{i+3/2,j}^u$ is computed using $C_{i+1/2,j}^u$ and $C_{i+3/2,j}^u$ which are approximations of the gradient $(uv - \epsilon u_y)_{yj}$ at $(x_{i+1/2}, y_j)$ and $(x_{i+3/2}, y_j)$, respectively, with $C_{i+1/2,j}^u$ defined as

$$C_{i+1/2,j}^u = \frac{1}{\Delta y} \left( (uv)_{i+1/2,j+1/2} - (uv)_{i+1/2,j-1/2} \right) - \frac{\epsilon}{\Delta y^2} \left( u_{i+1/2,j+1} - 2u_{i+1/2,j} + u_{i+1/2,j-1} \right).$$

The gradient $C_{i+1,j}^u$ can be computed similarly. For the inhomogeneous 1-D local BVP approximation, it can be observed that for $|P| \gg 1$, the upwind pressure gradient is dominant and for $|P| \ll 1$, both upwind and downwind pressure gradients have equal contribution. Analogously, we compute $C_{i+1,j}^u$ as a weighted average:

$$C_{i+1,j}^u = W(-P/2)C_{i+1/2,j}^u + W(P/2)C_{i+3/2,j}^u,$$

such that $C_{i+1,j}^u$ is a weighted average of $C_{i+1/2,j}^u$ and $C_{i+3/2,j}^u$, cf. (14). For $|P| \ll 1$ the cross flux $C_{i+1,j}^u$ reduces in good approximation to the average of $C_{i+1/2,j}^u$ and $C_{i+3/2,j}^u$ and for $|P| \gg 1$ to the upwind value.

4. Closure of the model

In the previous section, we presented integral representations for approximating the cell-face velocity component $u_{i+1,j}$. To compute the discrete convective operator $(N^u(u))_{i+1/2,j}$ we also need $u_{i+1/2,j+1/2}$ and $v_{i+1/2,j+1/2}$. These velocity components are also computed using local BVPs, as shown in Fig. 7. For computing both $u_{i+1/2,j+1/2}$ and $v_{i+1/2,j+1/2}$, we make use of the homogeneous 1-D local BVP method discussed in Section 3. The velocity component $u_{i+1/2,j+1/2}$ is computed from the local boundary-value problem

$$Vu_y - \epsilon u_y y = 0, \quad y_j < y < y_{j+1}, \quad x = x_{i+1/2},$$

$$u(x_{i+1/2}, y_j) = u_{i+1/2,j}, \quad u(x_{i+1/2}, y_{j+1}) = u_{i+1/2,j+1}.$$ 

Fig. 5. Function $H_2$.

Fig. 6. Function $\tanh(z/4)$. 

$$V = \left( \begin{array} {cc} 0 & 1 \\ -\epsilon & 0 \end{array} \right).$$
where $V$ is an estimate for $v_{i+1/2,j+1/2}$. Likewise for $v_{i+1/2,j+1/2}$, we use the following local BVP

$$
\begin{align*}
U v_x - \epsilon v_{xx} &= 0, \quad x_i < x < x_{i+1}, \quad y = y_j + 1/2, \\
v(x_i, y_{j+1/2}) &= v_{i,j+1/2}, \quad v(x_{i+1}, y_{j+1/2}) &= v_{i+1,j+1/2},
\end{align*}
$$

with $U$ being an estimate for $u_{i+1/2,j+1/2}$. Note that Eq. (23a) is the linearized form of the equation

$$
(u v - \epsilon v_y)_y = 0,
$$

which is the cross-flux gradient in the $x$-momentum equation. Similarly, Eq. (24a) is the linearized form of,

$$
(u v - \epsilon v_x)_x = 0,
$$

which is the cross-flux gradient in the $y$-momentum equation.

