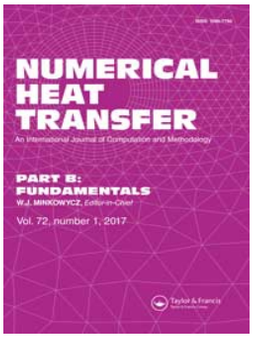

Taylor & Francis
Taylor & Francis Group

Numerical Heat Transfer, Part B: Fundamentals
An International Journal of Computation and Methodology

ISSN: 1040-7790 (Print) 1521-0626 (Online) Journal homepage: https://www.tandfonline.com/loi/unhb20

Implementation of boundary conditions in pressure-based finite volume methods on unstructured grids

I. Sezai

To cite this article: I. Sezai (2017) Implementation of boundary conditions in pressure-based finite volume methods on unstructured grids, Numerical Heat Transfer, Part B: Fundamentals, 72:1, 82-107, DOI: 10.1080/10407790.2017.1338077

To link to this article: https://doi.org/10.1080/10407790.2017.1338077

Published online: 26 Jun 2017.

Submit your article to this journal

Article views: 448

View related articles

View Crossmark data

Citing articles: 4 View citing articles

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=unhb20

NUMERICAL HEAT TRANSFER, PART B
2017, VOL. 72, NO. 1, 82–107
https://doi.org/10.1080/10407790.2017.1338077

Taylor & Francis
Taylor & Francis Group

Check for updates

Implementation of boundary conditions in pressure-based finite volume methods on unstructured grids

I. Sezai

Mechanical Engineering Department, Eastern Mediterranean University, Mersin, Turkey

ABSTRACT
In this study, the implementation of boundary conditions for the Navier–Stokes and the energy equations, including the pressure and pressure correction equations, are presented in the context of finite volume formulation on cell-centered, colocated unstructured grids. The implementation of boundary conditions is formulated in terms of the contribution of boundary face of a cell to the coefficients of the discretized equation for either Dirichlet- or Neumann-type boundary conditions. Open boundaries through which the flow is not fully developed are also considered. In this case, a data reconstruction method is proposed for finding the boundary values of the variables at the correction stage. The validity of implementations is checked by comparing the results with some well-known benchmark problems.

ARTICLE HISTORY
Received 8 February 2017
Accepted 16 May 2017

1. Introduction

In recent years, there has been considerable progress in the development of solution methods of flow and heat transfer equations on unstructured grids. For incompressible flows, the pressure-based methods based on SIMPLE family of algorithms, such as SIMPLE [1], SIMPLEC [2], SIMPLER [3], and PISO [4], have become the predominant methodology in the computational fluid dynamics (CFD) community. In pressure-based methods, the continuity equation is expressed in terms of a pressure or pressure correction. Then, the momentum, continuity, and other transport equations are solved in a segregated manner, where iterations are performed to provide the coupling between them. A lot of effort have been directed to obtain more robust algorithms in the context of cell-centered finite volume formulation of the unstructured grids with a collocated arrangement using the pressure-based methods in the last decade as exemplified in references [5–10]. The versatility of unstructured grids eventually led to the adoption of technique by the commercial software which resulted in the widespread use of CFD that was once used by a relatively small number of experts. The success of the control volume methods based on unstructured grids was made possible with a shift away from the staggered grid techniques [3] to colocated storage approaches [11–14] where the pressure and velocities are stored at the same locations. The central factor in this shift was the momentum interpolation method (MIM) of Rhie and Chow [15] which was proposed to circumvent the so-called checkerboard problem of the colocated grid storage methods.

Besides proper discretization of the governing differential equations and robust solution methods, proper specification and implementation techniques of the boundary conditions are of equal importance. This may not be a problem for the wall boundaries where the flow conditions can be readily determined. However, for open boundaries, where the flow is not developed, the specification and implementation of the boundary conditions are not so clear.

CONTACT I. Sezai ibrahim.sezai@emu.edu.tr Mechanical Engineering Department, Eastern Mediterranean University, Magosa, Mersin-10, Turkey.
Color versions of one or more of the figures in this article can be found online at www.tandfonline.com/unhb.
© 2017 Taylor & Francis

Nomenclature

A, $\mathbf{A}$	area in scalar or vector form
C	c_p for energy or 1 for momentum equations
$\mathbf{C}$	coefficient in vector form defined in appendix
c_p	constant pressure specific heat
d	distance from a cell center
$\mathbf{e}$	unit vector
$\mathbf{f}$	body forces per unit volume
fr	interpolation factor
g	gradient
H	height
$\mathbf{I}$	unit tensor
J, $\mathbf{J}$	flux in scalar or vector form
k	thermal conductivity
L	length
$\dot{m}$	mass flow rate
$\hat{m}$	pseudomass flow rate
$\mathbf{n}$	unit vector in normal direction
nb(P)	neighbors of cell P
$n_f(P)$	number of faces of cell P
p	pressure
p'	pressure correction
S, $\mathbf{S}$	source term of the discretized equation in scalar or vector form
s, $\mathbf{s}$	source term of the differential equation in scalar or vector form
t	time
u, v, w	velocity components in x, y, z-directions
V	volume
$\dot{V}$	volume flow rate
$\hat{V}$	pseudovolume flow rate
$\mathbf{v}$	velocity vector
x, y, z	Cartesian coordinates
α	relaxation parameter
β	coefficient of thermal expansion
ϕ	dependent variable
Γ	diffusion coefficient
μ	fluid viscosity
ρ	fluid density

Subscripts

b	boundary face
f	face
N	neighbor node
P	central node
t	tangential

Superscripts

cd	cross derivative term
conv	convection
dif	diffusion
H	higher order convection terms
l	current time
n	current iteration
T	transpose
U	upwind

Although a lot of effort have been spent in the development and solution of the finite volume methods (FVM) on unstructured collocated grids, relatively little study exists on the application of boundary conditions. For a developed flow at open boundaries, a methodology for incorporating pressure boundary conditions in the context of cell-centered, unstructured meshes, composed of arbitrary polyhedra has been presented by Mathur and Murthy [16]. The boundary pressure and mass fluxes were calculated through Bernoulli equation and the correction equation of the SIMPLE algorithm. The implementation of the constant pressure boundary conditions for flow in complex domains is presented by Kelkar and Choudhury [17], where the boundary velocities were constructed through extrapolation of the momentum equations. The method was applied for predicting forced flows in branched ducts and in buoyancy-driven external and internal flows.

Xu and Barton [18] give the implementation of boundary conditions for momentum, pressure, and pressure correction equations in the context of cell-centered finite volume formulation using unstructured grids and collocated variable arrangement. The formulation of boundary conditions is given for the Dirichlet case only, while the Neumann-type boundary condition is expressed in terms of the boundary value calculated explicitly from the boundary flux. The open boundary conditions for both the fully developed case with an imposed exit pressure and for the truncated domain, where the flow is undeveloped at the exit are formulated. For the latter case, the boundary pressures are linearly extrapolated from values at the boundary cell and its nearest neighbor, the details of which are given in the reference [19].

The open boundary condition case where the flow is not fully developed is not well established and research on this subject is still continuing. The importance of the issue lies in the fact that, fully developed flow can often be realized using a sufficiently long flow channel which necessitates large memory and long computational time. The computational load can be decreased by truncating the computational domain where the outflow boundary is placed at a position such that the flow is

not fully developed and even may have incoming flow if placed in the region of vortices. Most of the studies on open boundaries are related with the finite element formulations which use surface traction conditions on the boundary [19–23]. Studies on finite volume formulations of open boundaries are rather scarce. Shyy [24] used a pressure-based FVM to study the flow in expansion channels with open boundaries using a staggered grid. The outlet velocity was calculated using a zero velocity gradient condition in the streamline direction. Sani and Saidi [25] used the least square gradient reconstruction method to estimate the variables on open boundaries on the basis of a segregated flow solver based on FVM. However, they used ghost cells to implement the method implicitly. They also enforced overall mass conservation over the entire domain which was found to be a necessary condition for the convergence of the open boundary problems [26]. Overall mass conservation was also used in later studies [9, 18] related with open boundaries. The method proposed in Xue and Barton [9, 18] was later extended to problems where the flow channel splits to multiple branches having truncated open boundaries [27].

In this study, the implementation of Dirichlet- and Neumann-type boundary conditions for the flow equations in the context of cell-centered unstructured meshes is presented by expressing the contribution of a boundary face to the coefficients and source term of the discretized equations in generic form. Special attention is paid to the formulation of the boundary conditions for open boundaries where the flow conditions are either known, when the flow is fully developed, or unknown, such as an open boundary where the flow is not fully developed, which results from the truncation of a long domain to reduce the computational time and the memory usage. To this end, the formulation of momentum, pressure, and the pressure correction equations is presented in detail. Solutions are presented to verify the implementations with standard benchmark problems. These include steady flow over a backward-facing step using truncated domain, unsteady flow over a cylinder, natural convection from a horizontal isothermal cylinder, and natural convection in an open cavity.

2. Discretization of the generic transport equation

The conservation equation for a general variable ϕ can be written in the form:

$$\frac{\partial}{\partial t}(C\rho\phi) + \nabla\cdot(C\rho\mathbf{v}\phi) = \nabla\cdot(\Gamma\nabla\phi) + s_\phi \tag{1}$$

where $C = c_p$ for the energy equation and $C = 1$ for momentum equations. The general variable ϕ may be a scalar or a vector and s_ϕ is the source consisting of the remaining terms of governing differential equation. In general, it can be written in linearized form as $s_\phi = s_c + s_p\phi$ where s_c is the constant explicit part and s_p is the implicit part of the linearization. Using the first-order fully implicit method for the discretization of the transient term, the differential equation can be expressed in terms of an algebraic equation as:

$$\frac{(C\rho V\phi)_P^l - (C\rho V\phi)_P^{l-1}}{\Delta t} + \left(\sum_{f=1}^{n_f(P)} C_f J_f^{\text{conv}}\right)^l + \left(\sum_{f=1}^{n_f(P)} J_f^{\text{dif}}\right)^l = \left(s_\phi V\right)_P^l \tag{2}$$

where the superscripts l and $l-1$ refer to current and previous time levels, respectively. The subscripts P and f refer to the center and faces of the control volume, respectively. The summation is all over the bounding faces of cell P where $n_f(P)$ is the number of faces of cell P. The convective and diffusive fluxes, J_f^{conv} and J_f^{dif} are

$$J_f^{\text{conv}} = \rho\mathbf{v}_f\cdot\mathbf{A}_f = \dot{m}_f\phi_f, \quad J_f^{\text{dif}} = -\Gamma_f\nabla\phi_f\cdot\mathbf{A}_f \tag{3}$$

where $\dot{m}_f$ is the mass flow rate across face f, the calculation of which will be deferred to a later section.

