INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **57**:205–236
Published online 25 October 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1632

Accelerated convergence of the numerical simulation of incompressible flow in general curvilinear co-ordinates by discretizations on the double-staggered grids

A. Shklyar*,†,‡ and A. Arbel

Institute of Agricultural Engineering, Agricultural Research Organization, The Volcani Center, P.O. Box 6, Bet Dagan 50250, Israel

SUMMARY

The convergence rate of a methodology for solving incompressible flow in general curvilinear co-ordinates is analyzed. Double-staggered grids (DSGs), each defined by the same boundaries as the physical domain, are used for discretization. Both grids are MAC quadrilateral meshes with scalar variables (pressure, temperature, etc.) arranged at the center and the Cartesian velocity components at the middle of the sides of the mesh cells. The problem was checked against benchmark solutions of natural convection in a squeezed cavity, heat transfer in concentric horizontal cylindrical annuli, and a hot cylinder in a duct.

Poisson's pressure-correction equations that arise from the SIMPLE-like procedure are solved by several methods: successive overrelaxation, symmetric overrelaxation, modified incomplete factorization preconditioner, conjugate gradient (CG), and CG with preconditioner. A genetic algorithm was developed to solve problems of numerical optimization of SIMPLE-like calculation time in a space of iteration numbers and relaxation parameters. The application provides a means of making an unbiased comparison between the DSGs method and the widely used interpolation method. Furthermore, the convergence rate was demonstrated by application to the calculation of natural convection heat transfer in concentric horizontal cylindrical annuli. Calculation times when DSGs were used were 2–10 times shorter than those achieved by interpolation. With the DSGs method, calculation time increases slightly with increasing non-orthogonality of the grids, whereas an interpolation method calls for very small iteration parameters that lead to unacceptable calculation times. Copyright © 2007 John Wiley & Sons, Ltd.

Received 13 May 2007; Revised 4 September 2007; Accepted 4 September 2007

KEY WORDS: general curvilinear co-ordinates; incompressible flow; natural convection; convergence

*Correspondence to: A. Shklyar, Institute of Agricultural Engineering, Agricultural Research Organization, The Volcani Center, P.O. Box 6, Bet Dagan 50250, Israel.
†E-mail: shklyar@agri.gov.il
‡Scientist.

Contract/grant sponsor: Israeli Ministry of Agriculture and Rural Development
Contract/grant sponsor: Israeli Ministry of Energy

1. INTRODUCTION

In the previous study [1]we considered in some detail the discretization of the momentum–mass equations of the incompressible flow in general curvilinear co-ordinates on double-staggered grids (DSGs), each defined by the same boundaries as the physical domain.

The use of DSG does not restrict the applications of any one of the methods for solving incompressible flows. In developing new methods or algorithms for general co-ordinates, it is desirable to retain as many as possible the advantages and, as far as possible, the simplicity of existing widely used algorithms for Cartesian co-ordinates, e.g. the artificial compressibility approach [2], the MAC method [3], the pressure projection method [4], and SIMPLE-like methods [5, 6]. The adopted overall computational procedure in [1] is an extension of the SIMPLE algorithm [5]. The procedure is iterative to account for the coupling of velocity and pressure and for the non-linearity in the momentum equations. Application of the methodology was illustrated in relation to well-known external and internal flow problems: viscous flow over a circular cylinder with Reynolds numbers ranging from 10 to 40, and lid-driven flow in a cavity with inclined walls. The DSG method has been developed around the structure of conservation equations in general co-ordinates. The gradient term of any scalar in general co-ordinates (Figure 1) is

$$\varphi_x = (\varphi_\xi - \varphi_\eta)/J \tag{1}$$

A distinguishing feature of the gradient term in general co-ordinates is the existence of two terms. Cross-derivatives (such as p_ξ, p_η) appear in curvilinear co-ordinates and thus couple closely with the arrangement of the Cartesian velocity components and scalar values as pressure, temperature, etc. Gradient component or values of variables in curvilinear co-ordinates at n-point of the cell can be defined as (taking the pressure derivative as an example Figure 1(a))

$$p^1_\xi|_n \approx p^1_{ne} - p^1_{nw} \tag{2}$$

Approximation of $p^1_\xi|_n$ has been the subject of very many studies; throughout this paper, we will use the term 'standard interpolation (SI)' method [7] to mean any interpolation of the scalar variables. The effect of the arrangement of the velocity and scalar values and of handling with cross-derivative terms, on the convergence characteristic of Poisson's equation for pressure in the pressure projection method, or those of the pressure-correction equation in the SIMPLE-like methods is discussed below. Harlow and Welch [3] describe a scheme for regular Cartesian meshes, in which the pressure is located at cell centers but the velocity is distributed over the cell faces with horizontal velocity components prescribed at vertical faces and vertical velocity components prescribed at horizontal faces (Figure 1(a)). Faghri *et al*. [8] and Shyy [9] adopted this type of staggered grid for a body-fitted co-ordinate system. The scalar variables are located at the arithmetic center of the four adjacent grids. Both u and U are located at the midpoints of the east and west faces of the control volume. Both v and V are located at the midpoints of the north and south faces of the control volume (Figure 1(c)). Since the successive line under-relaxation method was used to solve the system of finite-difference equations by applying the efficient tri-diagonal equation solver, the p_ξ, or p_η, is dropped. An alternative formulation Shyy *et al*. [10] too eliminated p_ξ, or p_η.

Vanka *et al*. [11] located the Cartesian velocity components, densities, and enthalpies at the intersection of grid lines (grid nodes), and the pressures at the centers of the cells formed by the grid lines. This staggered mesh system enables convenient computation of pressure gradient terms

DOI: 10.1002/fld

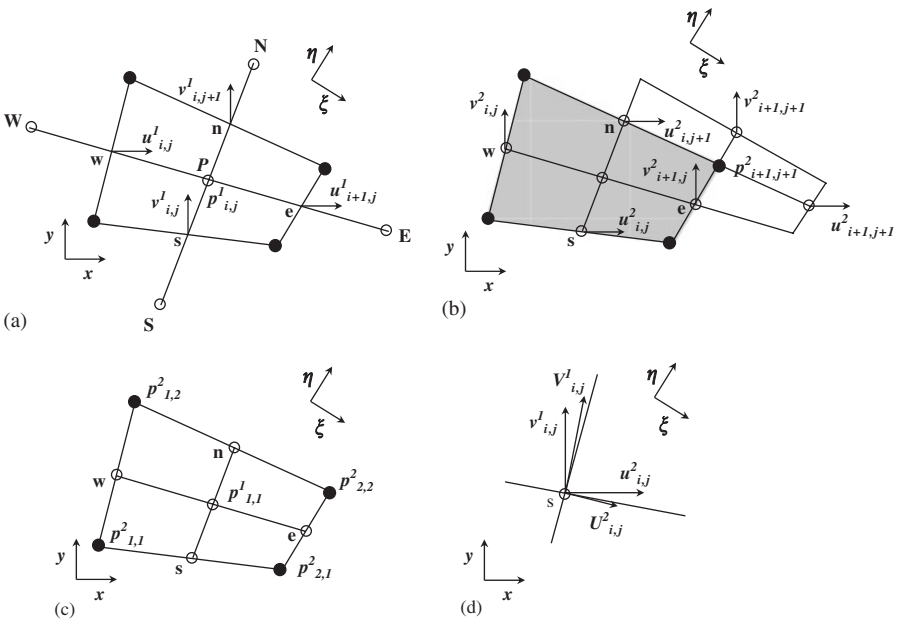

Figure 1. Double set of computational grids: (a) first staggered grid cell; (b) second staggered grid displacement; (c) second pressure cell displacement; and (d) a grid for which Cartesian and contravariant velocities are not aligned.

