INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 16, 659–682 (1993)

# A DECOUPLING NUMERICAL METHOD FOR FLUID FLOW

SHIJIE LIU AND JACOB H. MASLIYAH

*Department of Chemical Engineering, University of Alberta, 536 Chemical-Mineral Engineering Building, Edmonton, Canada T6G 2G6*

## SUMMARY

A first-order non-conforming numerical methodology, *Separation method*, for fluid flow problems with a 3-point exponential interpolation scheme has been developed. The flow problem is decoupled into multiple one-dimensional subproblems and assembled to form the solutions. A fully staggered grid and a conservational domain centred at the node of interest make the decoupling scheme first-order-acccurate. The discretization of each one-dimensional subproblem is based on a 3-point interpolation function and a conservational domain centred at the node of interest. The proposed scheme gives a guaranteed first-order accuracy. It is shown that the traditional upwind (or exponentially weighted upstream) scheme is less than first-order-accurate. The pressure is decoupled from the velocity field using the pressure correction method of SIMPLE. Thomas algorithm (tri-diagonal solver) is used to solve the algebraic equations iteratively. The numerical advantage of the proposed scheme is tested for laminar fluid flows in a torus and in a square-driven cavity. The convergence rates are compared with the traditional schemes for the square-driven cavity problem. Good behaviour of the proposed scheme is ascertained.

KEY WORDS Navier–Stokes equations 3-Point exponential upwind Pressure perturbation Stability Curved channel Laminar flow Square-driven cavity

## INTRODUCTION

The numerical study of fluid flow through direct formulations, i.e primitive variable approach, needs special attention. Great strides have been made in the past two decades. All the methodologies, e.g. finite element, finite difference and finite volume, gave rise to new issues to be studied. Some of the issues are: the inf–sup condition of the finite element method,[1–3] the checkerboard effect of the finite difference method,[4] the Petrov–Galerkin or streamline-weighted scheme[5] and the upwind differencing scheme.[6] Despite that certain conditions must be satisfied, a practical scheme must also use at least a partially direct algebraic equations solution strategy in order to obtain an accurate solution. This largely restricted the development of numerical schemes, especially for three-dimensional problems. In this regard, the pressure distribution cannot be easily solved as it appears in the momentum equations. This gave rise to the formulation of the pressure correction method of Patankar.[4]

It is well known that a convergent numerical scheme must be both consistent and stable. The consistency is defined as the limit behaviour of the discrete equations used in a numerical scheme to approach the governing differential equation as the mesh size approaches zero. The stability of a numerical scheme is related to the solution behaviour of the discrete equations. When an iterative method is used, an unstable scheme may have difficulty in generating the solution.

0271–2091/93/080659–24$17.00

*Received 21 June 1991*
*Revised 22 October 1992*

In this study, we develop a scheme which is spurious-modes-free and stable. Attention to stability will be directed towards obtaining an accurate representation of the flow behaviour (wiggle-free) rather than the ability to obtain a solultion alone. The strategy of the scheme is focused on making a complex problem easy to deal with. At each step, the flow problem is subdivided into multi-subproblems and a best possible choice of discretization and related treatments is used to obtain convergence.

The pressure field is solved with the pressure correction method due to Patankar[4] to avoid the painstaking solution of the standard finite element approach. For higher Reynolds number flows, the solution of the pressure correction equation due to the SIMPLE method is still not an easy task,[7] without a carefully chosen solution algorithm. The SIMPLE method requires a fast and a less demanding storage-wise algebraic equation solver. By separating the multidimensional dependence into multi-subproblems and introducing some extrapolation or first-order-accuracy-guaranteed exponential interpolation, we eliminate the stability problem associated with the traditional centred scheme. Owing to the strategy of the method, we call it the separation method.

To obtain a meaningful solution, we decided to give up the conforming property. That is, the variable of interest and/or its fluxes may not be uniquely defined inside the computational domain with respect to the region of consideration. Instead, the consistency of the discretization is to be checked. It should be noted that a conforming scheme yields a globally consistent interpolation for the variables under consideration. Hence, the variable of consideration at any given point has only one unique interpolation equation no matter how one refers to the value at that point. The conforming methods pay full attention to the consistency in the interpolation of the variables and their fluxes. However, they introduce excessive errors when the flow Reynolds number is large. We work from a different perspective by focusing on the local behaviour of every point under consideration. A 3-point interpolation scheme is introduced in this study. To accommodate the 3-point interpolation scheme, a conservational domain is used.

The following two sections serve as a general guideline for the proposed scheme. More details are given in the first numerical example.

## DOMAIN DISCRETIZATION AND NODAL ARRANGEMENTS

The primary goal here is to establish a centred conservational scheme and to use a fully staggered grid[8] to maintain both simplicity and the inf–sup condition requirement. For a relatively simple geometry, we discretize the domain into a relatively uniform mesh where the mesh lines are parallel to the co-ordinate lines, i.e. undistorted mesh. The non-equal spacing does not need a mapping. However, a centred conservational scheme can be obtained by simply placing the node of interest in the centre of the symmetrically chosen domain even though the actual nodes are not symmetrical with the co-ordinate axes. We make the undistorted non-uniform mesh as the general case for discussion in this study.

The numbering of the nodes is based on an appropriate book-keeping scheme. A good choice of node numbering can be expressed as follows. The three curvilinear directions are identified, followed by numbering each of these directions independently and, finally, the nodes are identified by a 3-indexed $(i,j,k)$ tensor related to the three curvilinear directions. In this way, no special attention is needed to keep track of the neighbouring nodes.

The exact conservational domain for each node on or near a boundary, however, will be considered in light of the specified boundary condition in order to avoid the introduction of an error by requiring values outside the computational domain. If a Dirichlet condition is specified, the nodes next to the boundary will have a conservational domain bounded by the boundary. This allows the values to be exactly specified at the boundary. For a Neumann boundary, a node

on the boundary will have a symmetrically extrapolated conservational domain. This allows the fluxes to be specified exactly at the boundary. The above treatment of boundaries can achieve consistency and avoid the non-consistent accuracy match for finite difference and for finite volume methods.

To address fully the arrangement of nodes for the different primitive variables, we establish the following convention: when the direction of the velocity component of interest (the centre nodal variable) coincides with a given co-ordinate axis, we denote that direction as the parallel direction, while all other directions become perpendicular directions. We name the parallel velocity as the velocity of coincident axis, while the perpendicular velocity stands for the velocity that is not in the direction of the axis. This convention will be very convenient as we deal with the axes one at a time.

The inf–sup condition can be represented as follows:

$$\exists_{\alpha>0} \inf_{q\in Q} \sup_{\mathbf{u}\in V} \frac{-\int_Q \nabla\cdot\mathbf{u}q}{\|\mathbf{u}\|_V \|q\|_Q} \geqslant \alpha, \tag{1}$$

where $\alpha$ is a positive number independent of the mesh size, $\mathbf{u}$ is the velocity field, $V$ is the functional space of $\mathbf{u}$ and $q$ is a variable taking its value from the functional space of the pressure, $Q$.

It is known that equation (1) can be satisfied by a proper choice of (match between) $V$ and $Q$. It is independent of the co-ordinate system. Equation (1) can be satisfied by arranging the velocity and pressure nodes in the following manner.

As shown in Figure 1, the pressure nodes are located only in the parallel direction (in one element), exactly at the centre of two adjacent velocity nodes. In other words, when the continuity equation is treated (pressure is the variable of interest), all the velocity components are presented symmetrically on the nodal axis away from the centre, and only the parallel velocity is present on each axis.

We will come back to address this condition later in the pressure iteration section.

## FORMAL DISCRETIZATION

To carry out the discretization of the flow equations, the problem of interest must be initially written either in an orthogonal co-ordinate or in a non-orthogonal co-ordinate system prior to domain discretization. For a steady flow problem, the Navier–Stokes equation, in the Cartesian co-ordinate system for a given $x_i$ direction, is given by

$$\sum_j \frac{\partial}{\partial x_j}\left(p\delta_{ij}+\rho u_i u_j-\mu\frac{\partial u_i}{\partial x_j}\right)=0, \tag{2}$$

where $\delta_{ij}$ is the Kronecker delta function, $x_j$ is the Cartesian co-ordinate, $p$ is the pressure and $u_i$ is the velocity field.

The governing equations can be rewritten in the weak form with an added weighting function. The weak form governing equation is

$$\frac{1}{V_e}\int\int_{\Omega_e}\int\sum_j \frac{\partial}{\partial x_j}\left(p\delta_{ij}+\rho u_i u_j-\mu\frac{\partial u_i}{\partial x_j}\right)w_i\,\mathrm{d}v=0, \tag{3}$$

where $V_e$ is the volume of the conservational domain and $w_i$ is a weighting function. We take $w_i$ as

$$w_i\in\{w_i=1 \text{ if } x\in \text{conservational domain; } w_i=0 \text{ for all other cases}\}.$$

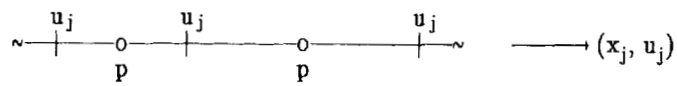


Figure 1. Pressure nodes placement

The multidimensionality of equation (3) is the most significant obstacle to developing a good numerical scheme. One-dimensional problems are easy to deal with and are better understood. To tackle the multidimensionality, we introduce a strategy that allows us to separate the multidimensional problem into multiple one-dimensional subproblems. Since the node of interest is at the centre of the conservational domain, we can quadratically approximate the integration over a dimension using the centre value when a derivative is not involved in that dimension. Hence, equation (3) can be decoupled further to reduce the multidimensional problem into multiple pseudo-one-dimensional problems. Before the decoupling is performed, the governing equation is regrouped in the following manner:

$$\sum_j E_{ij} = -\frac{1}{V_e}\int\int_{\Omega_e}\int w_i \frac{\partial p}{\partial x_i}\,dv, \tag{4a}$$

$$\sum_j E_{ij} = -\frac{1}{V_e}\int\int_{\Omega_e}\int w_i \frac{\partial}{\partial x_i}\left(\rho u_i u_j - \frac{\partial u_i}{\partial x_j}\right)dv. \tag{4b}$$

The decoupling of the multidimensional problem is achieved by approximating equation (4b) by

$$E_{ij} \simeq \frac{1}{l_e}\int w_i \frac{\partial}{\partial x_i}\left(\rho u_i u_j - \frac{\partial u_i}{\partial x_j}\right)dl, \tag{4c}$$

where $l_e$ is the length of the conservational domain the $x_j$ direction and $E_{ij}$ is the $x_j$-component of the $x_i$-momentum. Equations (4a) and (4b) are multidimensional problems. After using the approximation to the integration over the dimensons of $x_k$'s ($k \neq j$) by the centre nodal value, the multiple intregation of equation (4b) is approximated by a single integration as shown by equation (4c), which represents a one-dimensional problem. This leads us to consider one-dimensional discretization only and the resulting discrete equation coefficient (stiffness) matrix consists only of the node of interest and its two neighbours. This approximation of equation (4b) by (4c) is a key element of the method presented in this paper.