All the methods described so far are analogously applied for discretization of the convective term $(N^v(u))_{i,j+1/2}$ given by Eq. (4). To compute the discretized operator $(N^v(u))_{i,j+1/2}$, we need the velocity components $v_{i,j+1}$, $u_{i+1/2,j+1/2}$ and $v_{i+1/2,j+1/2}$. The velocity component $v_{i,j+1}$ is computed using the local BVP

$$
\begin{align*}
(v_y - \delta v)_y &= \frac{1}{\epsilon}(\delta_y p + C_y^{i,j+1}), \quad x = x_i, \quad y_{j+1/2} < y < y_{j+3/2}, \\
v(x_i, y_{j+1/2}) &= v_{i,j+1/2}, \quad v(x_i, y_{3/2}) = v_{i,j+3/2},
\end{align*}
$$

where $b := V/\epsilon$, with $V$ being an estimate for $v_{i+1/2,j+1/2}$. Again, we assume that $\delta_y p$ is piecewise constant and that $C_y^{i,j+1}$, the numerical approximation of the cross-flux gradient, is constant over the interval $(y_{j+1/2}, y_{j+3/2})$. Solving the local BVP, as discussed in Section 3, we get that $v_{i,j+1}$ is given by

$$
\begin{align*}
v_{i,j+1} &= v^h(\frac{1}{2}) + v^p(\frac{1}{2}) + v^f(\frac{1}{2}), \\
v^h(\frac{1}{2}) &= W(-P^v/2)v_{i,j+1/2} + W(P^v/2)v_{i,j+3/2}, \\
v^p(\frac{1}{2}) &= -\frac{1}{4\epsilon}\Delta y^2(H_2(-P^v/2)(\delta_y p)_{i,j+1/2} + H_2(P^v/2)(\delta_y p)_{i,j+3/2}), \\
v^f(\frac{1}{2}) &= -\frac{1}{8\epsilon}\Delta y^2 C^v_{i,j+1} \tanh(\frac{P^v}{4}),
\end{align*}
$$

where the Péclet number $P^v$ is defined as $P^v := V \Delta y/\epsilon$. Analogous to $C^u_{i+1,j}$, the discrete cross-flux gradient $C^v_{i,j+1}$ is computed using the weighted average

$$
C^v_{i,j+1} = W(-P^v/2)C^v_{i,j+1/2} + W(P^v/2)C^v_{i,j+3/2}.
$$

5. Iterative computation of the cell-face velocities

The discussion so far gave details about the derivation of an integral representation of the cell-face velocity components by solving linearized local BVPs. However, the local BVPs that we intend to solve are non-linear. To account for this non-linearity, the cell-face velocities are computed iteratively. The algorithm listed in the frame gives the details of the iteration procedure.
As the initial condition, we take \( u(x, y, t) = -\sin(\pi x) \cos(\pi y)e^{-2\pi^2 t/Re} \), \( v(x, y, t) = \cos(\pi x) \sin(\pi y)e^{-2\pi^2 t/Re} \), and \( p(x, y, t) = \frac{1}{4}(\cos(2\pi x) + \sin(2\pi y))e^{-4\pi^2 t/Re} \) (26c).

As initial condition we take (26) at \( t = 0 \) and we prescribe periodic boundary conditions derived from (26) on the domain \( \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \).

We compare the local BVP methods with the central and the upwind scheme. For time integration, we use the second-order accurate, energy-conserving implicit Gauss method (described in [2]) with a fixed time step \( \Delta t = 10^{-2} \). The simulation is run from \( t = 0 \) to \( t = 100 \), i.e., 10,000 time steps are taken. Table 1 shows the discretization error \( \|e\|_{\infty} = \|u_{\text{numerical}} - u_{\text{exact}}\|_{\infty} \) for \( \text{Re} = 10^5 \) on a \( 20 \times 20 \) grid, at times \( t = 20 \), 100. At \( t = 20 \), the numerical solution obtained using the central scheme does not exhibit oscillations, and is significantly more accurate than the solutions of the upwind and homogeneous 1-D local BVP method. The central scheme and the inhomogeneous 1-D and 2-D local BVP methods appear to produce practically identical results at \( t = 20 \). However, at \( t = 100 \), the central scheme gives an oscillating solution, which severely affects the accuracy. The spurious oscillations can be observed through the streamlines of the numerical solutions, as shown in Fig. 8. For two-dimensional flows with smooth initial conditions, the incompressible