in the momentum equations. Two averaging processes are used to discretize the momentum and continuity equations: pressures are averaged to obtain gradients for driving velocities and velocities are averaged to obtain fluxes at the faces of the cell. This averaging process leads to the poorly convergent ADI procedure, which necessitates use of an additional iteration procedure, because of the appearance of the second pressure gradient. In the arbitrary Lagrangian–Eulerian method [12, 13], the Cartesian flow velocity components and pressures are located at the corners and in the center, respectively, of the cell. This curvilinear 'half-staggered mesh' [14] is used for numerical simulation of unsteady incompressible flow George *et al*. [15] and Huang [16]. In this study, we operate with Cartesian components of velocities. At this point mention should be made of another set of dependent variables and its arrangements. Contravariant velocity components [17–19], contravariant physical component [20, 21], covariant physical velocity components, contravariant flux component [22–24], and contravariant volume flux component [25–30] could be chosen as the dependent variables. The merit of the covariant (physical) velocity component approach is that is uses a simple form of Poisson's equation of the pressure correction. In the grid arrangement by Karki [31], one covariant physical velocity component is stored at each face of the control volume. The curvature source term arises in the momentum equation because the velocity components do not have a fixed direction. In the application of this method to calculate a turbulent flow in the supersonic ejector, Shklyar *et al*. [32] showed that weak oblique shock waves are well predicted in a smooth grid, but that the algorithm loses the convergence rate and stability for a non-smooth grid [33]. The staggered mesh method of Harlow and Welch was generalized to unstructured (triangular) meshes (e.g. Hall *et al*. [34], Nicolaides [35]). These 'dual-mesh' or 'covolume' methods take explicit advantage of the fact that every unstructured tetrahedral or triangular mesh

DOI: 10.1002/fld

(a Delaunay mesh) has an orthogonal or dual mesh associated with it (a Voronoi tessellation). In a two-dimensional case other means of storage location and grid arrangements lead to the nine-diagonal pressure equation (or pressure correction). In a three-dimensional case, they lead to the 19-diagonal pressure equation (or pressure correction) and lack diagonal dominance, which may cause convergence difficulties. Perić [36] studied the property of the pressure correction equation in a non-orthogonal system and found, for two-dimensional cavity flows, that if the grid were significantly non-orthogonal, the simplified (averaging process) pressure-correction equation did not converge at all or the convergence rate was too small. Cho *et al*. [37] proposed a decomposition of the non-orthogonal parts in the full pressure-correction equation, in order to ensure the convergence of numerical solutions in a strongly non-orthogonal grid. Also Wang and Komori [38, 39] concluded that the treatment of the cross-derivatives in the pressure-correction equation had a quite important influence on the convergence rate in computing fluid flows on strongly non-orthogonal grids. All the arrangement of the scalar variables and velocity vector for the general co-ordinates covered in the above brief review share a common property: numbers of momentum and continuity equations in the discretized forms match the initial numbers of momentum and continuity equations. This is accomplished by eliminating the 'inconvenient' terms, interpolating them or expanding them by means of Taylor series, etc.

Maliska and Raithby [40] developed an algorithm with two sets of discretized momentum equations, which may be thought of as splitting discretized equations, but with a common pressure (pressure-correction) equation. Both the Cartesian velocity components were stored coincidentally at the middle of the cell faces and pressure was stored at the center of the staggered grid cell (Figure 1(b)). It was demonstrated that the grid layout dictated the number of points involved in the pressure equations and, importantly, it was responsible for the type of linkage between the pressure at a central point and its neighboring pressures. The one Poisson pressure equation involves nine points and two momentum equations must be solved. The advantage of the method lies in the tight coupling between the velocity and the pressure fields, which leads to rapid convergence of the equation set.

Shklyar and Arbel [1] had inputted additional momentum equation and continuity in DSGs (variables marked with superscript 2 in Figure 1) and developed an algorithm with a system of paired sets of discretized equation system, each with two momentum and one pressure correction. The DSG method can be viewed as one of the problem-solving tools for approximating 'cross-diffusion' terms in general curvilinear co-ordinates (p_ξ, p_η etc.). In terms of DSG (Figure 1(c)), Equation (2) may be rewritten as

$$p^1_\xi|_n \approx (p^2_{1,2} - p^2_{2,2})$$

The problem of the influence of the (p_ξ, p_η) and its approximation have been a focus of attention during two decades of the creation of effective algorithms in a general curvilinear co-ordinates; Lehnhäuser and Schäfer [41] used an approximation of the pressure derivatives based on a multi-dimensional Taylor expansion; their paper includes examples of pressure-correction equation problems.

The objective of the present study was to make a numerical comparison between the convergence rates of the DSG method and of the SI method.

This paper is organized as follows. In Section 2 we begin with a brief description of the treatment of physical domain discretization on DSGs. In Section 3, the DSG methodology is checked against natural convection of the benchmark numerical solutions and experimental results for heat transfer in annuli. In Section 4, the performance of the DSG is illustrated by application to the well-known

DOI: 10.1002/fld

problem of natural convection heat transfer in concentric and eccentric horizontal cylindrical annuli by means of orthogonal and non-orthogonal grids.

2. PHYSICAL DOMAIN DISCRETIZATION

With regard to the DSG method, the reader may find in [1] the treatment of physical domain discretization with evaluation of metrics, discretization continuity and momentum equations, pressure boundary conditions, velocity–pressure linkage, which lead to generalization of the SIMPLE procedure for external and internal problems.

To facilitate the derivation of the finite-difference formulation, the solution domain in the x and y co-ordinate system is first discretized. Thus, we want to use the staggered grid technique, which offers conservation of mass, momentum, and kinetic energy naturally, and which avoids the decoupling of odd–even points.

Two grids are used to discretize the physical domain. For the first grid, a MAC-staggered grid system is adopted, as shown in Figure 1(a). A grid system is generated numerically or algebraically at the positions marked by filled circles. The pressure (or any one of the scalar variables, e.g. temperature, etc.) is arranged at the arithmetic center of these four circles. The numerical notation of the p^1 is designed so that indexes 1,1 belong to the left bottom mesh of the domain. $p^1_{i,0}$ lies at the bottom and belongs to the left boundary. The Cartesian velocity components are located at the midpoints of the e and w faces of the control volume. The Cartesian velocity components $u^1_{i+1,j}$ and $u^1_{i,j}$ are located at the midpoints of the s and n faces of the control volume. A typical grid node, P, is enclosed in its cell and is surrounded by its neighbors N, S, E, and W. The second grid (Figure 1(b)) is displaced so that the center of its mesh coincides with the corner of that of the first mesh grid (shaded area) and its corners correspond to the center of the first grid. The pressure $p^2_{i,j}$ in the second grid is placed at the corner of the first mesh. Note that for convenience in numerical coding and for the relationship between first and second grid indexes, fractional indexes are not used in notations, e.g. $p^2_{1,1}$ belong to the left bottom corner and $p^2_{2,2}$ to the diagonally opposite corner of the first grid mesh (Figure 1(c)). The Cartesian velocity components $u^2_{i,j}$ and $u^2_{i,j+1}$ are located at the midpoints of the s and n faces of the first grid control volume, and the Cartesian velocity components $v^2_{i+1,j}$ and $v^2_{i,j}$ are located at the midpoints of the e and w faces of the first grid control volume. Thus, two sets of the primitive variables define the physical domain—indexes of the first velocity component u^1 coincide with those of the second velocity component v^2, and this also applies to u^2 and v^1. This type of domain discretization (DSG) can be regarded as overset grids, either of which occupies the overall numerical domain.