For example, the weak form of the two-dimensional momentum equations is given by

$$\begin{aligned}
&\frac{1}{A_e}\int\int_{xy} w_x\left[\rho\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left(\rho u_x u_x - \mu\frac{\partial u_x}{\partial x}\right) + \frac{\partial}{\partial y}\left(\rho u_x u_y - \mu\frac{\partial u_x}{\partial y}\right)\right]dx\,dy = 0,\\
&\frac{1}{A_e}\int\int_{xy} w_y\left[\rho\frac{\partial p}{\partial y} + \frac{\partial}{\partial x}\left(\rho u_x u_y - \mu\frac{\partial u_y}{\partial y}\right) + \frac{\partial}{\partial y}\left(\rho u_y u_y - \mu\frac{\partial u_y}{\partial y}\right)\right]dx\,dy = 0.
\end{aligned} \tag{4d}$$

The separation of space variables for the $x$-momentum equation is given by

$$E_{xx} + E_{xy} = -\int\int_{xy} \frac{w_x}{A_e}\frac{\partial p}{dx}\,dx\,dy \tag{4e}$$

and the momentum components are defined by

$$E_{xx}=\frac{1}{A_e}\int\int_{xy} w_x\frac{\partial}{\partial x}\left(\rho u_x u_x-\mu\frac{\partial u_x}{\partial x}\right)\mathrm{d}x\,\mathrm{d}y, \tag{4f}$$

$$E_{xy}=\frac{1}{A_e}\int\int_{xy} w_y\frac{\partial}{\partial y}\left(\rho u_x u_y-\mu\frac{\partial u_x}{\partial y}\right)\mathrm{d}x\,\mathrm{d}y. \tag{4g}$$

$E_{xx}$ and $E_{xy}$ are approximated, respectively, by

$$E_{xx}\simeq\frac{1}{\Delta x}\int_x w_x\frac{\partial}{\partial x}\left(\rho u_x u_x-\mu\frac{\partial u_x}{\partial x}\right)\mathrm{d}x, \tag{4h}$$

$$E_{xy}\simeq\frac{1}{\Delta y}\int_y w_y\frac{\partial}{\partial y}\left(\rho u_x u_y-\mu\frac{\partial u_x}{\partial y}\right)\mathrm{d}y, \tag{4i}$$

where $A_e=\Delta x\Delta y$, and $\Delta x$ and $\Delta y$ are the widths of the conservational domain in the $x$ and $y$ directions, respectively. Equations (4f) and (4g) are one-dimensional problems if $E_{xx}$ and $E_{xy}$ are regarded as source terms.

The quadratic interpolation function in a non-uniform one-dimensional grid can be written as follows:

$$\phi=\phi_i+\frac{(\phi_{i+1}-\phi_i)(h_1+x)h_1-(\phi_{i-1}-\phi_i)(h_2-x)h_2}{h_1h_2(h_1+h_2)}x, \tag{5}$$

where $h_1$ and $h_2$ are the two grid spacings of the node of interest. A brief illustration of a decoupled one-dimensional nodal arrangement is shown in Figure 2.

The space variable $x$ has its origin at the node of interest. The radius of the conservational domain is $h$.

The discretization of a one-dimensional problem is simple to handle. The general conservational domain can be described as follows:

1. For the perpendicular direction, as shown in Figure 3, we use the intersection with the corresponding perpendicular mesh lines of the parallel velocity as the conservational domain boundary.
2. For the parallel direction, however, the conservational domain can be arbitrary. A viable choice would be using at most half of the longer side length as the conservational radius, keeping the node of the shorter side within the conservational domain. An illustration of this situation is shown in Figure 2, where the variable of interest is $u_j$ and the space variable $x$ is $x_j$.

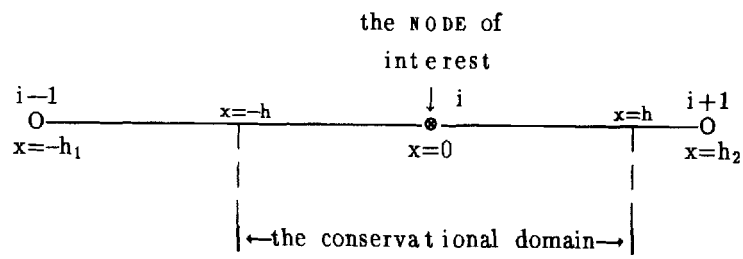


Figure 2. Nodal arrangement for a non-uniform one-dimensional mesh

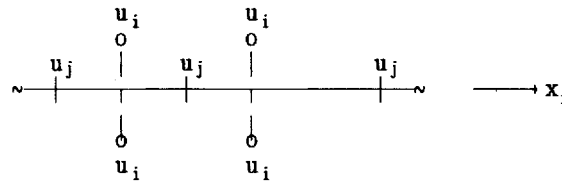


Figure 3. Conservational domain set-up in a perpendicular direction

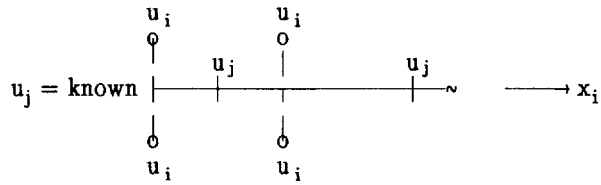


Figure 4. Conservational domain set-up next to a Dirichlet boundary for a perpendicular direction

3. If the node of interest is along the perpendicular direction and next to a Dirichlet boundary (no-slip or known entrance–exit), we can choose the side length towards the boundary as the radius of the conservational domain. This is shown in Figure 4.

Equation (4c) is treated using an appropriate interpolation functions based on the three available nodes, such as equation (5). Special attention, however, will be given to the advection-dominated subproblems. Since no simple normal mode exists (although Patankar[4] claimed to be exact) and the quadratic interpolation, equation (5), cannot be used, we employ the following 3-point exponential interpolation type:

$$\phi = \phi_i + a_i e^{Re\, u_i x} + b_i x, \tag{6}$$

where $\phi$ is the velocity of interest and $u_i$ is the parallel velocity of the $x$ direction evaluated at the centre of the conservational domain or the node of interest, $i$.

Leonard[9] proposed a 3-point exponential scheme and it has the following form;

$$\phi' = \phi_i + a_i'(e^{b_i' x} - 1). \tag{7}$$

However, equation (7) is rather difficult to work with and it leads to unstable solutions when advection is dominated.[9]

The interpolation function that corresponds to equation (6) can be obtained using the three available nodes as

$$\phi = \phi_i + \frac{h_1(e^{Re u_i x} - 1) + x(e^{-Re u_i h_1} - 1)}{h_1(e^{Re u_i h_2} - 1) + h_2(e^{-Re u_i h_1} - 1)}(\phi_{i+1} - \phi_i) + \frac{h_2(e^{Re u_i x} - 1) + x(e^{-Re u_i h_2} - 1)}{h_1(e^{Re u_i h_2} - 1) + h_2(e^{-Re u_i h_1} - 1)}(\phi_{i-1} - \phi_i). \tag{8}$$

The interpolation equation (8) reduces to a standard upwind scheme (but not the traditional one) for $Re u_i \to \pm\infty$, with the sign taken to be the same as that of $u_i$ in the above equation. It can be shown that when $Re u_i \to 0$, equation (8) reduces the quadratic interpolation equation (5). For a simple upwind scheme, we can combine the quadratic interpolation equation (5) by setting the off-diagonal terms to be no greater than zero in the resulting stiffness matrices. By doing so, we can dampen the oscillations while not altering the solution. It is worth mentioning that a tradi-

tional centred treatment of the right-hand side of equation (4b) for the stiffness matrix does not lead to absolute divergence. As a matter of fact, it can be solved quite successfully with careful arrangements.[10]

It is clear that the above strategy results in a quadratic accuracy for a small Reynolds number flow, i.e. $\phi = \phi_0 + ax + bx^2 + O(x^3)$, and linear accuracy for a large Reynolds number flow, i.e. $\phi = \phi_0 + ax + O(x^2)$. It should be noted that for the latter case, an ordinary upwind/exponentially weighted implementation gives only zeroth-order accuracy, i.e. $\phi = \phi_0 + O(x)$, in the interpolation.