To avoid the false diffusion due to the use of first-order upwind method, ϕ_f value should be estimated by a higher order method which is at least second-order accurate. Also, to avoid numerical

instabilities, the deferred correction technique of Khosla and Rubin [28] is used. That is, the convective flux at the cell faces are written as the sum of the upwind value and other higher order terms which are evaluated at the previous iteration $(n-1)$ as:

$$J_f^{\text{conv}} = \dot{m}_f \left[\phi_f^U + (\phi_f^H - \phi_f^U)^{n-1} \right] \tag{4}$$

where ϕ_f^U is found from first-order upwind method, expressed in a compact form as:

$$\dot{m}_f \phi_f^U = \max(\dot{m}_f, 0.)\phi_P + \min(\dot{m}_f, 0.)\phi_N \tag{5}$$

and ϕ_f^H is the face value of ϕ found from a higher order method. Also, the value of ϕ_f^H should be bounded to eliminate oscillations in the regions of high gradients during the solution. Total variation diminishing schemes [29, 30] are one of the approaches in estimating ϕ_f^H. Another approach is using the normalized variable diagram formulation, which was originally proposed by Leonard [31]. However, no such bounding schemes are used in the present study. Instead, the linear second-order upwind difference scheme for the convection term, which is given by (Figure 1):

$$\begin{aligned} \phi_f^H &= \phi_f^U + \nabla\phi_P \cdot \mathbf{Pf}, \quad \phi_f^U = \phi_P \quad \text{for } \dot{m} > 0 \\ \phi_f^H &= \phi_f^U + \nabla\phi_N \cdot \mathbf{Nf}, \quad \phi_f^U = \phi_N \quad \text{for } \dot{m} < 0 \end{aligned} \tag{6}$$

is used. The diffusion term at face f can be written using the overrelaxed approach where the cell face area $\mathbf{A}_f$ is decomposed as $\mathbf{A}_f = \mathbf{A}_d + \mathbf{A}_t$ such that $\mathbf{A}_t$ is perpendicular to $\mathbf{A}_f$. The result is

$$J_f^{\text{dif}} = -\Gamma_f \nabla\phi_f \cdot \mathbf{A}_f = -\Gamma_f \underbrace{\frac{\phi_N - \phi_P}{d_{PN}} A_d}_{\text{orthogonal term}} \underbrace{-\Gamma_f \overline{\nabla\phi_f} \cdot \mathbf{A}_t - \Gamma_f \frac{(\nabla\phi_N - \nabla\phi_P) \cdot \mathbf{f_1 f}}{d_{PN}} A_d}_{\text{non-orthogonal terms}} \tag{7}$$

where the overbar indicates linearly averaged value between points P and N, $A_d = A_f/(\mathbf{e}_{PN} \cdot \mathbf{n}_f)$ is a component of $\mathbf{A}_f$ in the direction of line PN using the overrelaxed approach, A_f being the magnitude of $\mathbf{A}_f$ and

$$\mathbf{A}_t = \mathbf{A}_f - A_d \mathbf{e}_{PN}, \quad \mathbf{f_1 f} = \mathbf{Pf} - \frac{\mathbf{Pf} \cdot \mathbf{n}}{\mathbf{PN} \cdot \mathbf{n}} \mathbf{PN}, \quad \frac{A_d}{d_{PN}} = \frac{\mathbf{A}_f \cdot \mathbf{A}_f}{\mathbf{A}_f \cdot \mathbf{PN}} \tag{8}$$

The orthogonal term is treated implicitly, whereas the nonorthogonal terms are treated explicitly. Expressing the convection and diffusion terms in a deferred correction manner, Eq. (2) can be written in the form:

$$a_P \phi_P + \sum_{N=\text{nb}(P)} a_N \phi_N = S \tag{9}$$

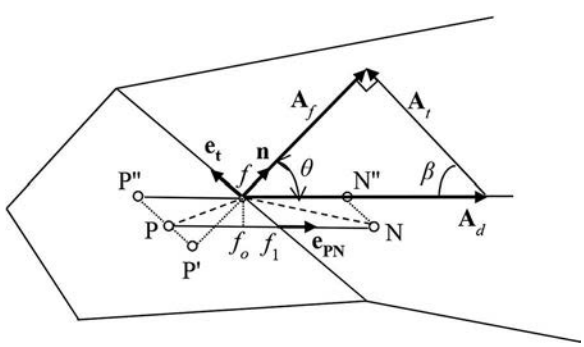

Figure 1. Diffusion flux across a surface.

where

$$a_N = -\frac{\Gamma_f A_d}{d_{PN}} + C_f \min\left[\dot{m}_f, 0\right], \quad a_P = \sum_{N=\mathrm{nb}(P)} -a_N + a_P^t - s_p V, \quad a_P^t = \frac{C\rho V}{\Delta t},$$

$$S = (a_P^t \phi_P)^{l-1} + s_c V + \sum_{f=1}^{n_f(P)} S_f, \quad S_f = S_f^{\mathrm{conv}} + S_f^{\mathrm{dif}}, \tag{10}$$

$$S_f^{\mathrm{conv}} = -\left[C_f \dot{m}_f (\phi_f^H - \phi_f^U)\right]^{n-1}, \quad S_f^{\mathrm{dif}} = \left[\Gamma_f \overline{\nabla \phi_f} \cdot \mathbf{A}_t + \Gamma_f \frac{(\nabla \phi_N - \nabla \phi_P) \cdot \mathbf{f_1 f}}{d_{PN}} A_d\right]$$

2.1. Underrelaxation

To improve the convergence characteristics of the steady flow equations, underrelaxation is usually applied [3]. After relaxation, Eq. (9) becomes

$$\frac{1}{\alpha} a_P \phi_P + \sum_{N=\mathrm{nb}(P)} a_N \phi_N = S + \frac{1-\alpha}{\alpha} a_P \phi_P^{n-1} \tag{11}$$

where ϕ_P^{n-1} is the previous iteration value of ϕ and the last term is treated as an additional source. The relaxation parameter α is between 0 and 1 for steady flows and it is set to 1 for unsteady flows.

2.2. Contribution of a boundary face to the generic equation

For a control volume near a boundary, there is no neighboring node N (Figure 1). In this case, node N is on the boundary replacing point f as $N \rightarrow f$, so that $\mathbf{f_1 f} = 0$ and $\mathbf{PN} = \mathbf{Pb}$ where f is replaced by b in reference to "boundary." As a result, at the boundaries, the following expressions and approximations are used:

$$\mathbf{f_1 f} = 0, \ \mathbf{PN} = \mathbf{Pb}, \quad \mathbf{a}_b \approx \mathbf{a}_P, \ (a_P^t)_b \approx a_P^t \tag{12}$$

Then, the diffusion flux equation (7) at a boundary face becomes

$$\begin{aligned} J_b^{\mathrm{dif}} &= -\Gamma_b \nabla \phi_b \cdot \mathbf{A}_b = -\Gamma_b \frac{\phi_b - \phi_P}{d_{Pb}} A_d - \Gamma_b \overline{\nabla \phi_b} \cdot \mathbf{A}_t \\ &= (a_N^{\mathrm{dif}})_b (\phi_b - \phi_P) - S_b^{\mathrm{dif}} \end{aligned} \tag{13}$$

where

$$(a_N^{\mathrm{dif}})_b = -\Gamma_b \frac{A_d}{d_{Pb}}, \quad S_b^{\mathrm{dif}} = \Gamma_f \overline{\nabla \phi_b} \cdot \mathbf{A}_t \tag{14}$$

Then, the contribution of a boundary face b to the coefficients and source term are given by:

$$\begin{aligned} S_b &= S_b^{\mathrm{dif}} + S_b^{\mathrm{conv}} = \Gamma_f \nabla \phi_b \cdot \mathbf{A}_t - \left[C_b \dot{m}_b (\phi_b^H - \phi_b^U)\right] \\ (a_N)_b &= (a_N^{\mathrm{dif}})_b + (a_N^{\mathrm{conv}})_b = -\Gamma_b \frac{A_d}{d_{Pb}} + C_b \min[\dot{m}_b, 0] \end{aligned} \tag{15}$$

2.3. Boundary conditions

The source term S and coefficients a_P and a_N in Eq. (10) are valid for interior cells where no face of a control volume coincides with a domain boundary. However, near-boundary cells have at least one face which is coincident with the domain boundary. For these near-boundary cells, a_P and a_N can be written as:

$$a_P = \tilde{a}_P - (a_N)_b, \quad S = \tilde{S} + S_b, \tag{16}$$

where the superscript $\sim$ refers to values obtained by excluding the contribution of the boundary face b.

2.3.1. Types of boundary conditions

The boundary conditions which are used in flow equations can be written as a combination of the following two basic types (1) specified value of ϕ (Dirichlet) and (2) specified normal gradient of ϕ (Neumann).

2.3.1.1. Dirichlet boundary condition (specified value of ϕ at the boundary). Sometimes, the value of ϕ at a boundary is specified as ϕ_b. For a near-boundary cell, the boundary node b can be regarded as a neighbor node and treated explicitly by transferring $a_N\phi_N$ to the source term. That is, substituting Eq. (16) into Eq. (9), an equation can be written for a near-boundary cell as:

$$\left[\tilde{a}_P - (a_N)_b\right]\phi_P + \sum_{\substack{N=\mathrm{nb}(P)\\ N\neq b}} a_N\phi_N = \tilde{S} + S_b - (a_N)_b\phi_b \tag{17}$$

Then, the coefficients and source term of a near-boundary cell are updated by adding the contribution of boundary face b;

$$\begin{aligned} S &= \tilde{S} + S_b - (a_N)_b\phi_b \\ a_P &= \tilde{a}_P - (a_N)_b \\ a_N &= 0 \end{aligned} \tag{18}$$

where $(a_N)_b$, $(a_P)_b$, and S_b are given in Eq. (15).

2.3.1.2. Neumann boundary condition (specified normal gradient at a boundary). In some cases, the value of ϕ at a boundary is not known, but the value of normal gradient, g_n, at the boundary is specified. Mathematically

$$g_\mathrm{n} = \nabla\phi_b \cdot \mathbf{n}_b \tag{19}$$

so that the diffusion flux can be expressed in terms of g_n as:

$$J_b^{\mathrm{dif}} = -\Gamma_b\nabla\phi_b \cdot \mathbf{A}_b = -\Gamma_b A_b\nabla\phi_b \cdot \mathbf{n}_b = -\Gamma_b A_b g_\mathrm{n} \tag{20}$$

Then, using Eq. (13), ϕ_b can be expressed in terms of g_n as:

$$\phi_b = \phi_P + \frac{d_{Pb}}{A_d}\left(A_b g_\mathrm{n} + \nabla\phi_b \cdot \mathbf{A}_t\right) \tag{21}$$

Substituting into Eq. (17) yields

$$\tilde{a}_P\phi_P + \sum_{\substack{N=\mathrm{nb}(P)\\ N\neq b}} a_N\phi_N = \tilde{S} + S_b - (a_N)_b\frac{d_{Pb}}{A_d}\left(A_b g_\mathrm{n} + \nabla\phi_b \cdot \mathbf{A}_t\right) \tag{22}$$

Then, the coefficients and source term of a near-boundary cell become:

$$\begin{aligned} S &= \tilde{S} + S_b - (a_N)_b \frac{d_{Pb}}{A_d}(A_b g_{\mathrm{n}} + \nabla\phi_b \cdot \mathbf{A}_t) \\ a_P &= \tilde{a}_P \\ a_N &= 0 \end{aligned} \tag{23}$$

Solution of Eq. (22) yields the ϕ values at the interior cells. The boundary value ϕ_b is required for the gradient calculations, which can be calculated from Eq. (21).

2.3.2. Reconstruction of variables at problematic boundaries

The implementation of boundary conditions outlined above is straightforward. However, there are situations where the boundary conditions are not so obvious. For example, at the exit of a flow channel which is not sufficiently long, the flow is not fully developed so that Neumann boundary condition is not applicable. Another example is the situation when the computational domain of the classical backward-facing step flow problem is truncated to reduce the computational costs.

In such cases, a proper method is needed for the estimation of variables at the boundary. The method proposed in this study involves the simultaneous solution of the Taylor series approximation relation, with the gradient relation at the center of the near-wall cell, found from the available flow data of the neighboring cells. Two methods of calculating the gradients will be considered; (1) Gauss method and (2) least square method.