3. VALIDATION

Application of the DSG methodology developed in [1] was demonstrated by the calculation of the external and internal problems. For the sake of brevity, in this present paper we omit the mathematical treatment of the DSG method as applied to natural convection problems; we considered the convergence rate of a DSG method. The use of the methodology will be illustrated in Section 3.1 by means of numerical comparison with a benchmark solution that includes internal buoyancy-driven flow in a squeezed cavity, and heat transfer from a cylinder enclosed in a square

DOI: 10.1002/fld

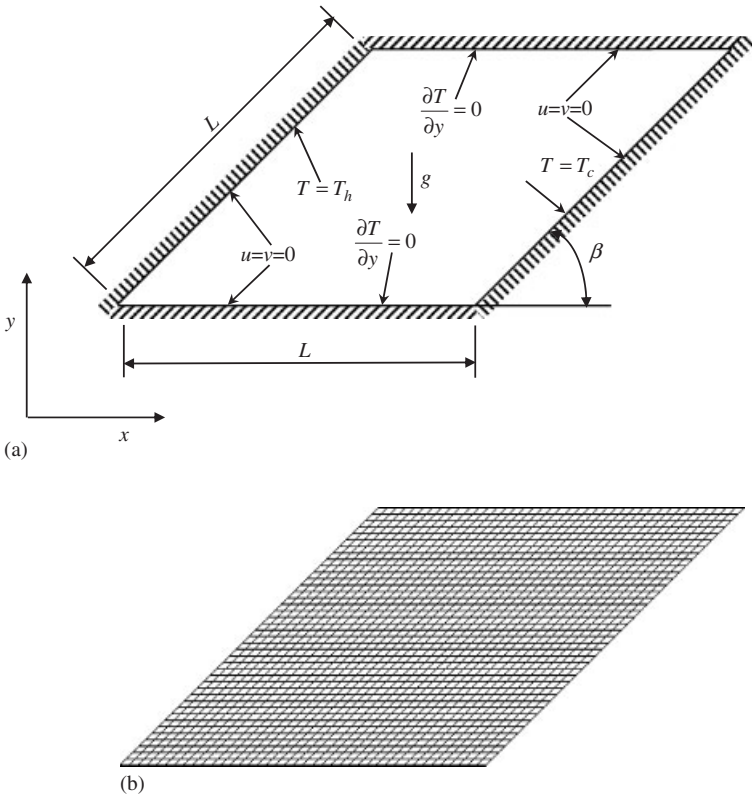

Figure 2. Squeezed cavity test case: (a) geometry and boundary conditions and (b) example of a coarse uniform grid.

duct (Section 3.2), and in Section 3.3 comparison is made with experimental studies of heat transfer between concentric and eccentric cylinders. The double-staggered algorithm was tested with double precision on a Pentium® 4 CPU 3 GHz computer. The pattern of the streamlines was calculated from the vorticity-stream function formulation of one of the momentum equations (see Reference [42]):

$$\Delta\Psi+\omega=0$$

or for grid type at Figure 2(b):

$$C_1\Psi_{\xi\xi}+C_3\Psi_{\eta\eta}-2C_2\Psi_{\xi\eta}+x_\eta u_\eta+y_\eta v_\xi-u_\xi=0 \tag{3}$$

The double-grid arrangement was also applied to Equation (3); Neumann conditions $\Psi^1_\xi|_{\xi=0,1}=\Psi^1_\eta|_{\eta=0,1}=0$ were set for the first grid function, Ψ^1, and wall boundary conditions were set to zero for the second grid function, Ψ^2.

DOI: 10.1002/fld

3.1. Flow in the squeezed cavity

The control test case involved buoyancy-driven cavity flows. Appropriate benchmark solutions were described by Demirdžić *et al*. [43]. Non-orthogonal grids were set up by inclining the sidewalls or by squeezing the cavity (Figure 2(a)). The inclined walls were kept at constant temperatures T_h and T_c, respectively; the horizontal walls were assumed to be adiabatic, and the inclination angle β was set to 45°. The dimensions and fluid properties used were $L=1$, density $\rho=1$, gravity constant $g=1$, expansion coefficient $\beta=0.1$, specific heat $C_p=1$, $T_h=1$, and $T_c=0$, so that the Rayleigh number was a function only of the viscosity and the Prandtl number. Flows at $Ra=10^6$ were studied for two values of the Prandtl number: $Pr=0.1$ and 10. Non-uniform grids, symmetrically expanding toward the centerline from all walls, were implied by Demirdžić *et al*. [43], e.g. on the finest grid, with 224×192 control volumes; the smallest x and y amounting to $\frac{1}{467}$ and $\frac{1}{454}$, respectively.

Uniform grids ranging from 480×480 to 512×512 are used for testing double-staggered algorithms. For the type of grid shown in Figure 2(b), $x_\xi=1$, $y_\xi=0$, $x_\eta=\cos\beta$, and $y_\eta=\sin\beta$. Boundary conditions are $v=u=0$ at the walls. Central differences are used to approximate both convection and diffusion fluxes. In the momentum equations, values of the pressure at the boundary nodes are required, in order to evaluate the pressure gradient. Linear extrapolation from interior values of the pressure was used to provide a benchmark solution. The presented algorithm and the squeezed cavity benchmark solutions were compared by linear extrapolation from interior values, e.g. at the bottom wall:

$$p^1_{i,0}=p^1_{i,1}-0.5\frac{\partial p^1_{i,1}}{\partial\eta}\Delta\eta \tag{4}$$

where p^1_η is calculated as the non-centered second-order derivative. p^2 at the walls is computed as the average of p^1; this method of calculating the wall pressure p^1 was used only for comparison purposes in this test case. In other cases, the wall pressure would be determined by means of the momentum equation. Figure 3 shows excellent agreement with benchmark solutions [42]. Isotherms and streamlines predicted on the 512×512 grids for flow case $\beta=45°$ are presented in Figures 4 and 5, respectively. They show all the typical features of such flow [42]. Minimum and maximum stream function values in vortex centers and their positions as predicted on the 512×512 grid are presented in Table I. Small quantitative differences were found in the displacement of the maximum stream function values of the vortexes and their intensities.

3.2. Cylinder in a square duct

Solution of the heat transfer from a cylinder, enclosed in a square duct, was a second benchmark solution (Figure 6(a)). The vertical walls of the duct were kept at temperature $T_c=0$; the horizontal walls were adiabatic. The cylinder axis was displaced vertically from the duct center line for $\Delta y_c=0.1$; the cylinder radius was $R=0.2$, and the duct dimensions were $L=1$. The fluid properties were the same as in the squeezed cavity case. The boundary-fitted physical co-ordinate system was created by numerically solving the following system of elliptic equations:

$$\begin{aligned}(fx_\xi)_\xi+(fx_\eta)_\eta&=0\\(fy_\xi)_\xi+(fy_\eta)_\eta&=0\end{aligned} \tag{5}$$

DOI: 10.1002/fld

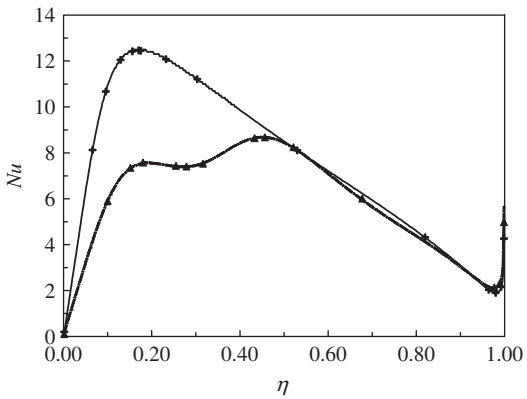

Figure 3. Profile of the local Nusselt number along the cold wall for $\beta=45°$; present study: -, $Pr=10.0$; $-\cdot-$, $Pr=0.1$; Demirdžić *et al.* [43]: +, $Pr=10.0$; ▲, $Pr=0.1$.