An interesting by-product of this 3-point exponential implementation is that we are able to produce an accuracy equivalent to a total pressure-corrected exponentially weighted scheme of De Henau *et al.*[11] while treating a problem in a multidimensional situation. The complexity of the proposed scheme is not more than those advanced by Raithby and Torrance,[12] and Patankar.[4]

The discrete momentum equation can be obtained for an undistorted mesh as

$$\phi\mathrm{IJK}_{ijk}\phi_{ijk} + \phi\mathrm{MJK}_{ijk}\phi_{i-1\,jk} + \phi\mathrm{PJK}_{ijk}\phi_{i+1\,jk} + \phi\mathrm{IMK}_{ijk}\phi_{ij-1\,k} + \phi\mathrm{IPK}_{ijk}\phi_{ij+1\,k}$$
$$+ \phi\mathrm{IJM}_{ijk}\phi_{ijk-1} + \phi\mathrm{IJP}_{ijk}\phi_{ijk+1} = -C\phi_{ijk}(p_\beta - p_{\beta-1}) + B\phi_{ijk},$$

where $\phi_{ijk}$ stands for any velocity component at the node $(i, j, k)$, $\phi$IJK, $\phi$MJK, $\phi$PJK, $\phi$IMK, $\phi$IPK, $\phi$IJM, $\phi$IJP, $C\phi$ and $B\phi$ are coefficients, $(i, J, k)$ is the global index for the node of interest and $\beta$ is the global index of the parallel direction with all the perpendicular directional indices fixed. For example, for the velocity component in the direction of $i$, i.e. $\phi = u_i$, the pressure term $(p_\beta - p_{\beta-1})$ is equivalent to $(p_{ijk} - p_{i-1\,jk})$. The pressure $p$ has been treated linearly in order to satisfy the inf–sup condition. The formulation as described above renders the coefficients to be similar to that of a 7-point difference stencil. With the present nodal arrangement, the required interpolations for the variables that are not currently under consideration can be made linearly. Best of all, a given interpolation involves only two points and no extrapolation is used. For example, the coefficient of the pressure gradient term is

$$C\phi_{ijk} = \frac{1}{x_\beta - x_{\beta-1}}, \tag{10}$$

where $x_\beta$ is the co-ordinate location where $x$ is of the parallel direction of the variable of interest.

It is important to note that, for the boundary, the treatment must be consistent with the interior domain. The boundary condition must be specified exactly. To summarize, we list the treatment of the boundary as follows:

1. Only when the gradient is known should one use a symmetrically extrapolated grid outside of the domain of interest. Under no other circumstances should one use a grid point outside the domain of interest.
2. For a Dirichlet velocity boundary condition, all the velocity components must have a grid point on the boundary. However, the pressure node is not necessarily placed on the boundary. As a matter of fact, only the interior nodes for the pressure are present in this scheme for a Dirichlet velocity boundary.
3. In the case where the velocity gradient is known on a boundary, the pressure node can be placed on the boundary; however, the normal velocity component should not be placed on the boundary.
4. The pressure node must be present on the pressure-known boundary, where the normal velocity is not present. Here we should point out that the boundary condition for the normal velocity component should not be imposed when the pressure is known on the boundary. Instead, one must use a deduced boundary condition (gradient type) from the continuity

equation to fulfill both the boundary condition requirement of the momentum equation and the continuity equation.

Finally, to solve the discretized equations, we introduce some under-relaxation $F_\phi$, where

$$(1+F_\phi)\phi\mathrm{IJK}_{ijk}\phi_{ijk}+\phi\mathrm{MJK}_{ijk}\phi_{i-1jk}+\phi\mathrm{PJK}_{ijk}\phi_{i+1jk}+\phi\mathrm{IMK}_{ijk}\phi_{ij-1k}+\phi\mathrm{IPK}_{ijk}\phi_{ij+1k}$$
$$+\phi\mathrm{IJM}_{ijk}\phi_{ijk-1}+\phi\mathrm{IJP}_{ijk}\phi_{ijk+1}=F_\phi\phi\mathrm{IJK}_{ijk}\phi^0_{ijk}-C\phi_{ijk}(p_\beta-p_{\beta-1})+B\phi_{ijk}. \tag{11}$$

$\phi^0_{ijk}$ is the currently available value of $\phi_{ijk}$. The formed matrix structure is a 7-point difference module.

Owing to the complexity of flow problems, the above treatise is merely a guideline for the proposed scheme. A more detailed discretization can be found in the first numerical example.

## ACCURACY AND CONVERGENCE

The accuracy of a numerical scheme depends on the order of the interpolation scheme used and the consistency of the discretization. Since all terms except the derivative terms in the direction under consideration are evaluated at the centre node, the integration is quadratic. The consistency is then first-order for the weak form of the Navier–Stokes equation, i.e. equation (3). Hence, the accuracy for the advective (non-linear) term is linear. Based on past experience, a higher-order approximation could lead to instability [9,13] (most likely to have oscillation). On the other hand, a lower-order approximation, although it may be stable, gives a lower overall accuracy.

The accuracy of the current scheme is to be discussed relatively to the existing popular upwind/exponentially weighted schemes. Let us first examine the upwind scheme and the exponential scheme of Raithby and Torrance,[6] which was claimed to be exact by Patankar.[4] For simplicity, we consider a one-dimensional analogy and assume an advection-dominated flow.

The interpolation formula due to Raithby and Torrance[6] is given by

$$u=u_i+(a_i\mathrm{e}^{Reu^+x}-a_i\mathrm{e}^{Reu^+x_i})+O(x), \tag{12}$$

with $x_i<x<x_{i+1}$ and $a_i=f(u_i, u_{i+1})$ being a constant.

The consistency for the advective term becomes

$$\frac{\mathrm{d}uu}{\mathrm{d}x}\Rightarrow\frac{u^+u_i-u^-u_{i-1}}{2h}$$
$$=\frac{(u_i+u_{i+1})u_i-(u_{i-1}+u_i)u_{i-1}}{4h}$$
$$=\frac{\left(u+\dfrac{\mathrm{d}u}{\mathrm{d}x}h+\dfrac{\mathrm{d}^2u}{\mathrm{d}x^2}h^2\right)u-2\left(u-\dfrac{\mathrm{d}u}{\mathrm{d}x}h+\dfrac{\mathrm{d}^2u}{\mathrm{d}x^2}\mathrm{h}^2\right)\left(\dfrac{u}{2}-\dfrac{\mathrm{d}u}{\mathrm{d}x}h+\dfrac{\mathrm{d}^2u}{\mathrm{d}^2x}h^2\right)}{4h}+\cdots$$
$$=\frac{\mathrm{d}uu}{\mathrm{d}x}-\left[\frac{\mathrm{d}^2u}{\mathrm{d}x^2}u+\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)^2\right]h+O(h^2), \tag{13}$$

where $h$ is the radius of the conservational domain, i.e. $h=\Delta x/2$, $u^+=u_{i+1/2}$ and $u^-=u_{i-1/2}$. All the treatments above are traditional except that Taylor expansion is introduced here to examine the consistency.

The advective term appears to be of a first-order consistency if one does not look at the details of the contents in the square bracket of equation (13). It is, however, obvious to note that if

$$u=\alpha+\beta x, \tag{14}$$

neither the interpolation, equation (12), nor the consistency, equation (13), can satisfy equation (14). Hence, the scheme is actually of zeroth-order convergence, or, based on the appearance, it may be called a pseudo-first-order-convergent scheme. However, we are aware that it is customary to call the traditional upwind/exponentially weighted schemes as being first-order. The scheme is likely to experience large error in boundary layer flows having sharp gradients.

A reduction in the numerical dispersion present in the 2-point schemes (traditional upwind/exponentially weighted schemes) could be achieved by introducing the same interpolation scheme for both the variable of interest and the parallel velocity $u_i$.

The proposed scheme described here has the interpolation formulae

$$\phi = \phi_i + a_i e^{Re\, u_i x_I} + b_i x_I + O(x_I^2), \tag{15}$$

with $x_{i-1} - x_i \leq x_I \leq x_{i+1} - x_i$ and $a_i$ and $b_i$ being constant.

The consistency is given by

$$\begin{aligned} \frac{\mathrm{d}u\phi}{\mathrm{d}x} &\Rightarrow \frac{u^+(3\phi_i - \phi_{i-1}) - u^-(\phi_{i-1} + \phi_i)}{4h} \\ &= \frac{u^+\left(\phi + \dfrac{\mathrm{d}\phi}{\mathrm{d}x}h - \dfrac{d^2\phi}{dx^2}h^2\right) - u^-\left(\phi + \dfrac{\mathrm{d}\phi}{\mathrm{d}x}h + \dfrac{d^2\phi}{dx^2}h^2\right)}{2h} + \cdots \\ &= \frac{\mathrm{d}u\phi}{\mathrm{d}x} - u\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2}h + O(h^2). \end{aligned} \tag{16}$$

The above consistency equation (16) not only looks like but is indeed first-order-convergent, where only the higher-order derivative (second order being the lowest) is contained in the leading truncated term. It is easy to verify that equation (14) can be exactly satisfied. The current scheme gives the same order of accuracy as the traditional upwind or exponential scheme would give for a constant coefficient linear equation. In keeping with the convention of calling the 2-point schemes first-order, the current scheme may be regarded as a second-order scheme. However, we still consider the proposed scheme first-order.

By forcing the off-diagonal terms in the stiffness matrices (or difference molecule) not to be larger than zero, this scheme has guaranteed stability and optimum convergence. The interpolation and consistency evaluation are accurate up to the second-order derivative of the variable of interest. Although a more sophisticated upwind scheme can reduce the numerical dissipation, it is more than likely that this would introduce the symptom of an unstable scheme, an oscillation alien to the exact solution behaviour.[13] A quadratic or higher-order scheme for the advective terms alone does not improve the accuracy drastically. A quadratic scheme will have to make use of at least full 27 nodes and, hence, it adds complexity.

It should be noted that a conforming scheme for the proposed set-up can use at most 2-point interpolation functions. For a multidimensional problem, the use of a 2-point interpolation function does not yield a conforming scheme. This is a limitation of existing finite volume and control volume methods.

## PRESSURE ITERATION

The solution strategy of the pressure is that of the pressure correction method.[4] By perturbing the pressure field, the resulting change of velocities is obtained simply by using a Gauss–Seidel iteration. The new perturbed velocity field is then applied to satisfy the continuity. By forcing the continuity equation to be satisfied, we are able to solve for the appropriate pressure perturbation.

This procedure can be illustrated as follows:

$$\Delta\phi = \frac{1}{\phi \mathrm{IJK}} \frac{\partial \Delta p}{\partial x} = \frac{C\phi}{\phi \mathrm{IJK}} (\Delta p_{\beta} - \Delta p_{\beta-1}), \tag{17}$$

$$\nabla \cdot (u + \Delta u) = 0, \tag{18}$$

$$\nabla_{\mathrm{d}}^2 (\Delta p) = -\nabla \cdot u, \tag{19}$$

where $\Delta$ denotes the perturbation, i.e. a numerical difference and $\nabla_{\mathrm{d}}^2$ is the discrete Laplacian.