2.3.2.1. Reconstruction using Gauss method.

2.3.2.1.1. General case. For a near-boundary control volume, face f in Figure 1 becomes the boundary face b. Let ϕ denotes any variable of interest. Replacing f with b, the boundary value of variable ϕ can be obtained using Taylor series approximation as

$$\phi_b = \phi_P + \nabla\phi_P \cdot \mathbf{Pb} \tag{24}$$

However, this equation cannot be used directly to find ϕ_b, since $\nabla\phi_P$ is also a function of ϕ_b. Using the Gauss method for finding the gradient, we have

$$\nabla\phi_P = \frac{1}{V_P}\sum_{f=1}^{N_f} \phi_f \mathbf{A}_f \tag{25}$$

where the summation is all over the faces of the control volume of point P. Inserting Eq. (25) in Eq. (24), we get

$$\phi_b = \frac{\phi_P + \left(\frac{1}{V_P}\sum_{\substack{f=1 \\ f\neq b}}^{N_f} \phi_f \mathbf{A}_f\right) \cdot \mathbf{Pb}}{1 - \frac{1}{V_P}\mathbf{A}_b \cdot \mathbf{Pb}} \tag{26}$$

where the summation is over the faces of the control volume, excluding the boundary face.

2.3.2.1.2. Special case: Zero normal gradient at boundary. The method outlined above can be extended to include cases where the normal gradient at the boundary is zero. From Figure 1 for a zero normal gradient boundary condition, we can write

$$\phi_b = \phi_{P'} = \phi_P + \nabla\phi_P \cdot \mathbf{PP'} \tag{27}$$

where $\mathbf{PP'} = \mathbf{Pb} - (\mathbf{Pb} \cdot \mathbf{n})\mathbf{n}$. In this equation, both ϕ_b and $\nabla\phi_P$ are unknown. Using the Gauss method for finding $\nabla\phi_P$ and inserting in Eq. (27), an equation similar to Eq. (26) is obtained except that $\mathbf{Pb}$ is replaced with $\mathbf{PP'}$.

2.3.2.2. Reconstruction using the least square method.

2.3.2.2.1. General case. From Appendix A, the general formula for the reconstruction of the boundary value is

$$\phi_b = \frac{\phi_P + \left(\sum_{\substack{i=1\\ i\neq b}}^{\mathrm{nb}(P)} \mathbf{C}_i w_i(\phi_i - \phi_P)\right)\cdot \mathbf{Pb} - \mathbf{C}_b w_b \phi_P \cdot \mathbf{Pb}}{1 - \mathbf{C}_b w_b \cdot \mathbf{Pb}} \tag{28}$$

where the summation is over the neighboring points, excluding the boundary.

2.3.2.2.2. Special case: Zero normal gradient at boundary. Using the same reasoning as in the case of Gauss method of gradient evaluation, when the normal gradient at the boundary is zero, then ϕ_b is found from Eq. (28) by replacing **Pb** with **PP**′.

3. Discretization of the momentum equations

The momentum equations can be written in the form of the general transport equation as:

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla\cdot(\rho \mathbf{v}\mathbf{v}) = \nabla\cdot(\mu\nabla\mathbf{v}) + \mathbf{s}_\mathbf{v} \tag{29}$$

where the viscosity, μ, corresponds to Γ and the source term is

$$\mathbf{s}_\mathbf{v} = \nabla\cdot(\mu\nabla\mathbf{v}^{\mathrm{T}}) - \nabla p\mathbf{I} + \mathbf{f} \tag{30}$$

and **f** is the body forces per unit volume, such as buoyancy and electromagnetic forces and p is the hydrostatic pressure with **I** being the unit tensor. The term $\nabla\cdot(\mu\nabla\mathbf{v}^{\mathrm{T}})$ contains the cross derivatives in the momentum equations, which are treated in a deferred correction manner, so that they are considered as explicit source terms. The cross derivative source terms are zero for incompressible fluids if μ is constant. The discretized form of Eq. (29) is

$$\frac{(\rho V\mathbf{V})_P^l - (\rho V\mathbf{V})_P^{l-1}}{\Delta t} + \left(\sum_{f=1}^{n_{fP}} \mathbf{J}_f^{\mathrm{conv}}\right)^l = -\left(\sum_{f=\mathrm{nb}(P)} \mathbf{J}_f^{\mathrm{dif}}\right)^l - \left(\sum_{f=\mathrm{nb}(P)} \mathbf{J}_f^{\mathrm{cd}}\right)^l - \left(\sum_{f=1}^{n_f(P)} p_f \mathbf{A}_f \mathbf{I}\right)^l + (\mathbf{f}_P V_P)^l \tag{31}$$

where $\mathbf{J}_f^{\mathrm{cd}}$ is the diffusion flux resulting from the cross derivative terms and calculated from

$$\mathbf{J}_f^{\mathrm{cd}} = -(\mu\nabla\mathbf{v}^{\mathrm{T}})_f\cdot\mathbf{A}_f = -\begin{bmatrix} \mu\frac{\partial u}{\partial x} & \mu\frac{\partial v}{\partial x} & \mu\frac{\partial w}{\partial x} \\ \mu\frac{\partial u}{\partial y} & \mu\frac{\partial v}{\partial y} & \mu\frac{\partial w}{\partial y} \\ \mu\frac{\partial u}{\partial z} & \mu\frac{\partial v}{\partial z} & \mu\frac{\partial w}{\partial z} \end{bmatrix}_f \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_f = -\begin{bmatrix} \mu\nabla_x u & \mu\nabla_x v & \mu\nabla_x w \\ \mu\nabla_y u & \mu\nabla_y v & \mu\nabla_y w \\ \mu\nabla_z u & \mu\nabla_z v & \mu\nabla_z w \end{bmatrix}_f \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix}_f \tag{32}$$

and ∇_x, ∇_y, ∇_z are the x-, y-, z-components of the gradient vector. The gradients of velocities can be calculated using either the Gauss or the least square method. Equation (31) can be written in the form

$$\frac{1}{\alpha}\mathbf{a}_P\mathbf{v}_P + \sum_{f=1}^{n_f(P)} \mathbf{a}_N\mathbf{v}_N = \mathbf{S} \tag{33}$$

where the coefficients $\mathbf{a}_P$ and $\mathbf{a}_N$ have forms analogous to Eq. (10), with Γ replaced by μ. If a collocated grid system is used, where all variables are stored at the cell centroids, then these coefficients are same

for all three velocity components, provided that Γ is the same for x, y and z-directions for each cell, including the near-boundary cells. As a result, memory requirements are less compared with the staggered grid systems. However, the formulation is given for the general case of Γ being direction dependent, which is the case for flow in anisotropic media.

The source term $\mathbf{S}$ is similar to that given in the equation (10), but takes the form of a vector:

$$\mathbf{S} = (a_P^t \mathbf{v}_P)^{l-1} + \frac{1-\alpha}{\alpha}\mathbf{a}_P \mathbf{v}_P^{n-1} + \mathbf{f}_P V_P + \sum_{f=1}^{n_f(P)} \mathbf{S}_f \tag{34}$$

where

$$\begin{aligned} \mathbf{S}_f &= \mathbf{S}_f^{\text{conv}} + \mathbf{S}_f^{\text{dif}} + \mathbf{S}_f^{\text{cd}} + \mathbf{S}_f^{\text{pres}}, \\ \mathbf{S}_f^{\text{conv}} &= -\left[\dot{m}_f(\mathbf{v}_f^H - \mathbf{v}_f^U)\right]^{n-1}, \\ \mathbf{S}_f^{\text{dif}} &= \left[\Gamma_f \overline{\nabla \mathbf{v}_f} \cdot \mathbf{A}_t + \Gamma_f \frac{(\nabla \mathbf{v}_N - \nabla \mathbf{v}_P)\cdot \mathbf{f_1 f}}{d_{PN}} A_d\right]^{n-1}, \\ \mathbf{S}_f^{\text{cd}} &= -\mathbf{J}_f^{\text{cd}}, \quad \mathbf{S}_f^{\text{pres}} = -p_f \mathbf{A}_f \mathbf{I} \end{aligned} \tag{35}$$

3.1. Calculation of the mass flow rate

The coefficients a_P and a_N in Eq. (10) require the calculation of the mass flow rate, $\dot{m}_f$, across the cell faces, which in turn, require velocities at cell faces. However, for a collocated arrangement of variables, the velocities are known at the centroid of a cell. The momentum equations (33) can be rearranged as:

$$\frac{1}{\alpha}\mathbf{a}_P \mathbf{v}_P = \mathbf{H}_P - V_P \nabla p_P + \frac{(1-\alpha)}{\alpha}\mathbf{a}_P \mathbf{v}_P^{n-1} + (a_P^t)^{l-1}\mathbf{v}_P^{l-1} + \mathbf{f}_P V_P \tag{36}$$

where

$$\mathbf{H}_P = -\sum_{f=1}^{n_f(P)} \mathbf{a}_N \mathbf{v}_N + \sum_{f=1}^{n_f(P)} (\mathbf{S}_f^{\text{conv}} + \mathbf{S}_f^{\text{dif}} + \mathbf{S}_f^{\text{cd}}) \tag{37}$$

Note that the source terms resulting from pressure, relaxation, force, and transient terms have not been included in the $\mathbf{H}$ term. It is a pseudoflux vector at node P which can also be expressed as:

$$\mathbf{H}_P = \frac{1}{\alpha}\mathbf{a}_P \mathbf{v}_P + V_P \nabla p_P - \frac{(1-\alpha)}{\alpha}\mathbf{a}_P \mathbf{v}_P^{n-1} - (a_P^t)^{l-1}\mathbf{v}_P^{l-1} - \mathbf{f}_P V_P \tag{38}$$

Using linearly interpolated values for the velocities at the cell faces results in a "checkerboard" pressure field. This problem can be avoided using the MIM originally proposed by Rhie and Chow [15]. However, in the original MIM method, the obtained solution depends both on the relaxation parameter α and the time step size Δt. To remove these effects, we mimick Eq. (36) to find the face velocities in the following way:

$$\mathbf{v}_f = \alpha \mathbf{a}_f^{-1}\mathbf{H}_f - \mathbf{D}_f(\nabla p)_f + (1-\alpha)\mathbf{v}_f^{n-1} + \alpha \mathbf{a}_f^{-1}(a_P^t)_f^{l-1}\mathbf{v}_f^{l-1} + \mathbf{D}_f \mathbf{f}_f \tag{39}$$

where the face values are linearly interpolated between the adjacent cell centroids as:

$$\begin{aligned} &\mathbf{a}_f = (1-\text{fr}_P)\mathbf{a}_P + \text{fr}_P \mathbf{a}_N, \quad \mathbf{H}_f = (1-\text{fr}_P)\mathbf{H}_P + \text{fr}_P \mathbf{H}_N, \\ &\mathbf{D}_f = \alpha \mathbf{a}_f^{-1} V_f, \quad V_f = (1-\text{fr}_P)V_P + \text{fr}_P V_N, \quad \mathbf{f}_f = (1-\text{fr}_P)\mathbf{f}_P + \text{fr}_P \mathbf{f}_N \end{aligned} \tag{40}$$

and the interpolation factor fr_P is (Figure 1)

$$\mathrm{fr}_P = \frac{\mathbf{Pf} \cdot \mathbf{e}_{PN}}{(\mathbf{PN} \cdot \mathbf{PN})^{1/2}} \tag{41}$$

Equation (39) can also be expressed in terms of a pseudovelocity $\hat{\mathbf{v}}_f$ as

$$\mathbf{v}_f = \hat{\mathbf{v}}_f - \mathbf{D}_f(\nabla p)_f \tag{42}$$

where

$$\hat{\mathbf{v}}_f = \alpha \mathbf{a}_f^{-1}\mathbf{H}_f + (1-\alpha)\mathbf{v}_f^{n-1} + \alpha \mathbf{a}_f^{-1}(a_P^t)_f^{l-1}\mathbf{v}_f^{l-1} + \mathbf{D}_f\mathbf{f}_f \tag{43}$$