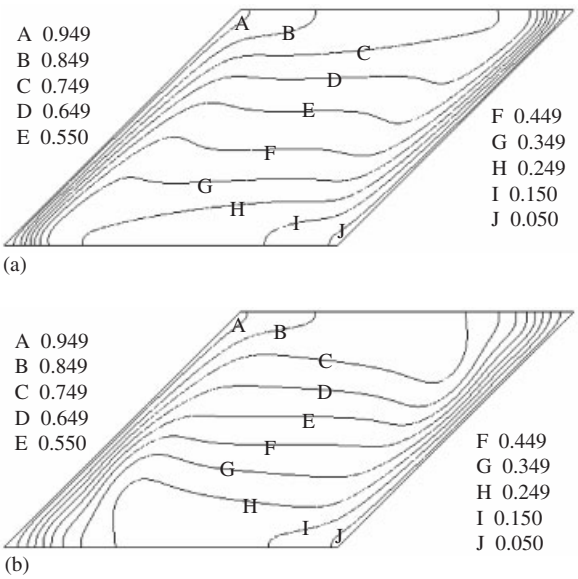

Figure 4. Predicted isotherms for $\beta=45°$: (a) $Pr=10.0$ and (b) $Pr=0.1$.

where $f=g_{22}^{1/2}g_{11}^{1/2}$. Equations (5) was used for smoothing the guessed values of x and y, by iteration. An example of this procedure (32×32 grid) is presented in Figure 6(b). Non-orthogonal grids ranging from 256×128 to 512×256 were developed for testing double-staggered algorithms. The pressure boundary conditions were obtained from the differential form of the momentum equation (for the grid type shown in Figure 6(b)) at the left and right sides of the computational

DOI: 10.1002/fld

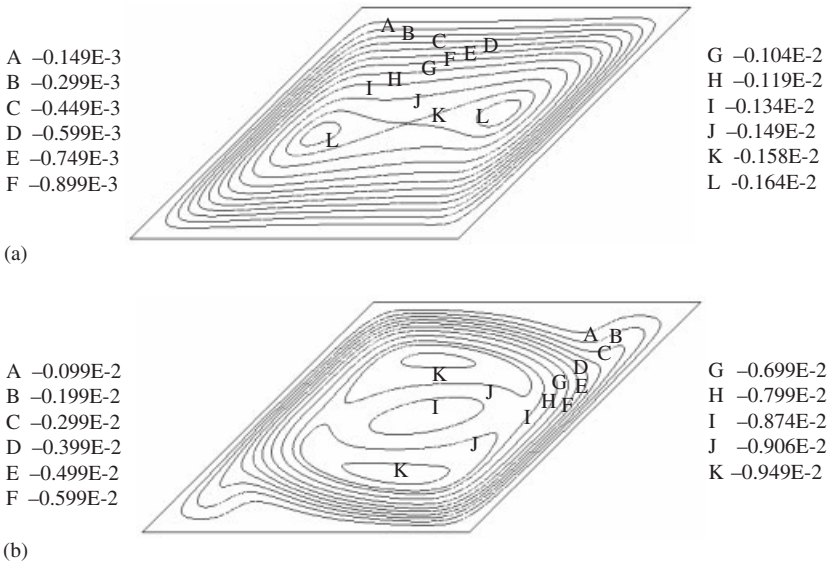

Figure 5. Predicted streamlines for $\beta=45°$: (a) $Pr=10.0$ and (b) $Pr=0.1$.

Table I. Characteristic stream function values for squeezed cavity.

	$Pr=0.1$		$Pr=10.0$	
	Demirdžić *et al.* [43]	Present study	Demirdžić *et al.* [43]	Present study
$\psi_{\min}$	-9.68706×10^{-1}	-9.657881×10^{-1}	-1.662127×10^{-1}	-1.664881×10^{-1}
x	0.78926	0.778861	0.57276	0.572366
y	0.178134	0.176777	0.319989	0.318199
$\psi_{\max}$	7.705×10^{-8}	2.59398×10^{-6}	—	—
x	1.271738	0.401979	—	—
y	0.702687	0.010312	—	—

domain ($\xi=0,1$):

$$p_\xi=(C_1u_{\xi\xi}x_\xi+C_2u_{\eta\xi}x_\xi+C_1v_{\xi\xi}y_\xi+C_2v_{\eta\xi}y_\xi)\frac{Pr}{J}$$
$$-(Uu)_\xi\frac{x_\xi}{J}-(Vu)_\xi-(Vv)_\eta\frac{y_\xi}{J}+RaPrT\frac{y_\xi}{J} \tag{6}$$

At these sides p_ξ is used for calculations of p^1 and p_η for calculating p^2 along these walls:

$$p_\eta=(C_2u_{\eta\xi}x_\eta+C_1v_{\eta\eta}y_\eta+C_2v_{\eta\xi}y_\eta)\frac{Pr}{J}$$
$$-(Uu)_\xi\frac{x_\eta}{J}-(Uu)_\xi-(Vv)_\eta\frac{y_\eta}{J}+RaPrT\frac{y_\xi}{J} \tag{7}$$

DOI: 10.1002/fld

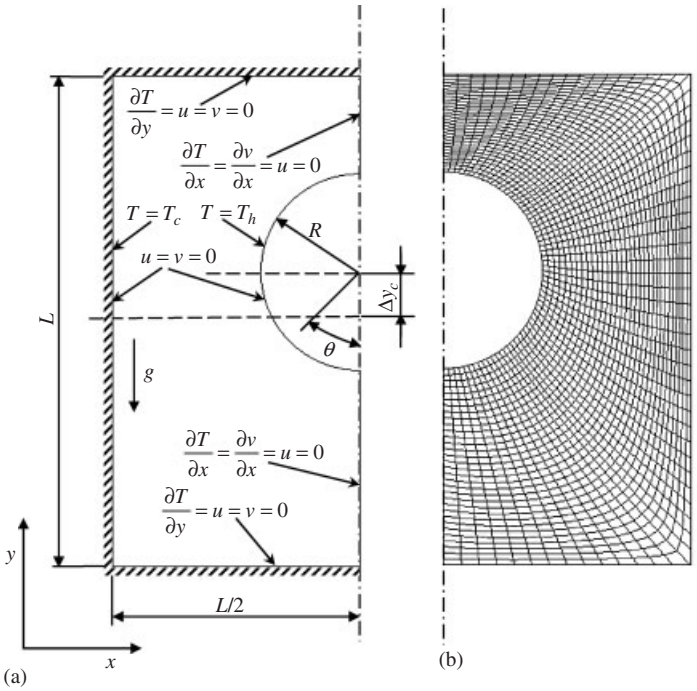

Figure 6. Duct test case: (a) geometry and boundary conditions and (b) example of coarse non-orthogonal grid (32×32).