Table 1

<table>
<thead>
<tr>
<th>Method</th>
<th>( |e|_{\infty} (t = 20) )</th>
<th>( |e|_{\infty} (t = 100) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central</td>
<td>2.9897 \times 10^{-5}</td>
<td>1.7822</td>
</tr>
<tr>
<td>Upwind</td>
<td>8.2059 \times 10^{-1}</td>
<td>8.8515 \times 10^{-1}</td>
</tr>
<tr>
<td>Homogeneous 1-D local BVP</td>
<td>8.2051 \times 10^{-1}</td>
<td>8.8512 \times 10^{-1}</td>
</tr>
<tr>
<td>Inhomogeneous 1-D local BVP</td>
<td>2.9897 \times 10^{-5}</td>
<td>1.5480 \times 10^{-4}</td>
</tr>
<tr>
<td>2-D local BVP</td>
<td>2.9897 \times 10^{-5}</td>
<td>1.5480 \times 10^{-4}</td>
</tr>
</tbody>
</table>

For the iterative computation of \( u_{i+1,j} \), we make an initial estimate for \( U \) in the linearization of Eq. (5a). Corresponding to this estimate, we obtain the Péclet number given by \( P = U \Delta x/\epsilon \), which is used to compute the velocity component using one of the three methods described in the previous section. The computed cell-face velocity component is used to update the estimate \( U \) and the Péclet number and the procedure is repeated until the computed values converge to a given tolerance. The velocity components \( u_{i+1/2,j+1/2} \), \( v_{i+1/2,j+1/2} \) and \( v_{i,j+1} \) are computed iteratively in a similar manner.

6. Numerical results

We now compare the local BVP methods with the central and upwind schemes for the following test cases: the Taylor–Green vortex problem and the standard lid-driven square cavity flow.

6.1. Taylor–Green vortex

The Taylor–Green vortex in two dimensions admits an exact solution to the incompressible Navier–Stokes equations, given by

\[
\begin{align*}
    u(x, y, t) &= -\sin(\pi x) \cos(\pi y)e^{-2\pi^2 t/Re}, \\
    v(x, y, t) &= \cos(\pi x) \sin(\pi y)e^{-2\pi^2 t/Re}, \\
    p(x, y, t) &= \frac{1}{4}(\cos(2\pi x) + \sin(2\pi y))e^{-4\pi^2 t/Re}.
\end{align*}
\] (26a) (26b) (26c)

As initial condition we take (26) at \( t = 0 \) and we prescribe periodic boundary conditions derived from (26) on the domain \( \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \).

We compare the local BVP methods with the central and the upwind scheme. For time integration, we use the second-order accurate, energy-conserving implicit Gauss method (described in [2]) with a fixed time step \( \Delta t = 10^{-2} \). The simulation is run from \( t = 0 \) to \( t = 100 \), i.e., 10,000 time steps are taken. Table 1 shows the discretization error \( \|e\|_{\infty} = \|u_{\text{numerical}} - u_{\text{exact}}\|_{\infty} \) for \( \text{Re} = 10^5 \) on a \( 20 \times 20 \) grid, at times \( t = 20 \), 100. At \( t = 20 \), the numerical solution obtained using the central scheme does not exhibit oscillations, and is significantly more accurate than the solutions of the upwind and homogeneous 1-D local BVP method. The central scheme and the inhomogeneous 1-D and 2-D local BVP methods appear to produce practically identical results at \( t = 20 \). However, at \( t = 100 \), the central scheme gives an oscillating solution, which severely affects the accuracy. The spurious oscillations can be observed through the streamlines of the numerical solutions, as shown in Fig. 8. For two-dimensional flows with smooth initial conditions, the incompressible
Navier–Stokes equations have a unique solution [12, 13]. Unlike the central scheme, the numerical solutions obtained using the upwind, the homogeneous 1-D local BVP, the inhomogeneous 1-D local BVP and the 2-D local BVP scheme exhibit the same streamlines as the exact solution (26).