Since the pressure is corrected (perturbed) through the enforcement of the continuity equation and the velocity field is solved by satisfying the momentum equations, the order of approximation on the velocity and the pressure must be fixed such that the scheme is stable and spurious-mode-free. It is well known that schemes which are at least slightly more accurate in velocity interpolation than in pressure interpolation can give rise to a stable approximation.[14] In view of the type of approximation we use here, a quadratic approximation of velocity and linear pressure would give a stable scheme. By placing the pressure node of interest exactly in the symmetric position with the velocity nodes surrounding it, one can ensure that the continuity is second-order-accurate in velocity. In the case of a non-uniform grid, a velocity element has two asymmetric pressure nodes and the pressure approximation becomes linear. For the case of a uniform grid, even though the pressure approximation is second-order, past experience tells us that such a set-up does converge and the checkerboard effect is eliminated by the staggering.

After the justification of the pressure–velocity approximation, we can now discuss the practical part of the scheme, which is the solution for the pressure. Owing to the behaviour of the Gauss–Seidel iteration, the pressure equation itself needs special attention. If storage permits, a direct solver is advised. To ease the storage requirement of a direct solver, we have developed a two-dimensional direct solver based on block matrix inversion without any pivoting. This two-dimensional direct solver requires a storage place less than that of a band solver.[15]

Finally, we must apply some under-relaxation when perturbing the pressure. We have been successful in using a self-adjusting relaxation factor in correcting the pressure. The relaxation factor range is different from problem to problem. In general, a more complex problem is likely to require a smaller relaxation factor. The relaxation factor in the pressure perturbation equation is, however, different from that in the velocity field evaluation. Since the initial value (before perturbation) of the pressure perturbation is uniformly zero, the relaxation factor is introduced in the following way:

$$\mathrm{PIJK}_{ijk}\Delta p_{ijk} + \mathrm{PMJK}_{ijk}\Delta p_{i-1\,jk} + \mathrm{PPJK}_{ijk}\Delta p_{i+1\,jk} + \mathrm{PIMK}_{ijk}\Delta p_{ij-1k} + \mathrm{PIPK}_{ijk}\Delta p_{ij+1k}$$

$$+ \mathrm{PIJM}_{ijk}\Delta p_{ijk-1} + \mathrm{PIJP}_{ijk}\Delta p_{ijk+1} = -F_p(\nabla \cdot u)_{ijk}. \tag{20}$$

Equation (20) can also be written as

$$K\,\Delta p = F_p\,\mathrm{PRH},$$

where the pressure perturbation equation is arranged in the same manner as the momentum equations. PIJK is a coefficient of the same type as that of $\phi$IJK, and the notation is the same as that of equation (9).

In the formulation above, we have directly linked the SIMPLEc and SIMPLE in the pressure iteration. Our approach towards the pressure correction is close to that of SIMPLEc.

## NUMERICAL PROCEDURE

An important step for a numerical method is the execution of the solution procedure of the discretized equations. This is a most time/space-consuming step. We adopt an approach that requires minimum storage while using as much direct solution strategy as possible. The solution procedure is summarized as follows:

1. Initialize the velocity and pressure fields.
2. Discretize the domain as described in the domain discretization section.
3. Establish the matrices for the $x_1$-momentum equation while keeping all the other components and coefficients unchanged as described in the formal discretization section.
4. Solve the discretized equations through sweeping directions with a tri-diagonal solver and update the velocity component for the purpose of the pressure correction (step 6).
5. Repeat 1–3 but for the $x_2$- and $x_3$-velocity components.
6. Calculate the perturbation of pressure and update the pressure if the continuity is not satisfied.
7. Repeat 2–6 for a given number of inner loop iterations or until a certain criterion is met and update the velocity field.
8. Repeat 7 until the continuity is satisfied up to a preset tolerance at the first inner loop iteration.
9. Interpolate for the flow field, store the results and stop.

The inner loop (step 7) is necessary in order to separate the non-linearity from velocity–pressure coupling, especially when the Reynolds number is high and the initial guess is poor. If storage permits, store the coefficient matrices and step 7 becomes a repetition of steps 4–6.

## NUMERICAL EXAMPLES 1: LAMINAR FLOW IN CURVED (TOROIDAL) PIPES

A numerical flow problem will be solved to demonstrate the use of the proposed numerical scheme. In our group, the problem of laminar fluid flow in helical ducts has been solved previously for zero pitch, i.e. a torus.[16–18] We shall treat this problem once again using the proposed numerical scheme.

The laminar fluid flow will be that of a developing Newtonian fluid in a torus having a circular cross-section. The momentum equations are parabolized in the axial flow direction. A sketch of the co-ordinate set-up and the parameters scaling are shown in Figure 5.

The Navier–Stokes equations are written in the generic toroidal co-ordinate system. The radius of the pipe, $a$, and the average axial velocity, $U$, are used to render the various variables dimensionless, that is,

$$s=s'/a, \qquad r=r'/a, \qquad u=u'/2U. \qquad v=v'/2U,$$

$$p=Re\,p'/(4\rho U^2) \qquad \lambda=a/R_c, \qquad Re=2aU/\nu, \qquad Dn=Re\,\lambda^{1/2},$$

where $Re$ is Reynolds number, $Dn$ is the Dean number, $R_c$ is the radius of the coil, $\lambda$ is the curvature ratio, $u$ is the dimensionless axial velocity, $v$ is the dimensionless radial velocity, $w$ is the dimensionless angular velocity and the primed variables are dimensional.

The dimensionless governing equations become

*continuity equation*

$$\frac{1}{h_1}\frac{\partial u}{\partial s}+\frac{1}{rh_1}\frac{\partial(rh_1v)}{\partial r}+\frac{1}{rh_1}\frac{\partial(h_1w)}{\partial\theta}=0. \tag{21}$$

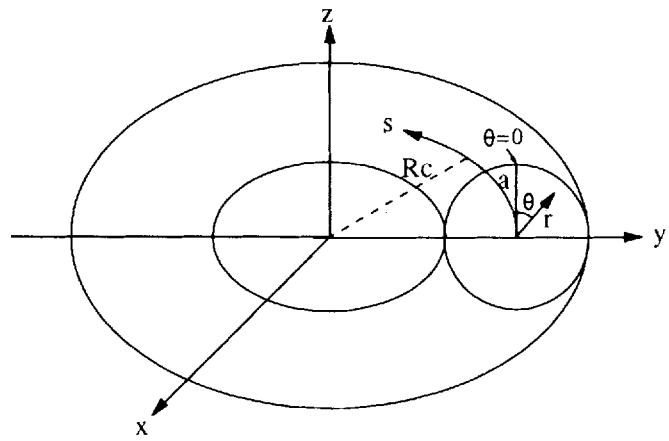


Figure 5. Toroidal co-ordinate set-up

The general form of the momentum equations:

$$\frac{1}{h_1}\frac{\partial}{\partial s}\left(Re\,u\phi-\frac{1}{h_1}\frac{\partial\phi}{\partial s}\right)+\frac{1}{rh_1}\frac{\partial}{\partial r}\left[rh_1\left(Re\,v\phi-\frac{\partial\phi}{\partial r}\right)\right]+\frac{1}{rh_1}\frac{\partial}{\partial\theta}\left(Re\,h_1 w\phi-\frac{h_1}{r}\frac{\partial\phi}{\partial\theta}\right)+d_\phi=S_\phi. \quad (22)$$

For various directions, the corresponding quantities $\phi$, $d_\phi$ and $S_\phi$ take the form:

*s-momentum*: $\phi=u$

$$d_\phi=\frac{v\sin\theta+w\cos\theta}{h_1}\lambda Re+\frac{\lambda^2}{h_1^2}, \quad (23a)$$

$$S_\phi=-\frac{1}{h_1}\frac{\partial p}{\partial s}+\frac{2\lambda}{h_1^2}\left(\sin\theta\frac{\partial v}{\partial s}+\cos\theta\frac{\partial w}{\partial s}\right), \quad (23b)$$

*r-momentum*: $\phi=v$

$$d_\phi=\frac{1+2h_1\lambda r\sin\theta}{r^2h_1^2}, \quad (24a)$$

$$S_\phi=-\frac{\partial p}{\partial r}+Re\left(\frac{\lambda\sin\theta}{h_1}u^2+\frac{w^2}{r}\right)-\frac{2\lambda\sin\theta}{h_1^2}\frac{\partial u}{\partial s}-\frac{2}{r^2}\frac{\partial w}{\partial\theta}-\frac{2h_1-1}{rh_1^2}\lambda\cos\theta w, \quad (24b)$$

*θ-momentum*: $\phi=w$

$$d_\phi=Re\frac{v}{r}+\frac{1}{h_1^2}\left(\lambda^2+\frac{2h_1-1}{r}\right), \quad (25a)$$

$$S_\phi=-\frac{\partial p}{\partial\theta}+Re\frac{\lambda\cos\theta}{h_1}u^2-\frac{2\lambda\cos\theta}{h_1^2}\frac{\partial u}{\partial s}+\frac{3h_1-1}{r^2h_1^2}\frac{\partial v}{\partial\theta}+\frac{\lambda\cos\theta}{rh_1^2}v, \quad (25b)$$

where $h_1$ is the metric coefficient, $h_1=1+\lambda r\sin\theta$.

Together with the boundary conditions,

$$u=v=w=0 \quad \text{at } r=1 \text{ for all } \theta,$$

$$p=0 \quad \text{at a certain interior point (reference).}$$

The average $u$ across the pipe is given by

$$\frac{\int u \, dA}{\int dA} = \frac{1}{2}.$$

For a fully developed flow, the streamfunction $\psi$ is given by

$$h_1 w = \frac{\partial \psi}{\partial r}. \tag{26}$$

*Domain discretization*

Since the domain is regular, we use an undistorted non-uniform mesh. The three curvilinear directions are those of the three co-ordinate axes. Since the governing equations are parabolized, the problem reduces to a two-dimensional problem with an extra time-like variable in the axial flow direction. Hence, the nodal arrangement is the same as that of a two-dimensional problem. A sketch of the nodal arrangement is shown in Figure 6. Where $p$ and $u$ share the same grid points, the $w$ grid is placed symmetrically surrounding each $p$ grid point in the $\theta$ direction, while $v$ grid points are placed to symmetrically surround each $p$ grid point in the $r$ direction, i.e. $\theta_j = \frac{1}{2}(\theta_{J+1} + \theta_J)$ and $r_i = \frac{1}{2}(r_{I-1} + r_I)$. The indices $I$ and $J$ stand for the $r$- and $\theta$-directional indices of the parallel velocity components, respectively. They are introduced to distinguish them from the indices $i$ and $j$ used for the grid staggering.