This procedure of finding the face velocity is equivalent to the MIM method proposed for structured Cartesian grids in Yu et al. [13]. However, storing volume flow rates through cell faces is more economical than storing the three components of the velocity vector at faces. Volume flow rate, $\dot{V}_f$, at a cell face f is found by inserting $\mathbf{v}_f$ given by Eq. (39) into, $\dot{V}_f = \mathbf{v}_f \cdot \mathbf{A}_f$. The result is

$$\dot{V}_f = \hat{V}_f - (\mathbf{D}_f \nabla p_f) \cdot \mathbf{A}_f \tag{44}$$

where

$$\hat{V}_f = \alpha(\mathbf{a}_f^{-1}\mathbf{H}_f) \cdot \mathbf{A}_f + (1-\alpha)\dot{V}_f^{n-1} + \frac{\rho_f^{l-1}}{\Delta t}\frac{(\mathbf{D}_f\mathbf{A}_f)\cdot\mathbf{A}_f}{\mathbf{A}_f \cdot \mathbf{A}_f}\dot{V}_f^{l-1} + (\mathbf{D}_f\mathbf{f}_f)\cdot\mathbf{A}_f \tag{45}$$

To discretize the term $(\mathbf{D}_f\nabla p_f) \cdot \mathbf{A}_f$ in Eq. (44), it is first written in the form $\nabla p_f \cdot \mathbf{A}_f''$ where $\mathbf{A}_f'' = \mathbf{D}_f\mathbf{A}_f$ and then discretized similar to the discretization of the diffusion term in Eq. (7), by replacing $\mathbf{A}_f$ with $\mathbf{A}_f''$. The volume flow rate across a face becomes

$$\dot{V}_f = \hat{V}_f - \frac{\mathbf{A}_f'' \cdot \mathbf{A}_f''}{\mathbf{A}_f'' \cdot \mathbf{PN}}[(p_N - p_P) + (\nabla p_N - \nabla p_P)\cdot \mathbf{f_1 f}] - (\overline{\nabla p})_f \cdot \mathbf{A}_t'' \tag{46}$$

where $\mathbf{A}_t''$ is found from Eq. (8) by replacing $\mathbf{A}_f$ with $\mathbf{A}_f''$ and consequently, mass flow rate at cell faces, $\dot{m}_f$, is calculated from $\dot{m}_f = \rho_f \dot{V}_f$

4. Pressure correction equation

In SIMPLE algorithm, the solution is obtained by iteratively solving the momentum equations and a pressure correction equation, derived from the continuity equation, which is written in terms of corrections of the velocity field [1]. The velocity field obtained from the solution of the momentum equations does not satisfy the continuity equation and require correction. In SIMPLE method, the discretized continuity equation is written as:

$$\frac{(\rho_P - \rho_P^o)}{\Delta t}V_P + \sum_{f=\mathrm{nb}(P)} \dot{m}_f^* + \sum_{f=\mathrm{nb}(P)} \dot{m}_f' = 0 \tag{47}$$

where $\dot{m}_f'$ is the mass correction approximated by:

$$\dot{m}_f' = -\rho_f(\mathbf{D}_f\nabla p_f')\cdot\mathbf{A}_f = -\frac{\rho_f\mathbf{A}_f'\cdot\mathbf{A}_f'}{\mathbf{A}_f'\cdot\mathbf{PN}}(p_N' - p_P') - (S_f^{\mathrm{dif}})^{p'} \tag{48}$$

and

$$(S_f^{\mathrm{dif}})^{p'} = \frac{\rho_f \mathbf{A}_f'' \cdot \mathbf{A}_f''}{\mathbf{A}_f'' \cdot \mathbf{PN}} (\nabla p_N' - \nabla p_P') \cdot \mathbf{f_1 f} + \rho_f \left(\overline{\nabla p'}\right)_f \cdot \mathbf{A}_t'' \tag{49}$$

Substituting Eq. (48) into the continuity equation (47) gives the pressure correction equation expressed as:

$$a_P^{p'} p_P' + \sum_{N=\mathrm{nb}(P)} a_N^{p'} p_N' = S^{p'} \tag{50}$$

where

$$\begin{aligned} a_N^{p'} &= -\frac{\rho_f (\mathbf{A}_f'') \cdot \mathbf{A}_f''}{\mathbf{A}_f'' \cdot \mathbf{PN}}, \quad a_P^{p'} = -\sum_{N=\mathrm{nb}(P)} a_N^{p'} \\ S^{p'} &= -\frac{(\rho_P - \rho_P^o)}{\Delta t} V_P - \sum_{f=1}^{n_f(P)} \dot{m}_f^* + \sum_{f=1}^{n_f(P)} (S_f^{\mathrm{dif}})^{p'} \end{aligned} \tag{51}$$

The term $(S_f^{\mathrm{dif}})^{p'}$ can be neglected since an exact solution of Eq. (50) is not required. However, for severely distorted grids, it can be incorporated into the calculations as explained in Ferziger and Peric [32]. Mass flow rate through faces, $\dot{m}_f^*$, is calculated by multiplying Eq. (44) with density. Once pressure corrections are obtained through the solution of pressure correction equation (50) the velocities, pressures and mass flow rates are corrected using

$$\mathbf{v}_P = \mathbf{v}_P^* + \mathbf{v}_P' = \mathbf{v}_P^* - \mathbf{D}_P \nabla p_P' \tag{52}$$

$$p_P = p_P^* + \alpha_p p_P' \tag{53}$$

$$\dot{m}_f = \dot{m}_f^* + \dot{m}_f' \tag{54}$$

where α_p is a relaxation parameter between 0 and 1 and $\dot{m}_f'$ is given by Eq. (48).

4.1. Types of boundary conditions for the pressure correction equation

In general, there are two types of boundary conditions that are applied to the pressure correction equation (50): (a) specified pressure at the boundary and (b) specified velocity at the boundary.

4.1.1. Specified pressure at the boundary (Dirichlet type for p')

Sometimes, the pressure at an inlet and/or outlet is specified. In such cases, the pressure correction at the boundary, p_b', is zero. Then, for a near-boundary cell, Dirichlet boundary condition, similar to Eq. (18), is used by setting $\phi_b = 0$. For a near-boundary cell, the coefficients are given by:

$$\begin{aligned} S^{p'} &= \tilde{S}^{p'} - \dot{m}_b \\ a_P^{p'} &= \tilde{a}_P^{p'} - (a_N^{p'})_b \\ a_N^{p'} &= 0 \end{aligned} \tag{55}$$

where the source term $(S_b^{\mathrm{dif}})^{p'}$ has been neglected. When the pressure at a boundary is specified, the velocities and mass flow rates are unknown and need to be corrected. For example, for cases where the flow is fully developed, such as at the exit of a channel, the boundary pressure is uniform and can be set to a reference value. In that case, the boundary velocities are corrected using Eq. (52). Mass flow rate at a boundary is calculated from

$$\dot{m}_b = \rho_b \mathbf{v}_b \cdot \mathbf{A}_b \tag{56}$$

4.1.2. Specified velocity at the boundary (Neumann type for p′)

When the velocities at a boundary are known, there is no need to correct the velocities at the boundary for the derivation of the pressure correction equation. For example, at solid walls, all velocity components are zero. For inlets, sometimes the inlet velocities are specified rather than the pressure. For cases where normal velocities at the boundary are known, $\mathbf{v}'_b = 0$ and hence, $\dot{m}'_b = 0$. Then, Eq. (50) for a near-boundary cell can be obtained by replacing the diffusion flux, $\rho_f(\mathbf{D}_f \nabla p'_f) \cdot \mathbf{A}_f$ at the boundary face, in Eq. (48) with $\dot{m}'_f$, which is zero when boundary normal velocities are specified. Then, the coefficients for a near-boundary cell become

$$\begin{aligned} a_P^{p'} &= \tilde{a}_P^{p'} \\ S^{p'} &= \tilde{S}^{p'} - \dot{m}_b \\ a_N^{p'} &= 0 \end{aligned} \tag{57}$$

Since normal velocity at the boundary face is specified, mass flow rate, $\dot{m}_f$, through a boundary face in Eq. (57) is calculated directly from Eq. (56). When velocities at an inflow or outflow boundary are specified, the pressure is unknown and needs to be determined at the end of the correction stage. The boundary value of the pressure correction should also be found for the purpose of calculating gradient of pressure correction at a near-boundary control volume which is required for the correction of pressure in accordance with Eq. (53).

For boundaries where the normal velocity is specified, $\dot{m}'_b = 0$, and hence from Eq. (48) $(\nabla p' \cdot \mathbf{n})_b = 0$. Then, one might be tempted to use Eq. (21) with $g_n = 0$ for finding p'_b. However, $\nabla p'_P$ is not available since it involves p'_b and its previous iteration value cannot be used, as the two iteration values are completely different. In this case, the reconstructed value of p'_b can be used which is calculated from Eq. (26). Then, the boundary pressure is calculated by correcting the pressure in accordance with Eq. (53).

5. Pressure equation

In some methods of solving the momentum equations, the continuity equation is expressed in terms of pressure so that a pressure equation has to be solved instead of a pressure correction equation [33]. Also in some solution methods such as SIMPLER, [3] a pressure equation is solved in addition to the pressure correction equation. The derivation of the pressure equation is similar to that of pressure correction equation. However, in this case, no approximation is used for the reconstructed volume flow rate equation (44) at the cell faces so that the mass flow rate is

$$\dot{m}_f = \hat{m}_f - \rho_f(\mathbf{D}_f \nabla p_f) \cdot \mathbf{A}_f \tag{58}$$

where $\hat{m}_f = \rho \hat{V}_f$. Inserting $\dot{m}_f$ into the continuity equation yields

$$\sum_{f=\mathrm{nb}(P)} -\rho_f(\mathbf{D}_f \nabla p_f) \cdot \mathbf{A}_f = -\frac{(\rho_P - \rho_P^o)}{\Delta t} V_P - \sum_{f=\mathrm{nb}(P)} \hat{m}_f \tag{59}$$

Discretizing $(\mathbf{D}_f \nabla p_f) \cdot \mathbf{A}_f$ similar to the discretization of the diffusion term given by Eq. (7), equation (59) can be written as:

$$a_P^p p_P + \sum_{N=\mathrm{nb}(P)} a_N^p p_N = S^p \tag{60}$$

where

$$
\begin{aligned}
a_N^p &= -\frac{\rho_f \mathbf{A}_f'' \cdot \mathbf{A}_f''}{\mathbf{A}_f'' \cdot \mathbf{PN}}, \quad a_P^p = -\sum_{N=\mathrm{nb}(P)} a_N^p \\
S^p &= -\frac{(\rho_P - \rho_P^o)}{\Delta t} V_P - \sum_{f=1}^{n_f(P)} \hat{m}_f + \sum_{f=1}^{n_f(P)} (S_f^{\mathrm{dif}})^p \\
(S_f^{\mathrm{dif}})^p &= \frac{\rho_f \mathbf{A}_f'' \cdot \mathbf{A}_f''}{\mathbf{A}_f'' \cdot \mathbf{PN}} (\nabla p_N - \nabla p_P) \cdot \mathbf{f_1 f} + \rho_f \left(\overline{\nabla p}\right)_f \cdot \mathbf{A}_t''
\end{aligned}
\tag{61}
$$

In general, there are two types of boundary conditions that are applied to the pressure equation (60); (a) specified pressure at the boundary and (b) specified velocity at the boundary.

5.1. Types of boundary conditions used for the pressure equation

5.1.1. Specified pressure at the boundary (Dirichlet type for p)

When the pressure at the boundary is specified, Dirichlet boundary condition is used. For a near-boundary cell, the coefficients are similar to that of Eq. (18)

$$
\begin{aligned}
S^p &= \tilde{S}^p - \hat{m}_b + (S_b^{\mathrm{dif}})^p - (a_N^p)_b p_b \\
a_P^p &= \tilde{a}_P^p - (a_N^p)_b \\
a_N^p &= 0
\end{aligned}
\tag{62}
$$

The value of pressure gradient term at the boundary $(\overline{\nabla p})_b$ in Eq. (61) depends on the flow condition. For flow across the boundary, a linear change around P can be assumed so that $(\overline{\nabla p})_b = (\overline{\nabla p})_P$ is used. When pressure at inlet or exit is fixed, the velocity and hence the mass flow rates are unknown. The inlet or exit velocities are calculated from Eq. (42) and the mass flow is calculated from Eq. (56).