At the top and bottom walls ($\eta = 0, 1$):

$$p_\eta = (C_3 u_{\eta\eta} x_\eta + C_2 u_{\eta\xi} x_\eta + C_3 v_{\eta\eta} y_\eta + C_2 v_{\eta\xi} y_\eta)\frac{Pr}{J} - (Vu)_\eta \frac{x_\eta}{J} - (Vv)_\eta \frac{y_\eta}{J} + RaPrT\frac{y_\eta}{J} \tag{8}$$

At these sides, p_ξ is used for calculating p^1 and p_η for calculating p^2 along η:

$$p_\xi = (C_3 u_{\eta\eta} x_\xi + C_2 u_{\eta\xi} x_\xi + C_3 v_{\eta\eta} y_\xi + C_2 v_{\eta\xi} y_\xi)\frac{Pr}{J} - (Vu)_\eta \frac{x_\xi}{J} - (Vv)_\eta \frac{y_\xi}{J} + RaPrT\frac{y_\xi}{J} \tag{9}$$

Pressures at the walls are calculated accordingly, e.g. with Equation (4); p^2 at the walls is computed as the average of p^1; in contrast to the previous comparison with a benchmark solution for the squeezed cavity, in which the boundary pressure value is extrapolated from the interior points. In the case of the duct, we evaluate the pressure boundary conditions for p^2 from the projection of the momentum equation for the first set of variables. This procedure is similar to that used in the

DOI: 10.1002/fld

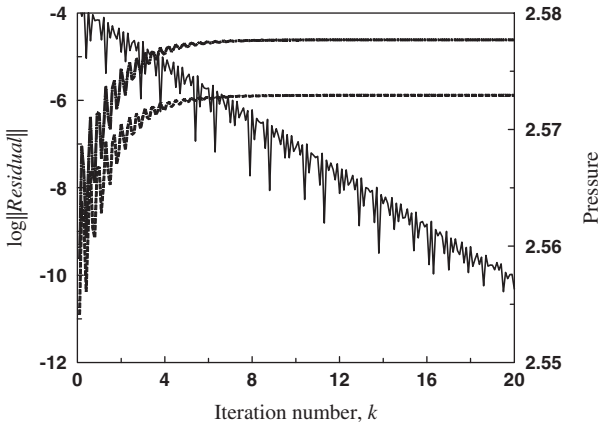

Figure 7. Convergence history in terms of log‖Residual‖ and pressures.

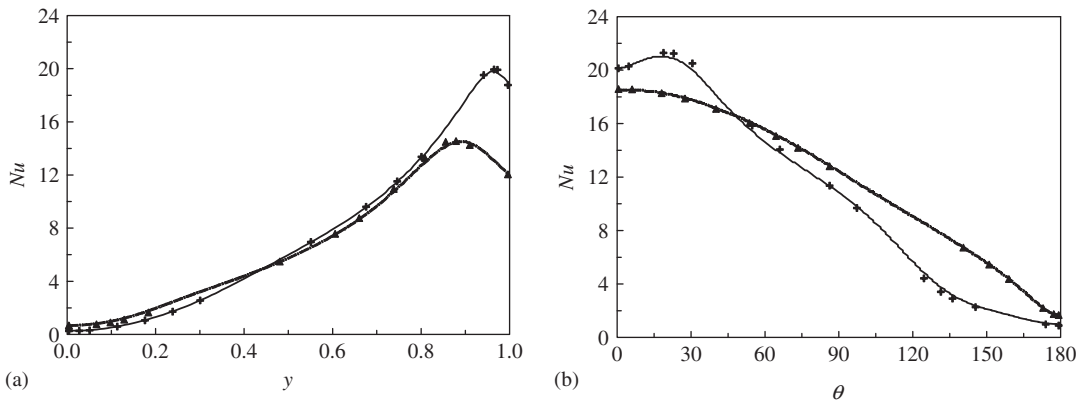

Figure 8. Profile of the local Nusselt number: (a) along the cold wall of the duct and (b) along the cylinder; present study: -, $Pr=10.0$; – · –, $Pr=0.1$; Demirdžić *et al.* [43]: +, $Pr=10.0$; ▲, $Pr=0.1$.

problem of the incompressible flow simulation with Neumann boundary conditions [44–47]. This algorithm requires holding the compatibility relationship:

$$\iint_{\Omega} \nabla^2 p \, \mathrm{d}\Omega = \oint_{\Gamma} p_n \, \mathrm{d}\Gamma \tag{10}$$

in which Ω denotes a two-dimensional region, with Γ as its boundary.

Failure to satisfy the discrete form of Equation (10) causes either very slow convergence or even divergence of the solution. Although we do not explicitly use Neumann conditions for the second pressure, in light of our experience, low approximation of the first momentum projection may lead to divergence of the solution. Figure 7 illustrates a residual and pressure convergence history for a 256×128 grid, where ‖Residual‖ is a vector norm of the mass flux through cells:

$$\|\mathrm{Residual}\| = \sum_i |\mathrm{Residual}_i|$$

DOI: 10.1002/fld

Table II. Characteristic stream function values for the duct.

	$Pr=0.1$		$Pr=10.0$	
	Demirdžić *et al.* [43]	Present study	Demirdžić *et al.* [43]	Present study
$\psi_{\min}$	-8.78350×10^{-5}	-3.88110×10^{-4}	—	—
x	0.46453	0.46692	—	—
y	0.05430	0.04820	—	—
$\psi_{\max}$	1.212974×10^{-2}	1.211576×10^{-6}	1.419586×10^{-3}	1.36635×10^{-3}
x	0.27188	0.276922	0.149187	0.133653
y	0.24593	0.19981	0.668196	0.665032

Figure 8 shows good agreement of a local Nusselt number with the benchmark solutions [43]. Streamlines such as those predicted for the 512×512 grids for the flow case of $\beta=45°$ are presented in Figure 9; they show typical features of such flow. Minimum and maximum stream function values in vortex centers and their positions as predicted on the 512×512 grid are presented in Table II.

We used another boundary approximation of the pressure and found small quantitative differences in the displacement of the maximum stream function values of the vortices and their intensities.

3.3. Natural convection in a cylindrical annulus

This problem concerns natural convection in the space between horizontal concentric cylinders (Figure 10). A comprehensive review of the relevant literature on the concentric annulus was given by Kuehn and Goldstein [48], who also studied the flow in the annulus between horizontal eccentric cylinders with a hot inner cylinder, by means of Mach–Zehnder techniques [49]. Guj *et al.* [50] analyzed the thermal field in horizontally eccentric cylinders, with a 32×32 grid; their treatment of the vertically eccentric problem included a demonstration that despite the cylindrical geometry of this problem, the solution was obtained in the Cartesian co-ordinate system. In other treatments, Kuehn and Goldstein [48] used a 16×19 grid, Farouk and Guceri [51] used a 24×50 grid, Cho *et al.* [52] and Projahn *et al.* [53] used 30×40 grids, and Shu *et al.* [54] used a vorticity-stream function formulation with a 31×21 grid. Primitive variable formulations were used by Date [55] and Guj and Stella [56]. Natural convection in annuli was widely used as test problems by Karki and Patankar [57], with a 32×32 grid and Choi *et al.* [58] with a 24×32 grid, as well as Kobayashi and Pereira [59], Ray and Date [60], and Galpin and Raithby [61] among others. The boundary conditions imposed in the present problem were two impermeable isothermal walls at constant radii and two vertical lines of symmetry at the upper and lower sides (Figure 10(a)). The flow was laminar, with non-slip conditions at the cylinder surfaces, and the temperatures of both cylinders were constant. The inner cylinder was heated ($T_h=1$) and the outer one cooled ($T_c=0$). Because of the symmetry with respect to the vertical centerline, only one half of the total annulus needs to be considered in the numerical analysis. Thus, at the centerline

$$u=0, \quad v_\xi=0, \quad T_\xi=0$$

The calculations for the eccentric geometry ($e_v=0.652$) were repeated with a grid system covering the whole region of flow, to demonstrate that symmetry conditions do not influence the numerical results. The double-grid setting procedure for natural convection problems is demonstrated on the orthogonal and non-orthogonal grids. The two-boundary technique of grid generation was used