On the boundary (pipe wall), the pressure grid point is not present. The $v$ grid is regularly placed (the same rule as that of the interior nodes), $w$ and $u$ are placed at the boundary as well and there are no nodes outside of the boundary. The nodal layout is shown in Figure 7.

At the centre of the pipe, the $\theta$ direction is not defined. To avoid a singularity, we simply avoid placing any grid point at the centre. The resulting mesh near the centre looks as if a $v$ node were present at the centre. A sketch of such a set-up is shown in Figure 8. When discretizing the nodes

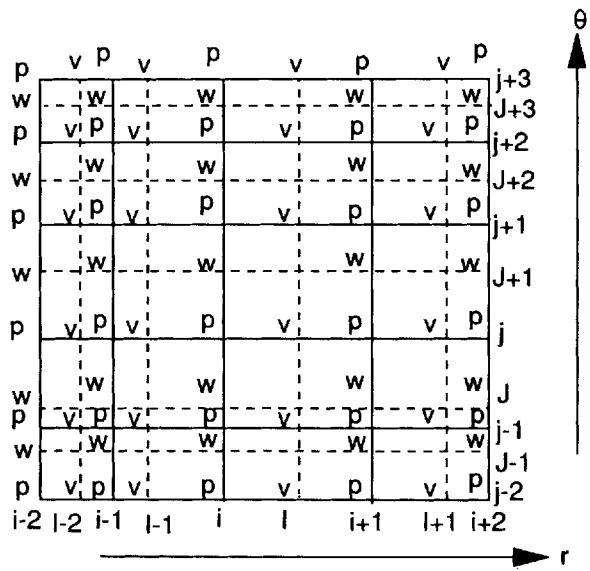


Figure 6. The mesh layout in the domain

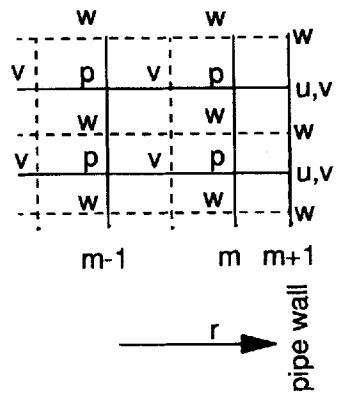


Figure 7. Grid layout near the wall

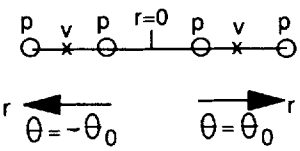


Figure 8. Grid arrangement near the centre

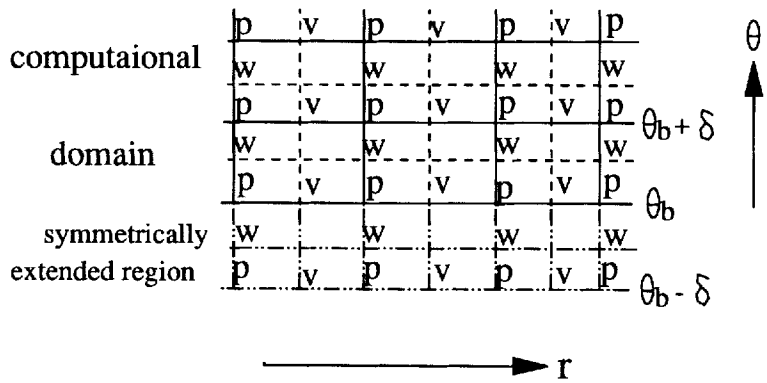


Figure 9. Mesh layout near a symmetrical pseudo-boundary

near the centre, the variables $r, v$ and $w$ are assigned opposite sign when referred from across the centre to the other side.

If we are to compute only half the domain, the horizontal line ($\theta = -\pi/2$ and $\theta = \pi/2$) is the breaking line to form a pseudo-boundary, where the flow is symmetrical. On such a symmetrical boundary, $(u, p)$ and $v$ grids are arranged regularly and the $w$ grid point is absent. The conservational domain is, however, symmetrically extended outside of the computational domain. A sketch of the mesh arrangement is shown in Figure 9. This set-up allows one to impose the following type of weak symmetrical conditions easily: $\phi_{\theta b+\delta} = \phi_{\theta b-\delta}$ and $W_{\theta b+\delta/2} = -w_{\theta b-\delta/2}$, where $\phi = u, v, p$. However, the condition of symmetry for $p$ need not be imposed in the numerical

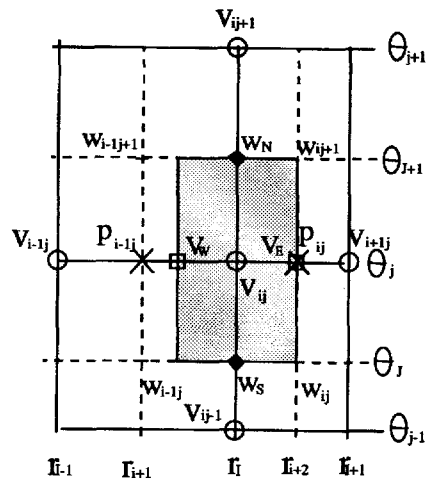


Figure 10. Conservational domain for node $v_{ij}$

computation. $\theta_b$ is the angle of the symmetry $-\pi/2$. $\delta$ is the angular incremental of grid lines across the symmetry line.

The global numbering of the nodes is shown in Figure 6. A sketch of the conservational domain set-up for a $v_{ij}$ node is shown in Figure 10.

*Formal discretization*

Since the governing equations can be written in the form of equation (3), we need only to identify the decoupling. To illustrate the proposed method, we take the $r$-momentum equation and discretize it at the node of $v_{ij}$ in an undistorted non-uniform mesh. A sketch of the conservational domain is shown in Figure 7.

The first step is to discretize in the axial direction by using a similar approach to that of Patankar and Spalding.[12] The final governing equation ($r$-momentum) after discretizing in the $s$-direction and separating $r$ and $\theta$ directions is given as

$$r\left(h_1 d_v+\frac{Re}{\Delta s}u^{\mathrm{U}}\right)v+E_{vr}+E_{v\theta}=\frac{1}{A_e}\int_{r_W}^{r_E}\int_{\theta_S}^{\theta_N}\left(h_1 S_v+\frac{Re}{\Delta s}u^{\mathrm{U}}v^{\mathrm{U}}\right)r\,\mathrm{d}r\,\mathrm{d}\theta, \tag{27a}$$

where

$$E_{vr}=\frac{1}{r_E-r_W}\int_{r_W}^{r_E}\left\{\frac{\partial}{\partial r}\left[rh_1\left(v^{\mathrm{U}}Re\,v-\frac{\partial v}{\partial r}\right)\right]-v\frac{\partial(rh_1v^{\mathrm{U}})}{\partial r}Re\right\}\mathrm{d}r \tag{27b}$$

and

$$E_{v\theta}=\frac{1}{\theta_N-\theta_S}\int_{\theta_S}^{\theta_N}\left[\frac{\partial}{\partial\theta}\left(h_1w^{\mathrm{U}}Re\,v-\frac{h_1}{r}\frac{\partial v}{\partial r}\right)-v\frac{\partial(h_1w^{\mathrm{U}})}{\partial\theta}Re\right]\mathrm{d}\theta. \tag{27c}$$

The superscript U stands for the upstream, $A_e=(r_E-r_W)(\theta_N-\theta_S)$, subscripts E and W denote the conservational domain boundaries along the $r$ direction for incrasing $r$ and decreasing $r$, respectively, subscripts N and S denote the conservational domain boundaries along the peripheral direction for increasing $\theta$ and decreasing $\theta$, respectively, $r_E=\frac{1}{2}(r_I+r_{I+1})$, $r_W=2r_I-r_E$,

$\theta_N=\theta_{J+1}$, $\theta_S=\theta_J$ and $\Delta_S$ is the axial step size. The evaluation of the right-hand side of equation (27a) is straightforward. Discretization of equations (27b) and (27c) is given in the Appendix.

The discretized $r$-momentum equation can be assembled from the individual one-dimensional subproblems to yield

$$\mathrm{VIJ}_{ij}v_{ij}+\mathrm{VMJ}_{ij}v_{i-1j}+\mathrm{VPJ}_{ij}v_{i+1j}+\mathrm{VIM}_{ij}v_{ij-1}+\mathrm{VIP}_{ij}v_{ij+1}=\mathrm{VRH}_{ij}-r_I h_{1Ij}\frac{p_{ij}-p_{i-1j}}{r_{i+2}-r_{i+1}}, \quad (28)$$

where $h_{1Ij}=1+\lambda r_I \sin\theta_j$ and the coefficient matrix and the right-hand side variables are defined in the Appendix.