5.1.2. Specified velocity at the boundary (Neumann type for p)

When the velocities at a boundary are known, for the derivation of the pressure equation, the mass flow rate at the boundary is not expressed in terms of pressure gradient but directly inserted into the continuity equation. Then, the coefficients and source term of the pressure equation for a near-boundary cell become

$$
\begin{aligned}
S^p &= \tilde{S}^p - \dot{m}_b \\
a_P^p &= \tilde{a}_P^p \\
a_N^p &= 0
\end{aligned}
\tag{63}
$$

The calculation of boundary pressure depends on whether there is mass flow or not across the boundary. From the momentum equation (29), for no mass flow across the boundary, $(\nabla p \cdot \mathbf{n})_b = 0$. Hence, Eq. (26) is used for finding the boundary pressure where **Pb** is replaced with **PP′**. For the case of open boundaries through which there is mass flow, the reconstructed value of p_b is used at the boundary in accordance with Eq. (26) or Eq. (28).

6. Frequently used boundary types for flow problems

6.1. Inlet boundary conditions

Two types of inlet boundary conditions are considered: (i) specified velocity and (ii) specified pressure.

6.1.1. Specified velocity at inlet

If the flow velocities at an inlet are known, then for momentum equations, the inlet boundary condition corresponds to specified value of ϕ (Dirichlet boundary condition) discussed earlier and is implemented using Eq. (18). In general, for inlet boundaries where the inlet velocities are known but inlet pressure is not specified, a Neumann boundary condition is used for the pressure correction and pressure equations which are implemented using Eq. (57) and Eq. (63), respectively. Since boundary velocity is specified, for solution methods using pressure correction equation, the boundary value of pressure correction is calculated from Eq. (26) and the boundary pressure is calculated from Eq. (53). For methods using only pressure equation, the boundary pressure is calculated from Eq. (26).

6.1.2. Specified pressure at inlet

Instead of velocities, sometimes pressure at the inlet is specified. In this case, a Dirichlet boundary condition is used for the pressure correction and pressure equations, which is implemented using Eqs. (55) and (62), respectively. The velocities and hence the mass flow rates at the boundary are unknown. For the solution methods which use a pressure correction equation, the boundary velocities are corrected using Eq. (52). Mass flow rates at the boundary are found from Eq. (56). For the solution methods which do not use a pressure correction equation, boundary velocities are corrected using Eq. (42) and mass flow rates are calculated from Eq. (56). A zero gradient boundary condition is applied for the momentum equations which is implemented using Eq. (23). For the scalar equations, the value of the variable should be specified at the inlet.

6.2. Outlet boundary conditions

Two different outlet boundary conditions are considered: (i) fully developed flow at outlet, and (ii) undeveloped flow at outlet.

6.2.1. Fully developed flow at outlet (specified pressure outlet)

The flow domain is usually selected such that the flow is fully developed at outlet, where the normal gradient of all velocity components are zero. Thus, for the solution of momentum equations, a zero gradient ($g_n = 0$) Neumann boundary condition is used, which is implemented using Eq. (23) and the exit velocity is obtained from Eq. (21). A rather uniform pressure is expected at the exit plane of the developed flow so that the exit pressure is set to a reference value. As a result, a Dirichlet boundary condition is used for the pressure correction and pressure equations, which is implemented using Eqs. (55) and (62), respectively. At the correction stage, the calculation of the exit velocities and mass flow rates are similar to that explained in section "specified pressure at inlet."

6.2.2. Undeveloped flow at outlet (open boundary)

For the case of an outflow boundary, where the flow is not fully developed, neither pressure nor velocities are known. This situation arises at truncated domains with synthetic boundaries which is often used to reduce the size of the computation domain and hence, the computer time. This type of boundary was proposed by Papanastasiou et al. [34] and is named as free or open boundary condition. In this case, both pressure and velocities at the outlet boundary are generated as a part of the solution process. A pseudo-Dirichlet boundary condition is applied for the momentum equations where the velocities calculated from the previous iteration are considered to be known, which is implemented using Eq. (18). The name "pseudo-Dirichlet" is used here to distinguish the usual Dirichlet boundary condition where the value of the variable is fixed, from the case where the boundary value of the variable is not fixed but changes at each iteration. However, there is no guarantee that these velocities will conserve mass over the

computational domain. To ensure global mass conservation, the mass flow rates at the boundary are corrected by:

$$\dot{m}_b = \overline{\dot{m}_b} - \frac{A_b}{A_{\text{out}}} \Delta M_\Omega \tag{64}$$

where $\Delta M_\Omega = M_{\text{in}} - M_{\text{out}}$ is net mass imbalance for the computational domain, M_{in} is the total mass flow rate into the domain, which is negative, M_{out} is the total mass flow rate going out of the domain, which is positive, A_b is the boundary face area of the cell, A_{out} is the total exit area, and $\overline{\dot{m}_b}$ is the mass flow rate at the outflow boundary face of a cell before modification. It should be pointed out that such global mass conservation is necessary only when "pseudo-Dirichlet" boundary condition is used, while mass conservation is automatically satisfied for all other boundary conditions discussed in this study.

Using the boundary velocities calculated in this way, Neumann boundary condition is used for the pressure correction and pressure equations, which is implemented using Eqs. (57) and (63), respectively. The boundary pressure is calculated from Eq. (26) or Eq. (28) assuming a linear variation at the exit. The boundary velocities are also reconstructed using Eq. (26) or Eq. (28) at the correction stage and are used as input for solving the momentum equations at the next iteration.

6.3. Inlet–outlet boundary condition (specified pressure inlet–outlet)

In some cases, there may be inflow through some parts of an open boundary and outflow through the other parts; however, the exact location of the inflow and outflow regions is not known. In this case, the open boundary is treated as a fully developed outflow boundary where the pressure is held constant and is implemented as discussed in the previous section. However, in parts of the boundary where mass flows into the domain ($\dot{m} < 0$), the boundary values of scalar variables, such as temperature, are assigned with the values of the surroundings while zero normal gradient is applied for the velocity components.

6.4. Wall boundary condition

For walls, a Dirichlet boundary condition given by Eq. (18) is applied. Boundary conditions for the pressure correction and pressure equations are implemented using Eqs. (57) and (63), respectively. For the solution methods using the pressure equation only, where the pressure correction equation is missing, boundary pressure is calculated using Eq. (26). For the solution methods which solve a pressure correction equation, the boundary pressure and pressure correction values are calculated from Eqs. (53) and (26), respectively, by replacing $\mathbf{Pb}$ with $\mathbf{PP}'$ for the latter case.

6.5. Symmetry boundary condition

In many steady flows, there is at least one symmetry plane, where the flow is symmetric with respect to this plane. In these cases, the flow problem can be solved only in part of the domain using symmetry conditions at the symmetry plane. At the symmetry plane, (a) the convective fluxes of all quantities are zero, or $\mathbf{v}_n = 0$, where n is normal to symmetry plane, (b) normal gradients of the velocity components parallel to symmetry plane are zero; that is, shear stresses on the symmetry plane are zero and (c) normal gradients of the scalar quantities are zero. Requirement (a) can be implemented by putting the mass flow rate at the symmetry boundary as zero, or $\dot{m}_b = 0$. Requirements (b) and (c) are implemented by setting g_{n} in Eq. (23) to zero for the related equations.

For the pressure correction and pressure equations, Neumann-type boundary condition is used at the symmetry boundary due to specified velocity (zero normal velocity) which is implemented using Eqs. (57) and (63), respectively. The boundary pressure and pressure correction values are calculated from Eq. (26) by replacing $\mathbf{Pb}$ with $\mathbf{PP}'$ for the latter case.

6.6. Specified heat flux at a wall boundary

If the heat transfer rate at a boundary cell is specified as q_b (W/m^2), then the boundary condition of the energy equation corresponds to that of a specified gradient case and is implemented using Eq. (23) where the gradient term is given by $g_n = -q_b/k_b$. The term k_b is the thermal conductivity of the fluid at the boundary. For an adiabatic boundary condition in energy equation, $g_n = 0$. Note that heat added to the control volume is negative while heat going out of the control volume is positive in accordance with the definition of diffusion flux J_b^{dif} given in Eq. (13). The boundary temperature is calculated from Eq. (21).

7. Results and discussion

The formulation of implementation of the boundary conditions is assessed for the following laminar incompressible flow problems: (1) flow over a backward-facing step, (2) cross-flow over a circular cylinder, (3) natural convection around a horizontal cylinder (4) natural convection in a square open cavity. The residual of the momentum and scalar equations is defined by:

$$\text{Res}_\phi = \frac{\sum_{\text{cells}} \sum_{N=\text{nb}(P)} |(a_N \phi_N + a_P \phi_P - S)|}{\sum_{\text{cells}} |a_P \phi_P|} \tag{65}$$

The computations are stopped when the residual of all equations in the computational domain is less than 10^{-6}.

The program code was developed at Eastern Mediterranean University by the author and can run on 2D or 3D unstructured grids in unsteady or steady mode and can be operated either in single or parallel mode. It is a pressure-based, finite volume program which can be switched between SIMPLE, SIMPLEC, and SIMPLER methods where the momentum and pressure or pressure correction equations are solved sequentially. Conjugate gradient method is used for the solution of the pressure or pressure correction equations using incomplete LU factorization as the preconditioner for the coefficient matrix. The open-source grid generation package GMHS [35] was used for generating the grids. The implementation of boundary conditions are tested for different solution methods mentioned above. However, the results are presented using the SIMPLEC method using a single processor.

7.1. Flow over a backward-facing step

The first problem is flow over a backward-facing step, which is a standard test case used by researchers to test the validity of their proposed methodologies of implementing the outflow boundary conditions when the flow at the outlet is not fully developed. This condition results when a truncated domain is used for reducing the computer time and memory requirements. In this case, there may be both outflow and inflow at the open boundary due to recirculation. The flow enters the channel *via* the entrance at the upper part of the left side for $y/H \geq 0.5$ with a parabolic profile given by $u(y) = 24\, y\, (0.5 - 4y)\, u_{\text{mean}}$, where H is the height of the channel where the origin is located at the bottom of the inlet. The ratio of the exit to inlet heights, H/h, is 2. Reynolds number is 800, which is based on u_{mean} and H. For a sufficiently long channel, i.e., $L/H > 30$, the flow at the outlet can be considered to be fully developed [19] where L is the length of the channel. For a shorter domain the flow is still developing and accurate results cannot be obtained using fully developed boundary conditions at the outlet. If the domain is truncated severely, such that $L/H < 9.5$, then, the outlet plane corresponds to a section where recirculation exists, resulting in inflow at some part of the outlet.

Simulations were made for both a long channel of $L = 30H$ and a truncated domain of $L = 8H$. For the latter case, the outlet plane is located at a section which corresponds to the recirculation region near the upper wall. For the long channel, a 301×61 uniform quadrilateral mesh is used,

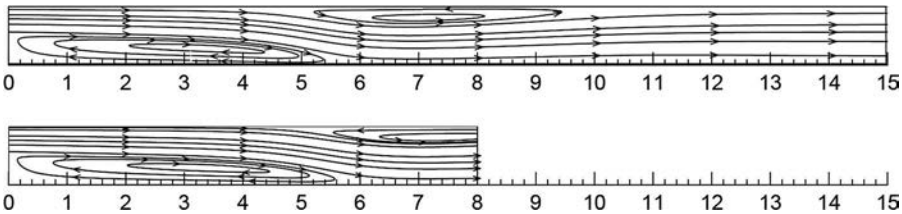

Figure 2. Streamlines for the long ($L = 30H$) and truncated ($L = 8H$) channels.

while for the truncated channel, the domain and grid are cut so that the outlet is located at $L = 8H$ with 81×61 uniform grid system, having the same grid distribution of the long channel. The wall boundary conditions are used on solid walls while specified velocity at inlet boundary condition is used at the upper left boundary for both cases. However, fully developed and undeveloped flows at outlet boundary condition are used for the long and the short channel cases, respectively, on the right section.