The breakdown of the central scheme is shown by the dissipation of the discrete kinetic energy. The central scheme results in a skew-symmetric convective operator, which has the property that the net work done by convection is zero. Further, for periodic boundary conditions it can be shown that work done by the pressure is also zero, thus the dissipation of discrete kinetic energy is only due to the discrete diffusion operator (for details see [14]). Moreover, the implicit Runge–Kutta methods presented in [2] are shown to be energy-conserving, thus for the central scheme the decay of the kinetic energy is solely due to the discrete diffusion operator \( L \). Fig. 9 shows the dissipation of the discrete kinetic energy for all the methods applied to the Taylor–Green vortex flow. As expected, the upwind discretization shows a high dissipation of kinetic energy. For a coarse grid and high Re (high Péclet numbers), the homogeneous 1-D local BVP method has a higher contribution from the upwind part \( u_{\text{upwind}} \) than from the central part \( u_{\text{central}} \), cf. Eq. (20). Consequently, here the homogeneous 1-D local BVP method has similar dissipation of energy as the upwind scheme. The inhomogeneous 1-D and 2-D local BVP methods exhibit the same dissipation of the kinetic energy as the central scheme, until the latter breaks down (at about \( t = 70 \)). Note that the difference between the kinetic energy of the exact solution (26), on the one hand, and the kinetic energy of all numerical solutions, on the other hand, is due to discretization errors.

Although the central scheme results in a symmetry-preserving discretization, which is energy-conserving [14], the presence of oscillations in its numerical solution results in non-physical dissipation of energy, as observed in Fig. 9. For the local BVP methods, the inclusion of the source terms appears to nullify the effect of numerical dissipation from the homogeneous part. A similar behavior is observed for the complete flux scheme. In [15] it is shown that including the
time derivative as a source term in the inhomogeneous flux effectively eliminates numerical diffusion introduced by the homogeneous part of the flux. Our current problem is reminiscent of this. Further analysis is required, however, this is beyond the scope of this paper.

Next, we investigate the convergence of spatial errors for a family of uniform grids ranging from $5 \times 5$ to $320 \times 320$ for $Re = 100$. For this case we use the fourth-order, four-stage explicit Runge–Kutta method, just as in [1], to ensure that for sufficiently small time step the temporal errors are negligible compared to the spatial errors. Fig. 10 shows the convergence of the errors ($\| u_{num} - u_{exact} \|_{\infty}$) for $Re = 100$ over a family of uniform grids, for a fixed time step $\Delta t = 10^{-3}$. It can be observed that the upwind discretization of the convective terms results in a first-order accurate scheme. The homogeneous 1-D local BVP method has a dominant upwind ($u_{upwind}$) contribution for coarser grids (due to higher Péclet numbers), hence exhibits first-order accuracy there. However, for finer grids (decreasing Péclet numbers), the central part ($u_{central}$) becomes dominant, leading to second-order convergence. The inclusion of the source terms in the local BVP improves the accuracy of the computed numerical solution significantly, resulting in the same error behavior (same order and same magnitude) as that of the central scheme.

The effect of the inclusion of the source terms on the spatial discretization, can be understood from the error analysis of the complete flux scheme in [16]. In this paper it is proven that it is essential to include the source terms in the numerical flux to obtain a uniformly second order accurate discretization. In our current discretization we employ standard central differences for the viscous terms, therefore the effect of the source terms should be incorporated in the convective flux.

6.2. Lid-driven cavity flow

This internal flow problem is very well suited for studying the effect of including the cross-flux term in the computation of cell-face velocity components due to the strong two-dimensionality of the flow. We investigate the lid-driven cavity flow...
for $Re = 100$, for fairly coarse grids, viz., $8 \times 8$ and $16 \times 16$, with at all four boundaries the no-slip boundary condition imposed, and with the results obtained by Ghia, Ghia and Shin [17] as reference solutions. Figs. 11 and 12 show the plots of the horizontal velocity component $u$ along the vertical center line of the cavity on the two grids. We compare the velocity profiles obtained using the central scheme, the upwind scheme and the three methods described in Section 3. The upwind method yields the least accurate results, as observed from Figs. 11 and 12. For the $8 \times 8$ grid, we have higher Péclet numbers, thus the homogeneous 1-D local BVP method behaves much like the upwind method. The inclusion of the discrete pressure gradient in the inhomogeneous 1-D local BVP method enhances the accuracy of the homogeneous 1-D local BVP method and gives velocity profiles similar to the central scheme as observed in both figures. As anticipated, the 2-D local BVP method performs the best of the considered methods, due to the strong two-dimensionality of the flow.