*Pressure perturbation*

When the pressure field is perturbed by $\Delta p$, it leads to changes in the velocity field. To estimate the velocity change, we apply Gauss–Seidel iteration on the discretized momentum equations:

$$\Delta v_{ij}=-\frac{r_I h_{1Ii}}{\mathrm{VIJ}_{ij}}\frac{\Delta p_{i+1j}-\Delta p_{ij}}{r_{i+2}-r_{i+1}}, \quad (29)$$

$$\Delta w_{ij}=-\frac{h_{1iJ}}{\mathrm{WIJ}_{ij}}\frac{\Delta p_{ij}-\Delta p_{ij-1}}{\theta_j-\theta_{j-1}}. \quad (30)$$

Hence, the resulting perturbed velocity field is

$$v_{ij}^{\mathrm{new}}=v_{ij}+\Delta v_{ij},$$

$$w_{ij}^{\mathrm{new}}=w_{ij}+\Delta w_{ij},$$

By forcing the continuity equation to be satisfied, we can solve for the pressure perturbation, i.e.

$$\int\int\left[\frac{\partial}{\mathrm{d}r}(rh_1\,\Delta v)+\frac{\partial}{\partial\theta}(h_1\,\Delta w)\right]\mathrm{d}r\,\mathrm{d}\theta=-\int\int\left[r\frac{\partial u}{\partial s}+\frac{\partial(rh_1 v)}{dr}+\frac{\partial(h_1 w)}{\partial\theta}\right]\mathrm{d}r\,\mathrm{d}\theta. \quad (31)$$

With some under-relaxation, the above equation can be rendered to the standard 5-point module pressure perturbation equation after substituting in the velocity perturbation:

$$\mathrm{PIJ}_{ij}\Delta p_{ij}+\mathrm{PMJ}_{ij}\Delta p_{i-1j}+\mathrm{PPJ}_{ij}\Delta p_{i+1j}+\mathrm{PIM}_{ij}\Delta p_{ij-1}+\mathrm{PIP}_{ij}\Delta p_{ij+1}=F_p\mathrm{PRH}_{ij}, \quad (32)$$

where

$$\mathrm{PIJ}_{ij}=-\mathrm{PMJ}_{ij}-\mathrm{PPJ}_{ij}-\mathrm{PIM}_{ij}-\mathrm{PIP}_{ij}, \qquad \mathrm{PMJ}_{ij}=-\frac{(r_{I-1}h_{1I-1j})^2}{(r_I-r_{I-1})(r_i-r_{i-1})\mathrm{VIJ}_{i-1j}},$$

$$\mathrm{PPJ}_{ij}=-\frac{r_I^2 h_{1Ij}^2}{(r_I-r_{I-1})(r_{i+1}-r_i)\mathrm{VIJ}_{ij}}, \qquad \mathrm{PIM}_{ij}=-\frac{h_{1iJ}^2}{(\theta_{J+1}-\theta_J)(\theta_j-\theta_{j-1})\mathrm{WIJ}_{ij}},$$

$$\mathrm{PIP}_{ij}=-\frac{h_{1iJ+1}^2}{(\theta_{J+1}-\theta_J)(\theta_{j+1}-\theta_j)\mathrm{WIJ}_{ij+1}}, \qquad \mathrm{PRH}_{ij}=\mathrm{FLOWIN}_{ij}-\mathrm{FLOWOUT}_{ij},$$

$$\mathrm{FLOWIN}_{ij}=\frac{r_{I-1}h_{1I-1j}v_{i-1j}}{r_I-r_{I-1}}+\frac{h_{1iJ}w_{ij}}{\theta_{J+1}-\theta_J}+\frac{r_i u_{ij}^U}{\Delta s}, \quad \mathrm{FLOWOUT}_{ij}=\frac{r_I h_{1Ij}v_{ij}}{r_I-r_{I-1}}+\frac{h_{1iJ+1}w_{ij+1}}{\theta_{J+1}-\theta_J}+\frac{r_i u_{ij}}{\Delta s}.$$

when a good initial guess in the velocity field is used, the pressure perturbation alone is not the best approach to take for the pressure evaluation.[4] Following the same method as that of SIMPLER,[4] we can solve for the pressure directly based on the known velocity field.

Although the pressure is solved by using the above strategy, SIMPLER-type pressure iteration is not recommended. Without the pressure perturbation, the scheme would be like a plain iteration switching between various variables. Such an iteration scheme is usually not favourable. Instead, the SIMPLER strategy for the pressure evaluation is used only once to initiate the iteration for a few marching steps at the start or after we disturbed the velocity field substantially.

*Termination of iteration*

The iteration procedure for the momentum and pressure equations was repeated to a relative tolerance of $10^{-6}$. Hence, the momentum equations are satisfied as long as the pressure field is correct.

In order to have a correct pressure field, the velocity field must satisfy the continuity equation. This gives rise to a stopping criterion that is related to the continuity equation.

The relative error in the continuity equation is given by

$$\mathrm{RSD} = \frac{\sum_{i,j} \dfrac{2|\mathrm{PRH}_{ij}|}{|\mathrm{FLOWIN}_{ij}| + |\mathrm{FLOWOUT}_{ij}|}}{\sum_{i,j} 1},$$

where $\mathrm{PRH}_{ij}$ is the error in the flow in and out of the conservational domain of $p_{ii}$ and $\mathrm{FLOWIN}_{ij}$ is the flow into the conservational domain. In this example, we set the stopping criterion to be $\mathrm{RSD} < 10^{-5}$.

On close look at the definition of RSD, one can observe that RSD can be considered as an estimate of the $L_1$ norm of the normalized continuity equation error distribution.

*Computational results*

Table I shows the grid dependence and computational time required for the fluid flow development from a flat flow profile to the primary two-vortex solution with a NeXT station for 1200 axial steps.

The grid notation used is given as $aM \times Nbc$, where $a = n$ stands for the non-uniform mesh in the radial direction, $a = u$ stands for the uniform mesh in the radial direction, $M$ is the number of grid points in the radial direction, $N$ is the number of grid points in the peripheral direction, $b = \mathrm{f}$ stands for the full domain formulation (no symmetry is assumed), $b = \mathrm{h}$ stands for a half domain formulation (the symmetry is imposed), $c = \mathrm{n}$ stands for the non-uniform mesh in the peripheral direction and $c = \mathrm{u}$ stands for the uniform mesh in the peripheral direction. A typical mesh n16 × 14hn is shown in Figure 11.

In Figure 11, the mesh lines parallel to the peripheral direction are the perpendicular mesh lines of variable $v$, and the lines parallel to the radial direction are the perpendicular mesh lines of variable $w$. The cells formed by the mesh lines are the conservational domains for the variables of

Table I. Grid test for $Re = 1723$, $\lambda = 1/30$

| Grid | u10 × 20fu | n10 × 20fu | n15 × 24fu | n20 × 28fu | n25 × 32fu | n30 × 40fu | n40 × 48fu |
|---|---|---|---|---|---|---|---|
| f*Re* | 38·30 | 37·42 | 35·82 | 36·38 | 36·10 | 36·24 | 36·19 |
| cpu, s | 3161·4 | 2954·9 | 6506.9 | 9646·8 | 19282·1 | 43150 | 123530* |

Note that the literature value found in f$Re = 37{\cdot}15$ by Tarbell and Samuels.[20]

* 2000 axial steps are used for this case.

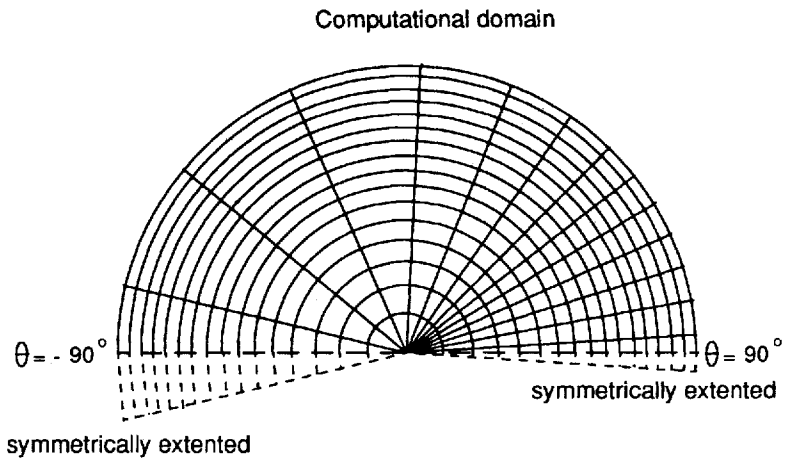


Figure 11. Mesh layout for n16 × 14hn

Table II. Grid test for $Re = 1928{\cdot}5$, $\lambda = 0{\cdot}01$, i.e. $Dn = 192{\cdot}85$

| Grid | n10 × 12hu | n10 × 12hn | n16 × 14hn | n20 × 16hn | n20 × 20hn | n30 × 25hn | n30 × hn |
|---|---|---|---|---|---|---|---|
| f*Re* | 30·64 | 29·10 | 29·35 | 29·50 | 29·40 | 29·36 | 29·36 |
| cpu, s | 4563·7 | 6472·1 | 10936·3 | 17049·2 | 20946·8 | 49123·2 | 61480·0 |

| Literature | Grid | f*Re* | *Dn* | *λ* |
|---|---|---|---|---|
| Dennis and Ng[21] | 60 × 30 | 29·33 | 192·9 | 0·0 |
| Yang and Keller[22] | 60 × 20 | 29·34 | 192·8 | 0·0 |

$u$ and $p$. The centre of the pipe has no grid points. $\theta = -\pi/2$ and $\theta = \pi/2$ lines are the computational domain boundaries and not the perpendicular mesh lines for $w$.

We chose $\lambda = 0{\cdot}01$ to generate some solutions in order to compare with the literature where loose-coiling approximation was employed.

Table II shows the grid dependence and the computational time required for a fully developed flow. The literature data for a loose coiling are also included in Table II.

From Tables I and II, one can observe that the proposed scheme converges fairly well. The grid of n16 × 14hn is adequate for the 4-vortex solution of Dn = 192.85, whereas, in the literature, a very large number of grid points were required. The solution from a grid of n10 × 12hn is also within 1% of that obtained using a fine grid.

The solutions were found not to be grid-sensitive once the grid size is sufficiently small, as is shown in both Tables I and II.

Figure 12 shows the secondary flow pattern of the 4-vortex solution used in Table II. The numbers denote the $\psi$ values of the secondary flow stream function as defined by equation (26).