Figure 2 shows the streamlines in the channel. It is observed that the qualitative distributions are almost the same irrespective of the position of the outlet plane. The outlet section at $x = 8H$ intersects the upper vortex. However, this truncated domain does not deteriorate the results. The positions of separation and the reattachment point on the lower upper walls are almost the same for both cases. It should be noted that only the portion up to $x = 15H$ is shown in the figure since no phenomena of interest exists after that section.

Figure 3 shows the pressure distribution on the bottom wall if the outlet is located at different places and compares them with that of the benchmark results of Gartling [19]. It can be observed that the present results agree remarkably well with that of Gartling, which were obtained for a channel length of $30H$. It is also observed that when the channel is truncated to $L = 8H$, the pressures agree well with that of the full domain ($L = 30H$) which ensures a fully developed flow. This shows that the current formulation of the "undeveloped flow at outlet" boundary condition correctly solves the problem when the domain is truncated to $L = 8H$. The close agreement of the pressure values obtained by applying the undeveloped flow at outlet boundary condition is despite of the rather coarse mesh near the open boundary. As indicated in Mitsoulis and Malamataris [21], the accuracy of pressures are heavily dependent on mesh distribution in the axial direction near the outlet for truncated domains and better results can be obtained with meshes clustered near the outlet.

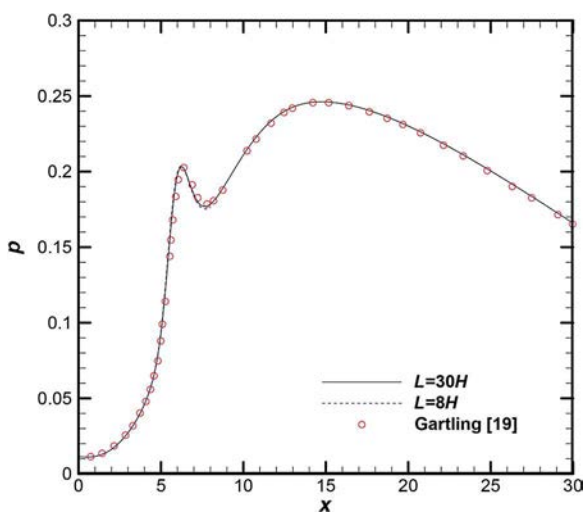

Figure 3. Pressure distribution on the bottom wall for the long and truncated channels.

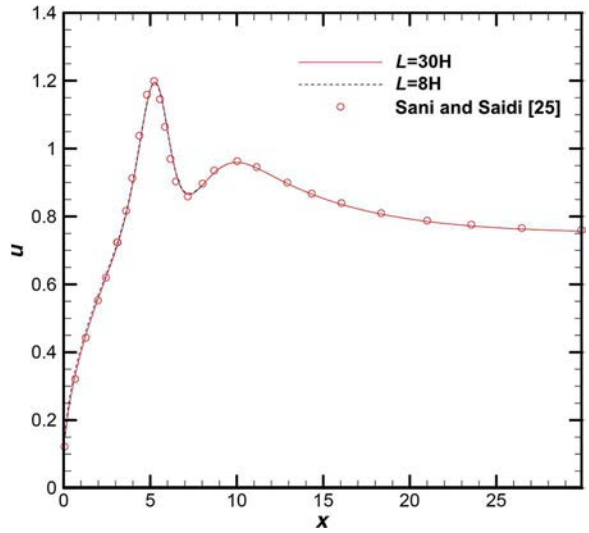

Figure 4. Axial velocity along the center line for the full and truncated domains.

A comparison of the u-velocity profiles for the full and truncated domains along the $y = 0$ axis is given in Figure 4, together with the benchmark results of Sani and Saidi [25]. It can be observed that the velocity profiles of the full and truncated domains are almost indistinguishable and they are also in good agreement with the results of Sani and Saidi [25]. Again considering the complex behavior of the flow at $x = 8H$, the close agreement of the velocity profiles of the full and truncated domains proves the success of the present formulation of outflow boundary conditions for undeveloped flows.

Figure 5 shows the u-velocity profile on the cross section at $x = 7H$. This plane is selected because the benchmark results are available in tabular form at this section in Gartling [19] and tested in Mitsoulis and Malamataris [21], Darbandi and Vakilipour [23], and Sani and Saidi [25]. Again, the results of the truncated domain agree well with that of the full domain and the benchmark values.

It should be pointed out that all results of the backward-facing step problem were obtained using the Gauss method for data reconstruction [Eq. (26)] in finding the values of velocity and pressures at

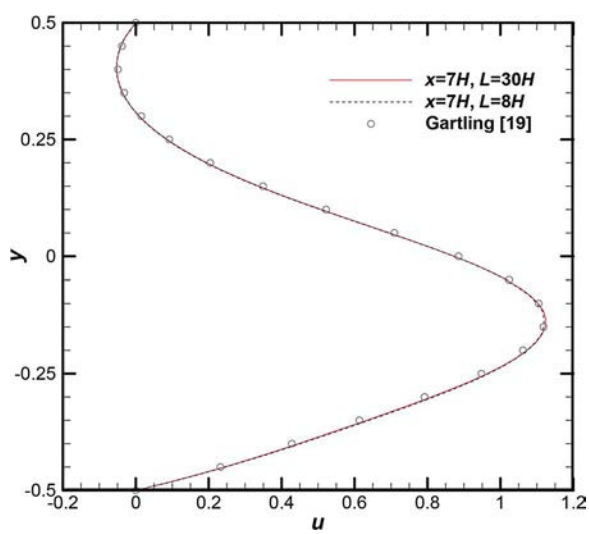

Figure 5. Axial velocity profile at $x = 7H$ for the truncated and full domains.

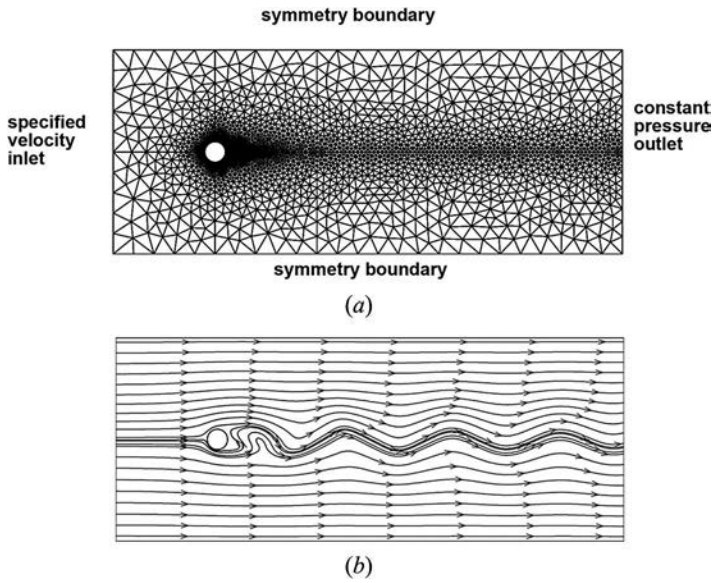

Figure 6. (*a*) Grid layout and boundary conditions for flow over a cylinder, (*b*) streamlines at $t = 250$.

the open boundary. Using the least-square method for data reconstruction given by Eq. (28) at the open boundary had negligible effect on the results.

7.2. Flow over a cylinder

The second problem is 2D flow over a circular cylinder in free stream at a Reynolds number, $Re = 100$, where $Re = \rho u_\infty D/\mu$, u_∞ is the free stream velocity and D is the diameter of the cylinder. The grid layout is shown in Figure 6(*a*) with a total of 6,208 triangular cells. The grids are clustered along the wake to resolve the vortex street and near the cylinder with 160 cells located on its surface. The lower and upper boundaries were located 10 diameters away from the cylinder center, on which symmetry boundary condition is imposed. The left boundary is located 5 diameters away from the center of the cylinder where air enters with a uniform free stream velocity of $u_\infty = 1$ m/s. On this boundary, a specified velocity at inlet boundary condition is used. The outlet boundary is 20 diameters away from the cylinder center where a fully developed flow at outlet boundary condition is applied. The cylinder diameter, inlet velocity and density were all set to unity and the dynamic viscosity was set to 0.01.

An unsteady flow simulation was performed using the Crank–Nicolson scheme with a time step, $\Delta t = 0.02$. After an initial oscillatory state, the flow attains a periodic state beyond $t = 100$. The vortex street formed during the vortex shedding period is illustrated in Figure 6(*b*) in terms of the streamlines at $t = 250$. Table 1 shows the time averaged drag coefficient and Strouhal number with the published results [7, 36–38]. The results given in the table belong to structured or block-structured grids except that of Dalal et al. [7] which use triangular grids. Different domain sizes

Table 1. Comparison of Strouhal number, *St* and the mean drag coefficient, C_D.

	St	C_D
Present	0.1770	1.468
Dalal et al. [7]	0.1587	1.4147
Lange et al. [36]	0.1656	1.327
Zhang et al. [37]	0.173	1.425
Lilek et al. [38]	0.1672	1.3582

and number of grids were used in these studies with the maximum number of grids being 20,736 which belongs to Zhang et al. [37]. Despite the rather small number of cells used in the present study, the comparison with the published results is good.

7.3. Natural convection from a horizontal isothermal cylinder

The problem considered is buoyant flow around a two-dimensional isothermal cylinder suspended horizontally in an infinite fluid. The physical problem and the grid layout are shown in Figure 7 (*a*). Only half of the cylinder is modeled due to symmetry of the problem and triangular grids are generated around the half-cylinder which are concentrated near the surface. A total of 6,208 cells is used with 80 elements located on the surface of the half-cylinder. The hypothetical outer boundary is located 5 diameters away from the cylinder center. The variation of density is evaluated using the Boussinesq approximation which gives the buoyancy force in Eq. (30) as:

$$\mathbf{f} = -\rho_\infty \mathbf{g}\beta(T - T_\infty) \tag{66}$$

where β is the coefficient of thermal expansion and $\mathbf{g}$ is the gravitational acceleration in vector form. The numerical calculations are performed in the range of Rayleigh numbers 10^3–10^6, where the Rayleigh number is defined by:

$$Ra = \frac{\rho g \beta D^3 (T_w - T_\infty)}{\mu\alpha} \tag{67}$$

for a fluid having a Prandtl number, $Pr = c_p\mu/k$, of 0.71. The pressure at the hypothetical outer boundary is set to zero where the flow enters mostly from below and leaves out of the domain from top. However, the exact position of the demarcation line between the inflow and outflow boundary is

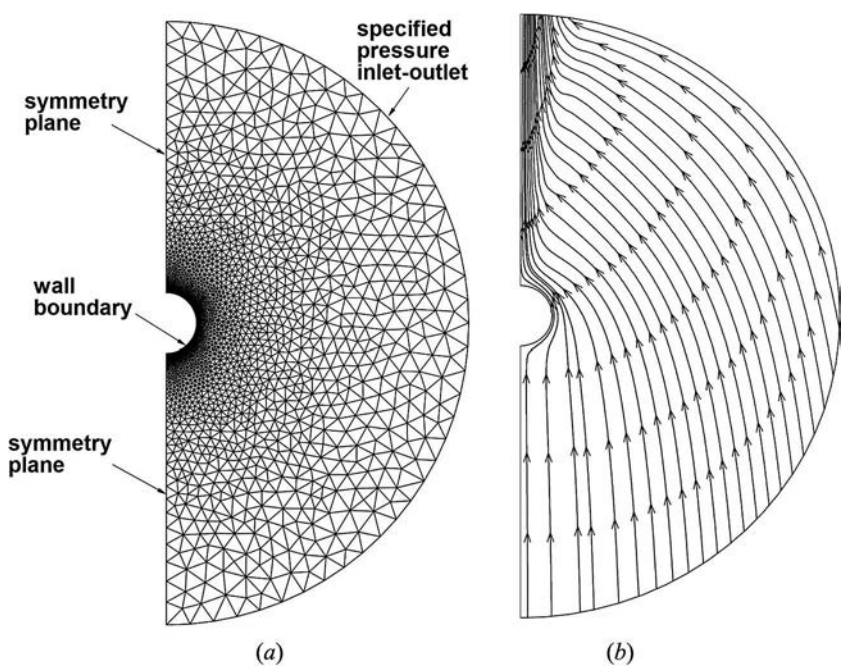

Figure 7. (*a*) The grid layout and the boundary conditions for natural convection over the horizontal cylinder, (*b*) streamlines for $Ra = 10^5$.