DOI: 10.1002/fld

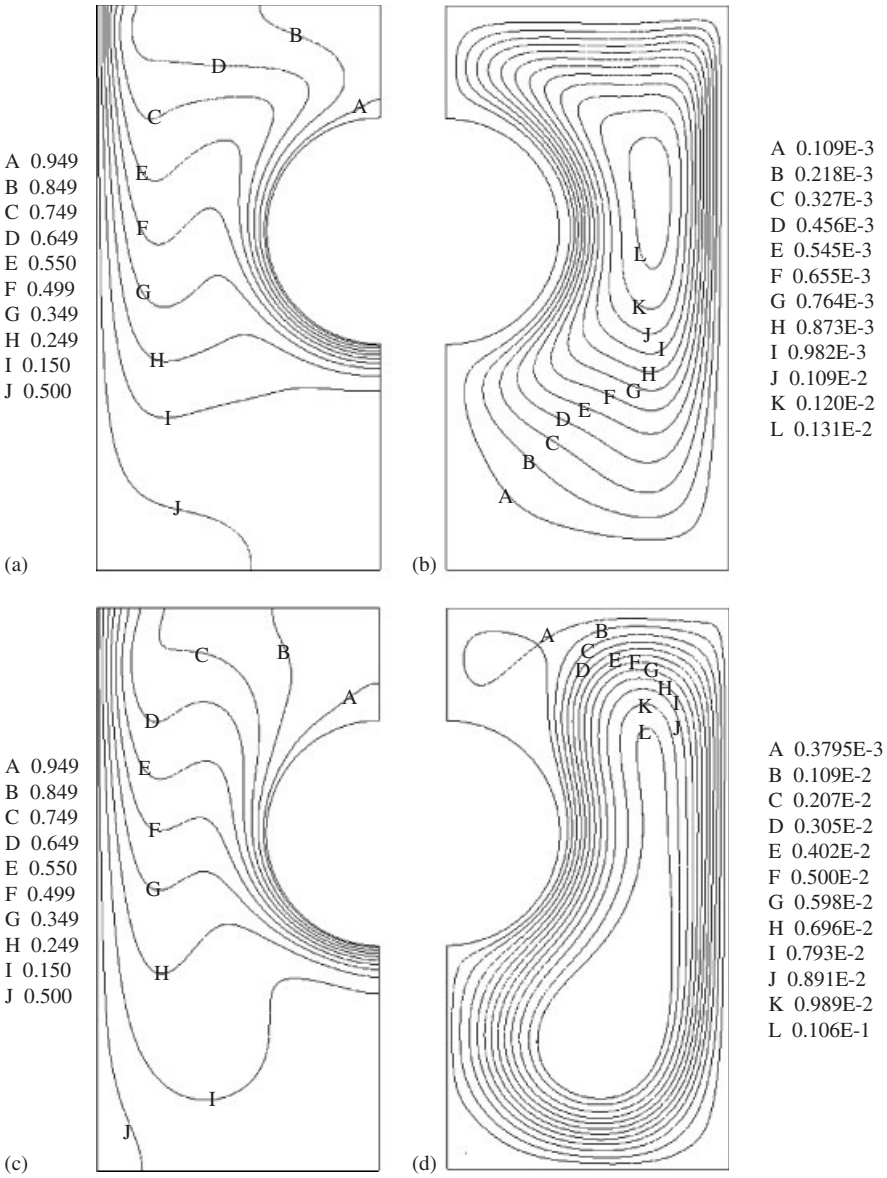

Figure 9. Predicted isotherms (a) and streamlines (b) for duct at $Pr=10.0$; and isotherms (c) and streamlines (d) for duct at $Pr=0.1$.

to build a non-orthogonal curvilinear grid in physical space. Simple interpolation of the interior between the two boundaries was provided by

$$\begin{aligned} x(\xi,\eta) &= (1-s)x_{AB}(q_{AB})+sx_{DC}(q_{DC}) \\ y(\xi,\eta) &= (1-s)y_{AB}(q_{AB})+sy_{DC}(q_{DC}) \end{aligned} \tag{11}$$

DOI: 10.1002/fld

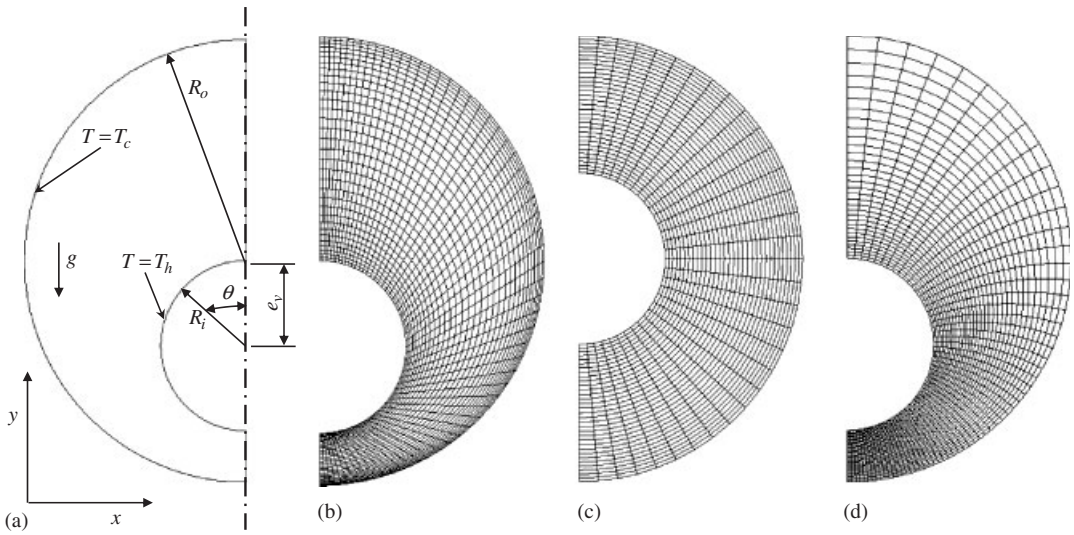

Figure 10. Geometry and meshing of an annulus (an example of a 32×32) mesh system: (a) eccentric annulus geometry; (b) non-orthogonal co-ordinates ($e_v=-0.623$); (c) polar orthogonal co-ordinates ($e_v=0$); and (d) bipolar orthogonal co-ordinates ($e_v=-0.623$).

in which s and q are stretching functions, and AB and DC are boundary curves. Figure 10(b) presents an example of the non-orthogonal grid with the same stretching functions q and s ($P=2.5$):

$$s=1+\tanh(P(\eta-0.5))/\tanh(P/2)/2 \tag{12}$$

Two types of orthogonal grid were used for calculations—polar co-ordinates as in the case $e_v=0$ and bipolar co-ordinates for $e_v=0.652$, -0.623 (Figures 10(c) and (d)). The local equivalent thermal conductivities k_{eq} for various eccentricities were compared with experimental data [48, 49]. The local equivalent thermal conductivity k_{eq} is defined as the ratio between the actual Nu and the Nu_{cond}, where Nu_{cond} is the Nusselt number for pure conduction in the concentric case:

$$\begin{aligned} k_{eq,i}(\theta) &= \frac{\partial T/\partial n|_{R_i,\theta}}{(T_h-T_c)} R_i \ln\frac{R_o}{R_i} \\ k_{eq,o}(\theta) &= \frac{\partial T/\partial n|_{R_o,\theta}}{(T_h-T_c)} R_i \ln\frac{R_o}{R_i} \end{aligned} \tag{13}$$

For a concentric configuration ($e_v=0$) (Figure 10(a)), following Kuehn's and Goldstein's [48] experiment, the Rayleigh number is taken as 4.7×10^4 and $Pr=0.706$. Numerical procedures were studied on the non-orthogonal 256×128 grid (Figure 10(b)) and on two orthogonal grids: -256×128 and 512×256.

Table III presents the results of calculations of the $k_{eq,i}$ on different grids. Power-law and central differences were used to approximate mass flux. The cell aspect ratio of the non-orthogonal grid reached 17.88, and the procedure lost stability in the central differences treatment. Table III also presents the good coincidence between the results obtained with the non-orthogonal and orthogonal grid with power-law treatment, and those obtained with central differences with 256×128 and

DOI: 10.1002/fld

Table III. Monitored $k_{\mathrm{eq},i}$ values as predicted on various grids for $Ra = 4.7 \times 10^{-4}$, e_{v}=0.