## NUMERICAL EXAMPLE 2: SQUARE-DRIVEN CAVITY FLOW

In the first example, we have shown the implementation of the proposed method. To show more clearly the convergence rate and the advantage of the 3-point exponential scheme over 2-point

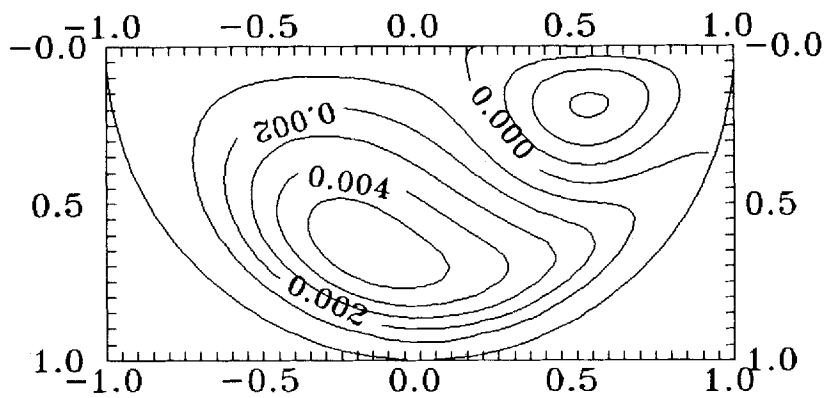


Figure 12. Four-vortex secondary flow pattern for $Dn=192{\cdot}85$ and $\lambda=0{\cdot}01$ for half of the flow domain

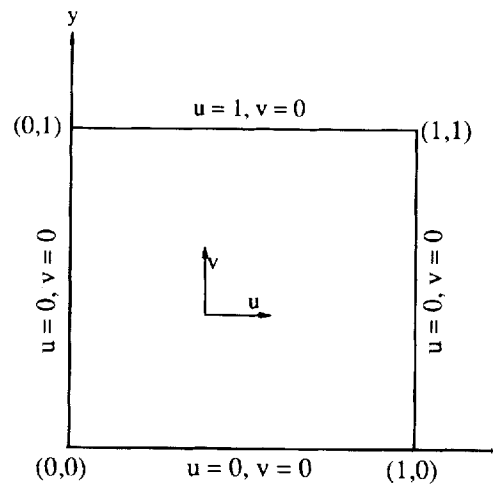


Figure 13. Square-driven cavity

schemes and non-exponential schemes, we choose the problem of a square-driven cavity flow. The system set-up and the boundary conditions are shown in Figure 13. The velocity components are normalized by the velocity $U_0$ of the moving plate (located at the topmost, i.e. $y=1$), i.e. $u=u'/U_0$ and $v=v'/U_0$. The geometrical variables are normalized by the side length of the square, $L$, i.e. $x=x'/L$ and $y=y'/L$.

The governing equations are

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0,$$

$$Re\left(\frac{\partial uu}{\partial x}+\frac{\partial vu}{\partial y}\right)-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}=-\frac{\partial p}{\partial x}, \tag{33}$$

$$Re\left(\frac{\partial uv}{\partial x}+\frac{\partial vv}{\partial y}\right)-\frac{\partial^2 v}{\partial x^2}-\frac{\partial^2 v}{\partial y^2}=-\frac{\partial p}{\partial y},$$

where $Re=LU_0/\nu$ and $p=Re\,p'/(4\rho U_0^2)$.

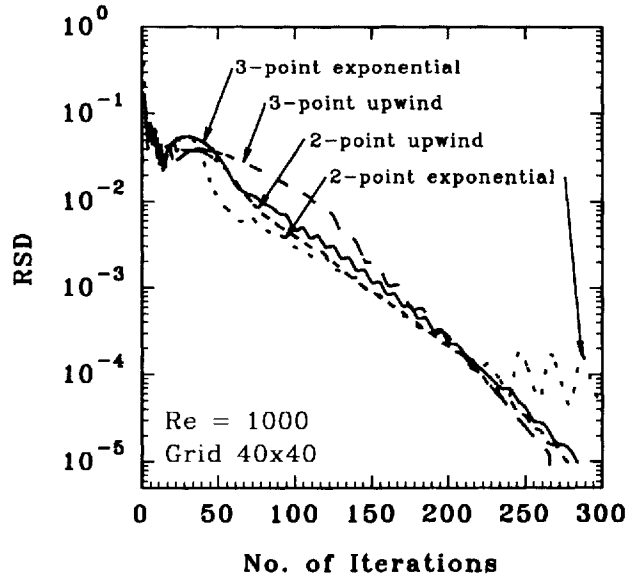


Figure 14. The convergence rate with the proposed method

Table III. The cpu time requirement for various schemes and meshes

| *Re* | Scheme | Mesh | ITER | RSD | cpu, s |
|---|---|---|---|---|---|
| 1000 | 2-p upwind | $40 \times 40$ | 278 | $1{\cdot}0 \times 10^{-5}$ | 5193·9 |
| 1000 | 2-p exponential | $40 \times 40$ | 300 | $4{\cdot}4 \times 10^{-5}$ | 7238·7 |
| 1000 | 3-p upwind | $40 \times 40$ | 266 | $9{\cdot}1 \times 10^{-6}$ | 6418·8 |
| 1000 | 3-p exponential | $40 \times 40$ | 284 | $9{\cdot}6 \times 10^{-6}$ | 6528·8 |
| 1000 | 3-p exponential | $20 \times 20$ | 69 | $9{\cdot}7 \times 10^{-6}$ | 550·6 |
| 1000 | 3-p exponential | $80 \times 80$ | 765 | $7{\cdot}9 \times 10^{-6}$ | 111784·8 |

ITER stands for the number of iterations performed.

Although a good approach to this problem would be to use a non-uniform mesh, a unform mesh was used. The initial guess of zero is used for all variables in all cases.

Figure 14 shows the convergence rate of using different interpolation functions. It can be seen that the 3-point interpolation schemes do not differ much from the 2-point schemes in terms of convergence rate. However, the 2-point exponential scheme did not reduce RSD beyond $10^{-4}$ and, consequently, showed difficulty at very low RSD.

Table III shows the cpu time requirement of the various schemes by a NeXT machine. The use of exponential functions does not increase the cpu time dramatically. A small increase of less than 2% is observed for the case of the 3-point exponential over the 3-point upwind scheme.

Figure 15 shows the velocity profiles using uniform grids of $80 \times 80$ and $140 \times 140$ with a 3-point exponential interpolation function. Results from Schreiber and Keller[19] are also included for comparison. The agreement is fairly good between the present solution and that presented by Schreiber and Keller.

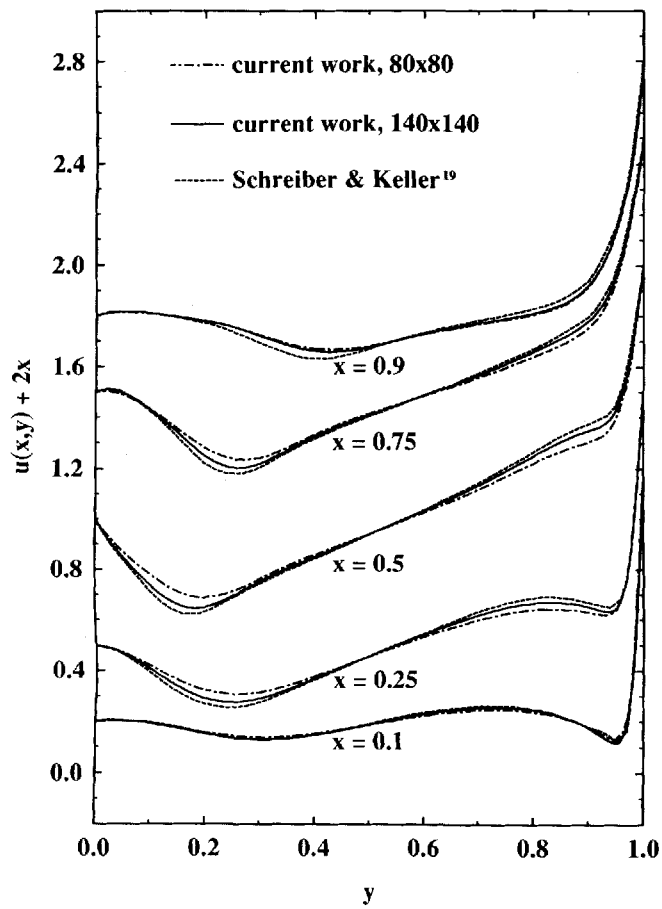


Figure 15. Comparison of $u$-velocity profiles for $Re = 1000$. Data from Schreiber and Keller[19] are extrapolated results based on a few mesh sizes including a mesh of $141 \times 141$ and using a stream function formulation

## CONCLUSIONS

A first-order non-conforming numerical scheme for fluid flow with a 3-point exponential interpolation has been developed. The multidimensional problem is first decoupled into multiple one-dimensional subproblems upon discretizing. After assembling the multiple one-dimensional subproblems to form the algebraic equations for the multidimensional flow problem, the resulting algebraic equations are solved iteratively using a tri-diagonal solver. The velocity and pressure are decoupled at each iteration using the pressure correction method due to Patankar.[4]

The accuracy of the traditional upwind/exponentially weighted schemes has been investigated and is compared with the proposed 3-point scheme for both interpolation and consistency. It is found that the proposed 3-point exponential interpolation scheme together with the multidimensional separating strategy render a guaranteed first-order convergence.

The computational results for laminar fluid flows in a torus and in a square-driven cavity show good agreement with the literature. The convergence rate is similar to the traditional upwind schemes. The use of exponential functions does not affect the computational time.

## APPENDIX

Discretization of the separated $r$-directional $r$-momentum component can be illustrated as follows:

$$(r_E - r_W)E_{vr} = \left[ rh_1 \left( v^U Re\, v - \frac{\partial v}{\partial r} \right) \right]_{r=r_E} - \left[ rh_1 \left( v^U Re\, v - \frac{\partial v}{\partial r} \right) \right]_{r=r_W} - v_{ij}(r_E h_{1E} v_E - r_W h_{1W} v_W) Re,$$

where

$$v_E = v^U|_{r=r_E} = \frac{v^U_{i+1\,j}(r_E - r_I) + v^U_{ij}(r_{I+1} - r_E)}{r_{I+1} - r_I}$$

$$v_W = v^U|_{r=r_W} = \frac{v^U_{ij}(r_I - r_W) + v^U_{i-1\,j}(r_W - r_{I-1})}{r_I - r_{I-1}}.$$

The subscripts E and W are defined in the paper following equation (27).