Table 2 shows the ratio of the computing times of the upwind method and the three local BVP methods over the computing time of the central method, for the $16 \times 16$ grid case. The costs of the local BVP methods appear to be only marginally higher than the cost of the central method.

For coarse grids the inclusion of the source terms in the local BVP for the approximation of the cell-face velocity components results in higher accuracy than the central scheme. Further, we observe that the inclusion of the source terms

---

**Fig. 11.** Horizontal velocity component along the vertical center line of the cavity, for $Re = 100$, on a uniform $8 \times 8$ grid.

**Fig. 12.** Horizontal velocity component along the vertical center line of the cavity, for $Re = 100$, on a uniform $16 \times 16$ grid.
appears to cancel the dissipative effects arising from the upwind part of the homogeneous part of the cell-face velocity component.

7. Conclusion

In this paper we presented methods for computing the cell-face velocity components involved in the discrete convective terms of the incompressible Navier–Stokes equations, using local BVPs. The cell-face velocities are computed by iteratively solving the nonlinear, local BVPs. Depending on the terms in the right-hand side of the local BVP we get three different methods, namely, the homogeneous 1-D local BVP method, the inhomogeneous 1-D local BVP method and the 2-D local BVP method. The homogeneous 1-D local BVP method provides an estimate for the cell-face velocity as a weighted average of the neighboring values depending on the Péclet number. In the limiting case $P \to 0$, it becomes identical to the central scheme and for $|P| \to \infty$ it reduces to the upwind scheme. The order of accuracy of the computed numerical solutions using the homogeneous 1-D local BVP method varies from first-order to second-order depending on the grid Péclet numbers.

In the inhomogeneous 1-D local BVP method we include the effect of the pressure gradient, where it is assumed that the pressure is piecewise linear. As an enhancement of the inhomogeneous 1-D local BVP method we have the 2-D local BVP method where we include both the pressure gradient and the cross-flux gradient. Unlike the homogeneous 1-D local BVP method, the inclusion of the source terms results in second-order accurate numerical solutions for all values of $P$ considered.

As observed for the case of the Taylor–Green problem, both the inhomogeneous 1-D local BVP method and the 2-D local BVP method exhibit perfect kinetic energy dissipation. The central scheme, which suffers from spurious oscillations, fails at higher Péclet numbers. The local BVP methods do not exhibit oscillations, which can be credited to the presence of the upwind component in the homogeneous part of the cell–face velocities.

For incompressible flow, a numerical scheme is energy conserving if the discrete convective operator is skew-symmetric, which is the case if the cell-face velocities are computed by central averaging. For the local BVP methods, the discrete convective operator is not skew-symmetric, thus the work done by the convective forces is formally not zero. Nevertheless, due to inclusion of the source terms, the inhomogeneous 1-D local BVP method and the 2-D local BVP method do not show any dissipative behavior. Further analysis of this is of interest.

Of the three local BVP methods, the 2-D local BVP method is to be preferred as it results in a more accurate approximation at marginally higher cost.

Acknowledgments

This work is part of the Industrial Partnership Programme (IPP) Computational sciences for energy research (project number: 101760) of the Foundation for Fundamental Research on Matter (FOM), Netherlands, which is part of the Netherlands Organization for Scientific Research (NWO). This research programme is co-financed by Shell Global Solutions International B.V., Netherlands.

Appendix. Spatial discretization

Applying the discretization presented in [10] to the incompressible Navier–Stokes equations (1) on staggered grids (Fig. 1) results in the semi–discrete system of equations

\[
\begin{align*}
\mathbf{D} \mathbf{u} & = \mathbf{r}_1(t), \\
\Omega \frac{d\mathbf{u}}{dt} & = -\mathbf{N}(\mathbf{u}) + \frac{1}{\text{Re}} \mathbf{L} \mathbf{u}(t) - \mathbf{Gp}(t) + \mathbf{r}_2(t),
\end{align*}
\]