	256×128	256×128	256×128	512×256
θ	Power-law, non-orthogonal	Power-law, orthogonal	Central orthogonal	Central orthogonal
0	0.6564	0.6569	0.6208	0.6199
15	1.1417	1.1412	1.1128	1.1118
30	1.7637	1.7616	1.7602	1.7587
45	2.2316	2.2267	2.2435	2.2429
60	2.5774	2.5716	2.5929	2.5938
75	2.8850	2.8793	2.8996	2.9012
90	3.2070	3.2046	3.2239	3.2248
105	3.5617	3.5635	3.5821	3.5818
120	3.8548	3.8632	3.8708	3.8697
135	3.9242	3.9344	3.9163	3.9150
150	3.8404	3.8465	3.8125	3.8106
165	3.8110	3.8156	3.7844	3.7820
180	3.8224	3.8270	3.7984	3.7956

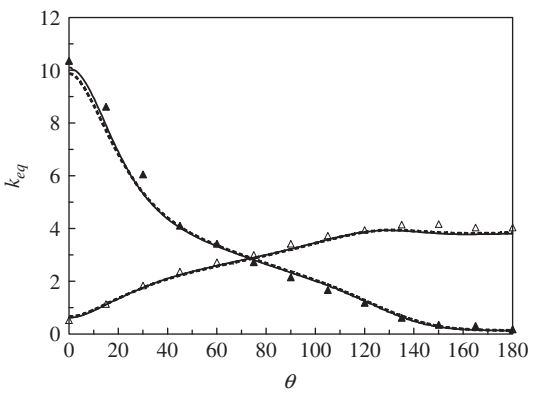

Figure 11. Comparison of concentric local heat transfer coefficients: $Ra = 4.7 \times 10^{-4}$, $e_{\mathrm{v}} = 0$: –, central differencing approximation and - -, power-law approximation, present calculation; △, inner cylinder and ▲, outer cylinder, experimental data [49].

512×256 grids. The local equivalent thermal conductivities are compared with the experimental data of Kuehn and Goldstein [49] in Figure 11. The results show good agreement.

Cho *et al.* [52], who used governing equations of the stream function, vorticity, and temperature in the cylindrical bipolar co-ordinate system on a 51×31 grid, also obtained under-prediction of the local equivalent thermal conductivities at $\theta = 180°$. Our calculations with 512×256 and 256×128 grids generated the same results. Figure 12 shows the calculated isotherms and the streamlines. Table IV presents the min and max values calculated on the 256×128 grid, together with the vortex center co-ordinates. The second vortex is always weaker.

DOI: 10.1002/fld

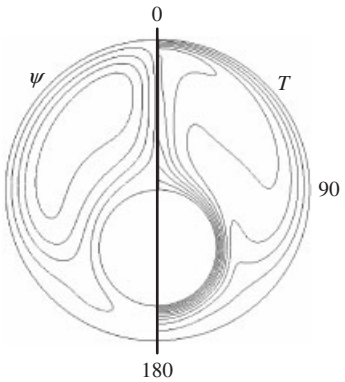

Figure 12. Streamlines and isotherms: $Ra=4.7\times10^4$, $Pr=0.706$ ($e_v=-0.623$); $\psi=[-1.5\times10^3(6)0.1]$; $\Delta T=0.1$.

Table IV. Minimum and maximum stream function values in vortex centers and their positions.

	$e_v=0$	$e_v=-0.623$	$e_v=0.652$
ψ_{min}	-1.4280×10^{-3}	-1.5714×10^{-1}	-2.6026×10^{-3}
x	0.27009	0.69181	0.16124
y	0.59300	2.4528	3.0697
ψ_{max}	9.7753×10^{-2}	4.1808×10^{-3}	6.4078×10^{-2}
x	0.67562	0.26377	1.1490
y	0.96701	2.1188	2.1681

4. CONVERGENCE PROPERTIES

In this section, the convergence properties of the DGG method are compared with the widely used 'SI' method in treating the problem of natural convection in annuli. A variety of iteration algorithms were used for solving Poisson's equation for pressure correction: the successive overrelaxation (SOR) method (Section 4.1), the symmetric overrelaxation (SSOR) method (Section 4.2), the incomplete factorization preconditioner method (Section 4.3), the conjugate gradient (CG) method (Section 4.4), and the CG methods with preconditioner (Section 4.5), and the SOR method are applied to non-orthogonal grids (Section 4.6).

Orthogonal and non-orthogonal grids were used in the above-described numerical simulations of the flow in annuli. The main advantages of the orthogonal grid are 'disappearance' of cross-derivatives (such as u_ξ, u_η, T_ξ, and T_η) and that Poisson's equation of the pressure correction contains only derivatives of the pressures correction $p_\xi^{1'}$, $p_\eta^{1'}$ or $p_\xi^{2'}$, and $p_\eta^{2'}$; also in this case, the formulation of natural variables in Cartesian co-ordinates requires a pressure gradient calculation along every curvilinear co-ordinate. Thus, it seems, in accordance with Figure 1, that one set of momentum–mass–energy equations may be used for calculations of u_1, v_1, p_1, and T_1 and gradients must be interpolated from the calculated pressure distribution. We showed that an

DOI: 10.1002/fld

additional calculation of the second set of momentum–mass–energy equations leads to accelerated convergence of the problem.

Linearization of every set of momentum–mass–energy equations by SIMPLE or a similar method (SIMPLER, SIMPLEC, etc.) gives rise to two systems for velocity components (u, v), and Poisson's equation for pressure and temperature. This iteration method inputs relaxation parameter for the momentum equation and an external pressure iteration parameter. In the DSG method, we also use common parameters for every type of two sets of the momentum–mass–energy equations [1]: in our notations, α_{p^1} is the first external pressure iteration parameter, α_{p^2} is the second external pressure iteration parameter, $\alpha_{\mathbf{u}^1}$ is the first velocity iteration parameter, and $\alpha_{\mathbf{u}^2}$ is the second velocity iteration parameter.

In the following discussion, we use a general notation for linear systems. Iteration methods can be used to give solutions for every linear system:

$$Au = b \tag{14}$$

Here A is a symmetric positive-definite matrix, b is a right-hand side vector, and u is a vector of unknowns. The linear stationary method for (14) may be expressed in the form:

$$u^{n+1} = Gu^n + k \tag{15}$$

Matrix A is decomposed in the form:

$$A = D - E - F \tag{16}$$

where D is a diagonal of A, $-E$ its strict lower part, and $-F$ its the strict upper part. The choice of an effective iterative solution method of a linearized momentum–mass–energy system depends heavily on the set of iteration parameters (the number of iteration parameters is 20 for methods such as GAUSS–ZEIDEL, SOR, SSOR, TDM, etc.). These parameters are ($i = 1$ indicates a first system, $i = 2$ indicates a second system) N_{p^i}—internal iteration number of Poisson's pressure-correction equation; N_{u^i}—iteration number of $\mathbf{u}^i$ velocity component equation; N_{T^i}—iteration number of temperature equations; α_{p^i}—external pressure, and $\alpha_{\mathbf{u}^i}$—velocity iteration parameters mentioned earlier; and relaxation parameters, whose number depends on the method of solution of the equation, e.g. ω_{T^i}—temperature relaxation factor; and ω_{p^i}—internal pressure relaxation factor.