Table 2. Comparison of average Nusselt numbers for natural convection about a horizontal isothermal cylinder.

	Nu				
Ra	Present	Kuehn and Goldstein [40]	Wu and Tao [39]	Saitoh et al. [42]	Wang and Kahawita [41]
10^3	3.065	3.09	3.026	3.024	3.06
10^4	4.860	4.94	4.831	4.826	4.86
10^5	7.951	8.00	7.944	7.898	7.97
10^6	13.603	13.52			

not fixed in the present model; rather, it is automatically determined by implementing the specified pressure inlet–outlet boundary condition, where zero normal gradient condition is used for the momentum equations. The surface of the cylinder is considered as a wall boundary, which is held at a constant temperature. Symmetry boundary condition is applied at the symmetry plane on left side of the domain. The streamlines around the horizontal cylinder are shown in Figure 7(*b*) for $Ra = 10^5$. The flow is entrained into the domain from bottom and the side due to flow set in by the buoyancy force and leaves at the top through a very narrow space. The inflow direction is not fixed but comes out naturally as the iterations proceed which are not necessarily perpendicular to the domain boundary.

The comparison of the average Nusselt number calculated in the present work with those of Wu and Tao [39], Kuehn and Goldstein [40], Wang and Kahawita [41], and Saitoh et al. [42] is given in Table 2. The present results agree closely with the previous predictions.

7.4. Natural convection in a square open cavity

The final problem considered is laminar natural convection in a square open cavity having one heated vertical wall facing a vertical opening and two insulated horizontal walls. Two different domains are considered; (1) domain consisting of the cavity only and (2) extended domain that includes the far-field boundary having a radius of 2.5*L*, where *L* is the side length of the square cavity and a vertical wall at upper and lower parts of the opening. The grid layout of the extended domain is shown in Figure 8(*a*) which has a structured grid in the cavity and a triangular grid outside the cavity, which are connected together at the opening. There are 10,000 and 23,384 cells inside and outside of the cavity, respectively. The grids are clustered near the walls to capture the flow in boundary layers and coarsened at the far-field boundary.

The heated wall is kept at a constant temperature ($T_w = 1$), higher than the far-field temperature ($T_\infty = 0$), while all other walls are insulated. The imaginary far-field boundary is considered as a specified pressure inlet–outlet boundary and set to a reference pressure of zero. The simulations are performed for a fluid of $Pr = 1$, over the Rayleigh number range of 10^3–10^7 of which the definition is based on *L*. The simulations were also performed over the truncated domain, which includes only the cavity. In the latter case, the grid layout was exactly the same as that of the extended domain. A zero pressure inlet–outlet boundary condition was used at the opening for the truncated domain. For $Ra = 10^7$, a magnified view of the flow profile in the cavity is drawn in Figure 8(*b*) for the case of the extended domain, which shows the presence of a vortex near the bottom wall. Figure 8(*c*) shows the flow profile in the cavity when only the cavity is considered as the flow domain for the simulations. In the latter case, no vortex flow is formed near the bottom wall, showing that using only the cavity as the domain, cannot predict the flow profile correctly at $Ra = 10^7$. For lower Rayleigh numbers, the flow profiles predicted by the two domains were similar.

The average Nusselt number on the heated wall is compared with that of Chan and Tien [43], Mohamad [44], and Hinojosa et al. [45] in Table 3. It should be noted that a rather small extended domain was used in Chan and Tien [43], whereas domain was only the cavity in the other references. It is observed that the results of the present work agree closely with that of the previous researchers. It is also observed that higher *Nu* values are obtained when only the cavity is used as the domain, the

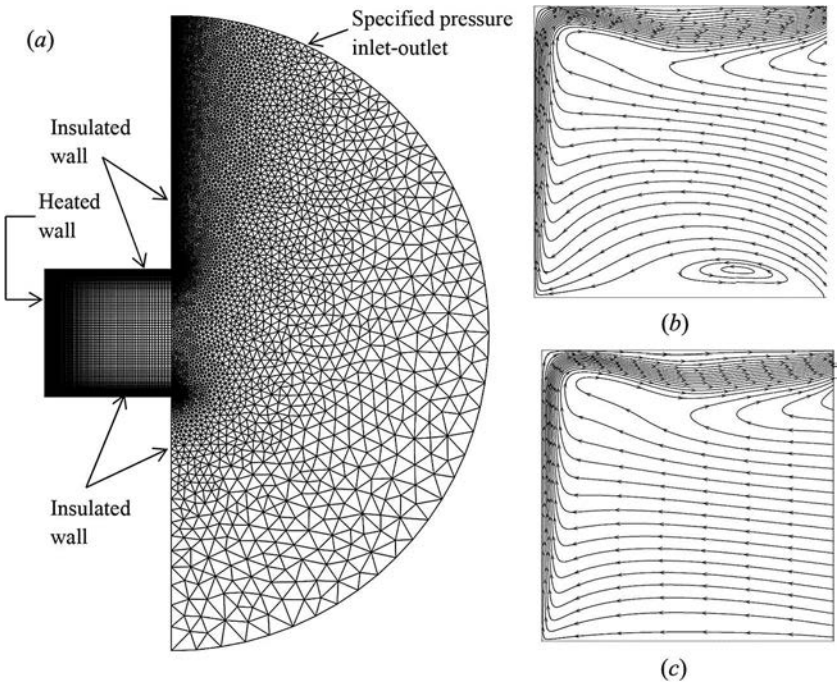

Figure 8. (*a*) Grid layout and the boundary conditions in the extended domain, (*b*) streamlines in the cavity for the extended domain, (*c*) streamlines in the cavity for the truncated domain.

Table 3. Comparison of the average Nusselt number for natural convection in a square open cavity.

	Nu				
Ra	Present (extended domain)	Present (cavity only)	Chan and Tien [43]	Mohamad [44]	Hinojosa et al. [45]
10^3	1.08	1.33	1.07	1.31	
10^4	3.36	3.54	3.41	3.44	3.57
10^5	7.65	7.71	7.69	7.41	7.75
10^6	14.95	15.05	15.0	14.36	15.11
10^7	27.91	27.99	28.60	28.60	28.70

difference reaching to a value of 23.1% which occurs at the lowest Rayleigh number (10^3) investigated. The rather high difference in Nusselt numbers between the cavity-only and the extended domain shows that the flow details outside the open boundary may be important in some problems. In this case, the upward cross-flow just outside the open boundary, which is induced by the buoyancy forces, changes the direction of the incoming and outgoing fluids which have not been accounted for by the boundary conditions of the cavity-only domain model.

8. Conclusion

The implementation of the boundary conditions is presented in terms of the contribution of the boundary face of a boundary cell to the discretized governing equation for the solution of flow problems on unstructured grids, using control volume method with colocated variable arrangement. The types of the boundary conditions considered are Dirichlet and Neumann, since the boundary conditions for the momentum and energy equations in the majority of the flow problems encountered in practice can be expressed in terms of these two basic types or their combinations.

However, there are situations in which the boundary conditions are unknown, such as the case at open boundaries formed as a result of truncation of long domains and the flow is undeveloped. In this

case, the data reconstruction method is proposed for finding the boundary values of the variables at the correction stage, which requires gradients at the near-boundary cells. However, gradient reconstruction in turn, depends on the unknown boundary value and thus is explicit. To avoid iterations, an implicit way is proposed by combining the gradient relation with a relation based on Taylor series approximation.

It has been found that there is no appreciable difference between the two data reconstruction methods proposed, which are based on Gauss or least square methods, in finding the boundary values for the truncated backward-facing step problem studied in this work. The results of truncated domain agree well with that of the full domain and the benchmark values when the channel of length $30H$ is truncated down to $8H$.

The boundary conditions for the pressure and pressure correction equations are categorized into two different types depending on whether pressures or velocities are specified at the boundary. The details of the implementation of different boundary conditions for the frequently used boundary types encountered in practice are formulated in concise form and can be used by programmers with no ambiguity.

The validity of implementations is checked by comparing the results with some well-known benchmark problems. For the unsteady flow problem, the comparison of the drag coefficient and the Strouhal number with the published results is satisfactory despite the rather small number of cell used in the present study. For the natural convection problem over a horizontal isothermal cylinder, the present results agree closely with the previous predictions.

The results of the natural convection in a square open cavity problem using an extended domain indicates the existence of a vortex flow at a rather high Rayleigh number of 10^7 in the vicinity of the bottom wall, which is missing in simulations performed over the domain consisting of the cavity only.

Appendix A

Haselbacher and Blazek [46] have derived formulation of the unweighted least square method for the calculation of the gradients on arbitrary grids. It is possible to give different weights, w_i, to the contribution of each neighbor node i to gradient reconstruction. The gradient using the weighted least square reconstruction becomes

$$(\nabla\phi)_P = \begin{bmatrix} (\nabla^x\phi)_P \\ (\nabla^y\phi)_P \\ (\nabla^z\phi)_P \end{bmatrix} = \begin{bmatrix} \left(\frac{\partial\phi}{\partial x}\right)_P \\ \left(\frac{\partial\phi}{\partial y}\right)_P \\ \left(\frac{\partial\phi}{\partial z}\right)_P \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{\mathrm{nb}(P)} C_i^x w_i(\phi_i - \phi_P) \\ \sum_{i=1}^{\mathrm{nb}(P)} C_i^y w_i(\phi_i - \phi_P) \\ \sum_{i=1}^{\mathrm{nb}(P)} C_i^z w_i(\phi_i - \phi_P) \end{bmatrix} \tag{A.1}$$

where the C terms are the same as that of the unweighted method:

$$C_i^x = \alpha_{j,1} - \frac{r_{12}}{r_{11}}\alpha_{i,2} + \psi\alpha_{i,3}, \quad C_i^y = \alpha_{i,2} - \frac{r_{23}}{r_{22}}\alpha_{i,3}, \quad C_i^z = \alpha_{i,3} \tag{A.2}$$

The α terms are

$$\begin{aligned} \alpha_{i,1} &= \frac{w_i\Delta x_i}{r_{11}^2}, \quad \alpha_{i,2} = \frac{1}{r_{22}^2}\left(w_i\Delta y_i - \frac{r_{12}}{r_{11}}w_i\Delta x_i\right) \\ \alpha_{i,3} &= \frac{1}{r_{33}^2}\left(w_i\Delta z_i - \frac{r_{23}}{r_{22}}w_i\Delta y_i + \psi w_i\Delta x_i\right), \end{aligned} \tag{A.3}$$

where

$$\psi = \frac{r_{12}r_{23} - r_{13}r_{22}}{r_{11}r_{22}}, \quad \Delta x_i = x_i - x_P, \quad \Delta y_i = y_i - y_P, \quad \Delta z_i = z_i - z_P \tag{A.4}$$

and the r terms are

$$\begin{aligned} r_{11} &= \sqrt{\sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta x_i)^2}, \quad r_{12} = \frac{1}{r_{11}} \sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta x_i)(w_i\Delta y_i) \\ r_{13} &= \frac{1}{r_{11}} \sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta x_i)(w_i\Delta z_i), \quad r_{22} = \sqrt{\sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta y_i)^2 - r_{12}^2} \\ r_{23} &= \frac{1}{r_{22}} \left(\sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta y_i)(w_i\Delta z_i) - r_{12}r_{13} \right), \quad r_{33} = \sqrt{\sum_{i=1}^{\mathrm{nb}(P)} (w_i\Delta z_i)^2 - (r_{13}^2 + r_{23}^2)} \end{aligned} \tag{A.5}$$