where $\mathbf{D}, \mathbf{N}, \mathbf{L}$ and $\mathbf{G}$ represent the discrete divergence, the non-linear convection, the viscous term and the gradient operator, respectively. In the above semi-discrete system $\mathbf{u}(t) \in \mathbb{R}^{Np}$ and $\mathbf{p}(t) \in \mathbb{R}^{Np}$ are vectors with unknown point values of velocity and pressure in the center of the control volumes. It holds: $\Omega, \mathbf{N}, \mathbf{L} \in \mathbb{R}^{Np \times Np}$ and $\mathbf{G} \in \mathbb{R}^{Np \times Np}$. The boundary conditions for the continuity equation are described by the vector $\mathbf{r}_1(t) \in \mathbb{R}^{Np}$. $\mathbf{r}_2(\mathbf{u}, t) \in \mathbb{R}^{Np}$ is the vector with boundary conditions and forcing terms for the momentum equations. Eqs. (3) and (4) describe the discretized convective operator for the control volumes $\Omega_{i+1/2,j}$ and $\Omega_{i,j+1/2}$, respectively. The discrete divergence operator is given by

\[
(D \mathbf{u})_{ij} = \tilde{u}_{i+1/2,j} - \tilde{u}_{i-1/2,j} + \tilde{v}_{i,j+1/2} - \tilde{v}_{i,j-1/2},
\]

where $\tilde{u}$ and $\tilde{v}$ represent the fluxes at the cell faces.
where \( \bar{u}_{i+1/2,j} \) and \( \bar{v}_{i,j+1/2} \) represent the face-averaged velocity components computed using the mid-point rule \([18]\), i.e.,

\[
\bar{u}_{i+1/2,j} = \Delta y \, u_{i+1/2,j}, \quad \bar{v}_{i,j+1/2} = \Delta x \, v_{i,j+1/2}.
\]

The discrete diffusion term is given by

\[
(L^\nu(u))_{i+1/2,j} = \left( \frac{\partial \bar{u}}{\partial x} \right)_{i+1/2,j} + \left( \frac{\partial \bar{v}}{\partial y} \right)_{i+1/2,j} - \left( \frac{\partial \bar{v}}{\partial x} \right)_{i+1/2,j} - \left( \frac{\partial \bar{u}}{\partial y} \right)_{i+1/2,j} - \left( \frac{\partial \bar{u}}{\partial y} \right)_{i+1/2,j-1/2},
\]

\[
(L^\nu(u))_{i,j+1/2} = \left( \frac{\partial \bar{u}}{\partial x} \right)_{i+1/2,j} - \left( \frac{\partial \bar{v}}{\partial y} \right)_{i+1/2,j} + \left( \frac{\partial \bar{v}}{\partial x} \right)_{i+1/2,j+1} - \left( \frac{\partial \bar{u}}{\partial y} \right)_{i+1/2,j+1} - \left( \frac{\partial \bar{v}}{\partial y} \right)_{i+1/2,j}.
\]

Finally, the discrete gradient operator \( G = (G_u, G_v) \) is given by

\[
G_u(p)_{i+1/2,j} = \tilde{p}_{i+1,j} - \bar{p}_{i,j}, \quad G_v(p)_{i,j+1/2} = \tilde{p}_{i+1,j} - \bar{p}_{i,j}.
\]

In the incompressible Navier–Stokes equations, the pressure, which is a Lagrange multiplier, ensures that the velocity-field is divergence free and is only determined up to an additive constant (unless pressure is prescribed at a boundary). The evolution of the pressure is governed by the pressure-Poisson equation, which is derived by taking the time-derivative of the continuity equation and then using the momentum equation, i.e.,

\[
\mathcal{L} \, p = D \, F(u, t) - r_1(t),
\]

where \( \mathcal{L} = DG \) is the discrete Laplace operator and

\[
F(u, t) = -N(u) + \frac{1}{Re} Lu(t) + r_2(t).
\]

In the above expression for \( F(u, t) \), \( \Omega^{-1} \) has been absorbed by the discrete operators \( N, L, G \) and \( r_2 \). The boundary conditions for the pressure-Poisson equation can be derived using the boundary conditions for the momentum equations, as described in \([19–21]\).

References