The momentum fluxes at the conservational surface can further be written as follows:

$$\left[ rh_1 \left( v^U Re\, v - \frac{\partial v}{\partial r} \right) \right]_{r=r_E} = \alpha_E v_{i-1\,j} + (r_E h_{1E} v_E Re - \alpha_E - \beta_E) v_{ij} + \beta_E v_{i+1\,j},$$

$$\left[ rh_1 \left( v^U Re\, v - \frac{\partial v}{\partial r} \right) \right]_{r=r_W} = \alpha_W v_{i-1\,j} + (r_W h_{1W} v_W Re - \alpha_W - \beta_W) v_{ij} + \beta_W v_{i+1\,j}.$$

Hence, the discretized one-dimensional module can be written as

$$(r_E - r_W)E_{vr} = (r_E h_{1E}\alpha_E - r_W h_{1W}\alpha_W)(v_{i-1\,j} - v_{ij}) + r_E h_{1E}\beta_E - r_W h_{1W}\beta_W)(v_{i+1\,j} - v_{ij}), \tag{34}$$

where $\alpha$ and $\beta$ are obtained by using the interpolation equation (8). In this case, $H_1 = r_I - r_{I-1}$, $H_2 = r_{I+1} - r_I$ and $Re\, u_i = v^U_{ij} Re$. Here the use of upper-case $H$ is to avoid any confusion with the metric coefficients. To simplify the notations, we use

$$a = v^U Re, \qquad b = Re\, u_i, \qquad H = r_E - r_I = r_I - r_W.$$

Application of the 3-point exponential interpolation equation (8) leads to

$$\begin{aligned}\left[ rh_1 \left( av - \frac{\partial v}{\partial r} \right) \right]_{r=r_E} = {} & a_E v_{ij} + \frac{H_1(e^{bH} - 1) + H(e^{-bH_1} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)} a_E (v_{i+1\,j} - v_{ij}) \\ & + \frac{H_2(e^{-bH} - 1) + H(e^{bH_2} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)} a_E (v_{i+1\,j} - v_{ij}) \\ & - \frac{bH_1 e^{bH} + e^{-bH_1} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)} (v_{i+1\,j} - v_{ij}) \\ & - \frac{bH_2 e^{-bH} - e^{bH_2} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)} (v_{i+1\,j} - v_{ij}),\end{aligned}$$

Hence, one can obtain

$$\alpha_E = \frac{(a_E + b)H_2(e^{-bH} - 1) + bH_2 - (a_E H - 1)(e^{bH_2} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)}, \tag{35a}$$

$$\beta_E = \frac{(a_E - b)H_1(e^{bH} - 1) - bH_1 + (a_E H - 1)(e^{-bH_1} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)}. \tag{35b}$$

similarly,

$$\alpha_W = \frac{(a_W + b)H_2(e^{-bH} - 1) + bH_2 - (a_W H - 1)(e^{bH_2} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)}, \tag{35c}$$

$$\beta_W = \frac{(a_W + b)H_1(e^{bH} - 1) - bH_1 + (a_W H - 1)(e^{-bH_1} - 1)}{H_1(e^{bH_2} - 1) + H_2(e^{-bH_1} - 1)}. \tag{35d}$$

In the above expressions for $\alpha$ and $\beta$ we have taken care of the large Reynolds number flow limit. When $b \to 0$, i.e. almost creeping flow, computationally, equation (35) is of 0/0 type. In order to avoid this numeric difficulty in the computation, we take the limit of $b \to 0$ in equation (35) to obtain the following expressions for small Reynolds number flows:

$$\alpha_E = \frac{(a_E H - 1)(H - H_2) - H}{H_1(H_1 + H_2)}, \tag{36a}$$

$$\beta_E = \frac{(a_E H - 1)(H + H_1) - H}{H_1(H_1 + H_2)}, \tag{36b}$$

$$\alpha_W = \frac{(a_W H + 1)(H + H_2) + H}{H_1(H_1 + H_2)}, \tag{36c}$$

$$\beta_W = \frac{(a_E H + 1)(H - H_1) + H}{H_1(H_1 + H_2)}, \tag{36d}$$

Equation (36) can also be obtained by applying the quadratic interpolation function, equation (9), instead of the exponential interpolation function, equation (8). In practice, we can use equation (36) when the cell Peclet number $bH_1$ and $bH_2 < 2$.

One can discretize the other direction in the same manner. The final discretized $r$-momentum is shown by equation (29), where the coefficients are defined as follows:

$$\mathrm{VRH}_{ij} = Re(\lambda r_I \sin\theta_j U_C^2 + h_{1Ij} W_C^2) - \frac{2\lambda r_I \sin\theta_j}{h_{1Ij}} \frac{U_C - U_C^U}{\Delta s} - \frac{2}{r_I} \frac{W_N - W_S}{\theta_{J+1} - \theta_J} - \left(2 - \frac{1}{h_{1Ij}}\right) \lambda \cos\theta_j W_C,$$

$$\mathrm{VMJ}_{ij} = \frac{r_E h_{1E} \alpha_E - r_W h_{1W} \alpha_W}{r_E - r_W}, \qquad \mathrm{VPJ}_{ij} = \frac{r_E h_{1E} \beta_E - r_W h_{1W} \beta_W}{r_E - r_W},$$

$$\mathrm{VIJ}_{ij} = \frac{1 + 2h_{1Ij}\lambda r_I \sin\theta_j}{r_I h_{1Ij}} - \mathrm{VMJ}_{ij} - \mathrm{VPJ}_{ij} - \mathrm{VIM}_{ij} - \mathrm{VIP}_{ij}, \quad W_C = \frac{W_N + W_S}{2},$$

$$W_N = \frac{w_{ij+1}(r_I - r_{i+1}) + w_{i-1j+1}(r_{i+2} - r_I)}{r_{i+2} - r_{i+1}}, \qquad W_S = \frac{w_{ij}(r_I - r_{i+1}) + w_{i-1j}(r_{i+2} - r_I)}{r_{i+2} - r_{i+1}},$$

$$U_C = \frac{u_{ij}(r_I - r_{i+1}) + u_{i-1j}(r_{i+2} - r_I)}{r_{i+2} - r_{i+1}}, \qquad U_C^U = \frac{u_{ij}^U(r_I - r_{i+1}) + u_{i-1j}^U(r_{i+2} - r_I)}{r_{i+2} - r_{i+1}},$$

and $\mathrm{VIP}_{ij}$ and $\mathrm{VIM}_{ij}$ are similar to $\mathrm{VMJ}_{ij}$ and $\mathrm{VPJ}_{ij}$.

## REFERENCES


1. I. Babuška, 'The finite element method with Lagrange multipliers', *Numer. math.*, **20**, 179–192 (1973).
2. F. Brezzi, 'On the existence, uniqueness and application of saddle point problems arising from Lagrangian multipliers', *RAIRO*, **8**, 129–151 (1974).
3. O. A. Ladyzhenskaya, *Mathematical Theory of Viscous Incompressible flow*, Gordon and Breach, New York, 1969.

4. S. V. Patankar, *Numerical Heat Transfer and Fluid flow*, Hemisphere, New York, 1980.
5. G. Carey and J. T. Oden, *Finite Elements: Fluid Mechanics, Vol. VI*, Prentice-Hall Canada, Toronto, 1986.
6. G. D. Raithby and K. E. Torrance, 'Upstream weighted differencing schemes and their applications to elliptic problems involving fluid flows', *Comput. Fluids*, **2**, 191–206 (1974).
7. J. W. Macharthur and S. V. Patankar, 'Robust semidirect finite difference methods for solving the Navier–Stokes and energy equations', *Int. j. numer. methods fluids*, **4**, 517–537 (1984).
8. F. H. Harlow and J. E. Welch, 'Numerical calculation of time-dependent viscous incompressible flow of fluids with free surface', *Phys. Fluids*, **8**, 2182–2189 (1965).
9. B. P. Leonard 'Locally modified QUICK scheme for highly convective 2-D and 3-D flows', in C. Taylor, U. G. Habashi and M. M. Haffez (eds), *Numerical Methods in Laminar and Turbulent flow, Vol. 5*, Pineridge Press, Swansea, 1987, pp. 35–47.
10. K. Nandakumar, J. H. Masliyah and H. Law, 'Bifurcation in steady laminar mixed convection flow in horizontal ducts', *J. Fluid Mech.*, **152**, 145–161 (1985).
11. V. D. de Henau, G. D. Raithby and B. E. Thompson, 'A total pressure correction for upstream weighted schemes', *Int. j. numer. methods fluids*, **9**, 855–864 (1989).
12. 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*, **15**, 1787–1806 (1972).
13. A. W. Neuberger, 'Error estimates and convergence acceleration of different discretization schemes' in C. Taylor, W. G. Habashi and M. M. Haffez, (eds), *Numerical Methods in Laminar and Turbulent Flow*, Vol. 5, Pineridge Press, Swansea, 1987,pp. 91–101.
14. D. N. Arnold, F. Brezzi and M. Fortin, 'A stable finite element for the Stokes equations', *Calcolo*, **21**, 337–344 (1984).
15. S. Liu, 'Laminar flow and heat transfer in helical pipes with finite pitch', *Ph.D. Dissertation*, University of Alberta, Edmonton, 1992.
16. J. H. Masliyah and K. Nandakumar, 'Fully developed viscous flow and heat transfer in curved semicircular sectors', *AIChE J.*, **25**, 478–487 (1979).
17. K.Nandakumar, J. H. Masliyah, 'Bifurcation in steady laminar flow through curved tubes', *J. Fluid Mech.*, **119**, 475–490 (1982).
18. S. R. Sankar, K. Nandakumar and J. H. Masliyah, 'Oscillatory flows in coiled square ducts', *Phys. Fluids*, **31**, 1348–1358 (1988).
19. R. Schreiber and H. B. Keller, 'Driven cavity flows by efficient numerical techniques', *J. Comput. Phys.*, **49**, 310–333 (1983).
20. J. M. Tarbell and M. R. Samuels, 'Momentum and heat transfer in helical coils', *Chem. Eng. J.*, **5**, 117–127 (1972).
21. S. C. R. Dennis and M. Ng, 'Dual solutions for steady laminar flow through a curved tube', *Quart. J. Mech. Appl. Math.*, **35**, 305–324 (1982).
22. Z. Yang and H. B. Keller, 'Multiple laminar flows through curved pipes', *Appl. Numer. Math.*, **2**, 257–271 (1986).