The weights, w_i can be selected in the form

$$w_i = \frac{1}{|\mathbf{r}_i - \mathbf{r}_P|^n} \quad n = 0, 1, 2 \tag{A.6}$$

where r_i is the position vector of the neighboring point i around point P. For inverse distance weighting $n = 1$ and hence

$$w_i = 1/\sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2} \tag{A.7}$$

Inserting the gradient relation (A.1) into Eq. (24)

$$\phi_b = \frac{\phi_P + \left(\sum_{\substack{i=1 \\ i \neq b}}^{\mathrm{nb}(P)} \mathbf{C}_i w_i (\phi_i - \phi_P) \right) \cdot \mathbf{Pb} - \mathbf{C}_b w_b \phi_P \cdot \mathbf{Pb}}{1 - \mathbf{C}_b w_b \cdot \mathbf{Pb}} \tag{A.8}$$

where $\mathbf{C}_i$ is the vector $\mathbf{C}_i = (C_i^x, C_i^y, C_i^z)$ and the summation is all over the neighboring points, except the boundary point b.

References

[1] S. V. Patankar and D. B. Spalding, A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-dimensional Parabolic Flows, *Int. J. Heat Mass Transfer*, vol. 15, pp. 1767–1806, 1972.

[2] J. P. Van Doormaal and G. D. Raithby, Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows, *Numer. Heat Transfer*, vol. 7, pp. 147–163, 1984.

[3] S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Washington, DC, 1980.

[4] R. Issa, A. D. Gosman, and A. P. Watkins, The Computation of Compressible and Incompressible Recirculating Flows by a Non-iterative Implicit Scheme, *J. Comput. Phys.*, vol. 62, pp. 66–82, 1986.

[5] P. L. Woodfield and K. Suzuki, Performance of a Three-dimensional, Pressure Based, Unstructured Finite-volume Method for Low-Reynolds-number Incompressible Flow and Wall Heat Transfer Rate Prediction, *Numer. Heat Transfer B*, vol. 43, pp. 403–423, 2003.

[6] Y.-Y. Tsui and Y.-F. Pan, A Pressure-correction Method for Incompressible Flows Using Unstructured Meshes, *Numer. Heat Transfer B*, vol. 49, pp. 43–65, 2006.

[7] A. Dalal, V. Eswaran, and G. Biswas, A Finite-volume for Navier–Stokes Equations on Unstructured Meshes, *Numer. Heat Transfer B*, vol. 54, pp. 238–259, 2008.

[8] Y.-Y. Tsui and T.-C. Wu, A Pressure-based Unstructured-grid Algorithm for All Speed Flows, *Numer. Heat Transfer B*, vol. 53, pp. 75–96, 2008.
[9] S.-C. Xue and G. W. Barton, A Finite Volume Formulation for Transient Convection and Diffusion Equations with Unstructured Distorted Grids and Its Applications in Fluid Flow Simulations with a Collocated Variable Arrangement, *Comput. Meth. Appl. Mech. Eng.*, vol. 253, pp. 146–159, 2013.
[10] P. Ding and D. L. Sun, A Pressure-based Segregated Solver for Incompressible Flow on Unstructured Grids, *Numer. Heat Transfer B*, vol. 64, pp. 460–479, 2013.
[11] S. Majumdar, Role of Underrelaxation in Momentum Interpolation for Calculation of Flow with Nonstaggered Grids, *Numer. Heat Transfer*, vol. 13, pp. 125–132, 1988.
[12] S. K. Choi, Note on the Use of Momentum Interpolation Method for Unsteady Flows, *Numer. Heat Transfer A*, vol. 36, pp. 545–550, 1999.
[13] B. Yu, W.-Q Tao, and J.-J. Wei, Discussion on Momentum Interpolation Method for Collocated Grids of Incompressible Flow, *Numer. Heat Transfer B*, vol. 42, pp. 141–166, 1999.
[14] A. Cubero and N. Fueyo, A Compact Momentum Interpolation Procedure for Unsteady Flows and Relaxation, *Numer. Heat Transfer B*, vol. 52, pp. 471–493, 2007.
[15] C. M. Rhie and W. L. Chow, Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation, *AIAA J.*, vol. 21, pp. 1525–1535, 1983.
[16] S. R. Mathur and J. Y. Murthy, Pressure Boundary Conditions for Incompressible Flow Using Unstructured Meshes, *Numer. Heat Transfer B*, vol. 32, pp. 283–298, 1997.
[17] K. M. Kelkar and D. Choudhury, Numerical Method for the Prediction of Incompressible Flow and Heat Transfer in Domains with Specified Pressure Boundary Conditions, *Numer. Heat Transfer B*, vol. 38, pp. 15–36, 2000.
[18] S.-C. Xue and G. W. Barton, Implementation of Boundary Conditions and Global Mass Conservation in Pressure-based Finite Volume Method on Unstructured Grids for Fluid Flow and Heat Transfer Simulations, *Int. J. Heat Mass Transfer*, vol. 55, pp. 5233–5243, 2012.
[19] D. K. Gartling, A Test Problem for Outflow Boundary Conditions – Flow Over a Backward-facing Step, *Int. J. Numer. Meth. Fluids*, vol. 11, pp. 953–967, 1990.
[20] J. Liu, Open and Traction Boundary Conditions for the Incompressible Navier–Stokes Equations, *J. Comput. Phys.*, vol. 228, pp. 7250–7267, 2009.
[21] E. Mitsoulis and N. A. Malamataris, Free (Open) Boundary Condition: Some Experiences with Viscous Flow Simulations, *Int. J. Numer. Meth. Fluids*, vol. 68, pp. 1299–1323, 2012.
[22] S. Dong, G. E. Karniadakis, and C. Chryssostomidis, A Robust and Accurate Outflow Boundary Condition for Incompressible Flow Simulations on Severely-truncated Unbounded Domains, *J. Comput. Phys.*, vol. 261, pp. 83–105, 2014.
[23] M. Darbandi and S. Vakilipour, Using Fully Implicit Conservative Statements to Close Open Boundaries Passing Through Recirculations, *Int. J. Numer. Meth. Fluids*, vol. 53, pp. 371–389, 2006.
[24] W. A. Shyy, Effects of Open Boundary on Incompressible Navier–Stokes Flow Computation: Numerical Experiments, *Numer. Heat Transfer*, vol. 12, pp. 157–178, 1987.
[25] M. Sani and M. S. Saidi, A Lagged Implicit Segregated Data Reconstruction Procedure to Treat Open Boundaries, *J. Comput. Phys.*, vol. 229, pp. 5418–5431, 2010.
[26] E. Blosch, E. Shyy, and R. Smith, The Role of Mass Conservation in Pressure-based Algorithms, *Numer. Heat Transfer B*, vol. 24, pp. 415–429, 1993.
[27] S.-C. Xue and G. W. Barton, Incompressible Fluid Flow Simulations with Flow Rate as the Sole Information at Synthetic Inflow and Outflow Boundaries, *Int. J. Numer. Meth. Fluids*, vol. 78, pp. 739–760, 2015.
[28] P. K. Khosla and S. G. Rubin, A Diagonally Dominant Second-order Accurate Implicit Scheme, *Comput. Fluids*, vol. 2, pp. 207–209, 1974.
[29] A. Harten, High Resolution Schemes for Hyperbolic Conservation Laws, *J. Comput. Phys.*, vol. 49, pp. 357–393, 1983.
[30] P. K. Sweby, High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws, *SIAM J. Numer. Anal.*, vol. 21, pp. 995–1011, 1984.
[31] B. P. Leonard, Simple High Accuracy Resolution Program for Convective Modeling of Discontinuities, *Int. J. Numer. Meth. Fluids*, vol. 8, pp. 1291–1318, 1988.
[32] J. H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, 3rd ed., Springer, Berlin, 2002.
[33] P. Ding and D. L. Ding, A Pressure-based Segregated Solver for Incompressible Flow on Unstructured Grids, *Numer. Heat Transfer B*, vol. 64, pp. 460–479, 2013.
[34] T. C. Papanastasiou, N. Malamataris, and K. Ellwood, A New Outflow Boundary Condition, *Int. J. Numer. Meth. Fluids*, vol. 14, pp. 587–608, 1992.
[35] C. Geuzaine and J. F. Remacle, Gmsh: A Three-dimensional Finite Element Mesh Generator with Built-in Pre- and Post-Processing Facilities, *J. Numer. Meth. Eng.*, vol. 79, no. 11, pp. 1309–1331, 2009.
[36] C. F. Lange, F. Durst, and M. Breuer, Momentum and Heat Transfer from Cylinders in Laminar Crossflow at $10^{-4} < Re < 200$, *Int. J. Heat Mass Transfer*, vol. 41, pp. 3409–3430, 1998.

[37] H. Q. Zhang, U. Fey, and B. R. Noack, On the Transition of the Cylinder Wake, *Phys. Fluids*, vol. 7, pp. 779–794, 1995.
[38] Z. Lilek, S. Muzaferija, M. Peric, and V. Seidl, Computation of Unsteady Flows Using Nonmatching Blocks of Structured Grid, *Numer. Heat Transfer B*, vol. 32, pp. 403–418, 2007.
[39] J. M. Wu and W. Q. Tao, Numerical Computation of Laminar Natural Convection Heat Transfer Around a Horizontal Compound Tube with External Longitudinal Fins, *Heat Transfer Eng.*, vol. 28, pp. 93–102, 2007.
[40] T. H. Kuehn and R. J. Goldstein, Numerical Solution to the Navier–Stokes Equations for Laminar Natural Convection About a Horizontal Isothermal Circular Cylinder, *Int. J. Heat Mass Transfer*, vol. 23, pp. 971–979, 1980.
[41] P. Wang and R. Kahawita, Numerical Computation of the Natural Convection Flow About a Horizontal Cylinder Using Splines, *Numer. Heat Transfer A*, vol. 17, pp. 191–215, 1990.
[42] T. Saitoh, T. Sajiki, and K. Maruhara, Bench Mark Solutions to Natural Convection Heat Transfer Problem Around a Horizontal Circular Cylinder, *Int. J. Heat Mass Transfer*, vol. 36, pp. 1251–1259, 1993.
[43] Y. L. Chan and C. L. Tien, A Numerical Study of Two-dimensional Natural Convection in Square Open Cavities, *Numer. Heat Transfer*, vol. 8, pp. 65–80, 1985.
[44] A. A. Mohamad, Natural Convection in Open Cavities and Slots, *Numer. Heat Transfer A*, vol. 27, pp. 705–716, 1995.
[45] J. F. Hinojosa, R. E. Cabanillas, G. Alvarez, and C. E. Estrada, Nusselt Number for the Natural Convection and Surface Thermal Radiation in a Square Tilted Open Cavity, *Int. Commun. Heat Mass Transfer*, vol. 32, pp. 1184–1192, 2005.
[46] A. Haselbacher and J. Blazek, Accurate and Efficient Discretization of Navier–Stokes Equations on Mixed Grids, *AIAA J.*, vol. 38, pp. 2094–2102, 2000.