Iteration numbers and relaxation factors are generally chosen on the basis of numerical experiments or published recommendations. We developed a genetic algorithm (GA) which searches for a global minimum calculation time in a space of iteration numbers and relaxation factors and which provides a means for unbiased comparison of DSG and SI methods. Unfortunately, a GA takes too long to search for a global minimum (calculation time and number of external iterations) for a grid of more than 32 000 elements (256×128) and thus, to make an estimate of a number iteration and relaxation parameters for general cases, with GA we will first look more closely at the convergence properties of first and second pressure Poisson equations. Some linkages between iteration parameters were obtained by preliminary numerical experiments. Iteration numbers of linearized momentum equations such as N_{u^1}, N_{v^1}, N_{u^2}, and N_{v^2} were found to be 1 for both DSG and SI methods. Velocity components relaxation factors and external pressure relaxation factors are established at less than or equal to one. In this study, we used a combination of iteration methods: the SOR method, the SSOR method, the incomplete factorization preconditioner (MILU0) method, the CG method, and CG methods with preconditioner—CG-SSOR and CG-MILU0. We use the

DOI: 10.1002/fld

SIMPLE-like method [1], but for convenience, the name of the solution method of the set of linear discretization equations was chosen to be the same as the name of the method of the solution of Poisson's equation for pressure-correction equations. The same combination of iterative methods is used when the convergence properties of the DSG method are compared with those of the SI method, for example, if the first pressure Poisson's equation is solved by the CG method and momentum and energy by the SOR method same methods for the same equations are used in a DSG method.

4.1. SOR method

The SOR method, as applied to calculations on the DSG, enables the estimation of *a priori* pressure relaxation factors, ω_p, and iteration numbers for first and second pressure-correction equations. It is possible to choose ω_p, designated as ω_{p_b}, so that the SOR method converges rapidly—more rapidly than the Gauss–Zeidel method, for example. A rough approximation to the number of iterations N_p needed to reduce the norm of the initial error vector by a factor ζ can be given by

$$N_p = -(\log \zeta)/R_\infty(\mathfrak{L}_\omega) \tag{17}$$

where $R_\infty(\mathfrak{L}_\omega)$ is the asymptotic rate of convergence and defined as

$$R_\infty(\mathfrak{L}_\omega) = -\log \mathbf{S}(\mathfrak{L}_\omega) \tag{18}$$

in which $\mathbf{S}$ is a spectral radius of the matrix $\mathfrak{L}_\omega$, and $\log x$ denotes the logarithm of x to the base e. The SOR method can be expressed in the form:

$$u^{n+1} = \mathfrak{L}_\omega u^n + k \tag{19}$$

in which the matrix $\mathfrak{L}_\omega$ is the SOR iteration matrix [62]:

$$\mathfrak{L}_\omega = (I - \omega L)^{-1}(\omega U + (1-\omega)I) \tag{20}$$

and

$$L \equiv D^{-1}, \quad U \equiv D^{-1}E \tag{21}$$

The optimum value of ω_{pb} is given by

$$\omega_{pb} = \frac{2}{1+\sqrt{1-M(B)^2}} \tag{22}$$

where M is the algebraically largest eigenvalue of B; B is the Jacobi iteration matrix [62]:

$$B = I - D^{-1}A \tag{23}$$

The corresponding spectral radius of $\mathfrak{L}_\omega$ is equal to $\omega_{pb} - 1$. Table V presents the spectral radii of the Jacobian iteration matrix of the pressure-correction equations, as computed by the power method [63] and the number of iterations required to reduce the error for the factor $\zeta = -1$. This and other similar tables that compare spectral properties of first and second Poisson equations for pressure correction may be used for the estimation of the iteration parameters. In such a manner, the relationship $N_{p^1} : N_{p^2} = 8$ (Table V) is suitable for approximately minimizing the overall calculation time on the DSG by the SOR method, as N_p is the number of iterations

DOI: 10.1002/fld

Table V. Spectral radii **S** and numbers of iterations, N_p of Poisson's pressure-correction equations solved by the SOR method on the DSG.

	$\mathbf{S}^1$	$\mathbf{S}^2$	N_{p^1}	N_{p^2}
64×32	0.9795	0.8542	48	6
128×64	0.9874	0.9190	78	11
256×128	0.9937	0.9533	157	20
512×256	0.9969	0.9695	292	32

determined from the asymptotic rate of convergence. Just as a numerical solution on a 512×256 grid converges at $N_{p^1}=292$ and $N_{p^2}=32$, approximately the same overall calculation time was obtained at $N_{p^1}=150$ and $N_{p^2}=20$. In the DSG method two Poisson equations for pressure correction are solved: the first pressure correction calculated on an $N_1\times N_2$ grid is a problem with Neumann boundary conditions and the second equation calculated on an $N_1+1\times N_2+1$ grid is a Dirichlet problem, whose boundary conditions are obtained from the solution of the first pressure-correction equation. Values of N_{p^1} and N_{p^2} differ considerably for these two problems. Temperature equations for both the DSG and the SI methods were calculated by the SOR and the MILU(0) iteration procedures; the SOR method was used in two variants with $N_{T^1}=N_{T^2}=1$, and with values of N_{T^1} and N_{T^2} calculated by GA. Three iteration procedures are used for the temperature equation (Table VI): upper index 1—the SOR method, $N_T=1$; upper index 2—the SOR method, $N_T=$var; and upper index 3—the MILU(0) method. Calculations by the DSG and SI methods, presented in the Table VI for 64×32 and 128×64 grids, indicated that in the DSG method the SOR relaxation parameter is always greater than 1.0. In the SI method these parameters are greater than 1.0 only for factor $\zeta\leqslant-4$, but the condition $\zeta\leqslant-7$ leads to an optimum value at $\alpha_p\leqslant1$ and the optimum value diminishes sharply as the meshing increases. The value of the α_p of DSG lies in the range from 0.722 to 1.096 and depends weakly on the meshing. Related results for high values of the α_p were obtained by Vuik *et al.* [64] with the Krylov's accelerated SIMPLE(R) method. At the same time, α_p is strongly controlled by meshing in the SI method. Thus, from Table VI it can be seen that the SOR iterative procedure is the best suited to temperature equations.

Table IX shows results of GA solutions to both DSG and SI methods. With the constraint $\|\text{Residual}\|\leqslant10^{-7}$, for the external iteration number of the SOR method (the iteration number of the SIMPLE method) is about 14 times greater and the computing time is about 7 times longer than that for the DSG method for a 64×32 grid, and the corresponding values of these parameters are 20 and 10 for a 128×64 grid. Comparison results were not obtained for finer grids, because the calculation time of the DSG method increases with increasing domain meshing, whereas an interpolation method calls for very small iteration parameters that lead to unacceptably long calculation times. The SOR calculation time of the solution of the first Poisson equation for pressure correction amounts to one-sixth of the general calculation time of the iteration procedure; that for the second Poisson equation amounts to only one-fiftieth of the general time. This time distribution is approximately the same as that in other iterative methods of solving of Poisson's equations in DSG. This finding supports the view that the DSG method enables external iterations into SIMPLE-like methods with higher values of iteration parameters in general co-ordinates and thereby leads to faster convergence than the SI method. The value of the relationship $N_{p^1}:N_{p^2}$ calculated by the GA (Table V) is less than would be expected from an asymptotic rate of convergence, but nevertheless, the number of iterations for the second Poisson pressure equation is less than that for

DOI: 10.1002/fld

Table VI. Optimal iteration parameters of the DSG and SI methods with the SOR iterating procedure for Poisson's pressure-correction equations.

	t	N_{p1}	N_{p2}	N_{t1}	N_{t2}	ω_{b1}	ω_{b2}	α_v	α_p
64×32 cells; $\|Residual\| \leqslant 10^{-4}$									
DSG[1]	12	19	48	1	1	1.853	1.806	0.605	0.782
DSG[2]	7	32	23	10	2	1.604	1.915	0.875	0.722
DSG[3]	9	55	16	11	4	1.909	1.901	0.668	1.096
SI[1]	24	64	—	16	—	1.283	—	0.247	0.093
SI[2]	23	48	—	4	—	1.413	—	0.291	0.077
64×32 cells; $\|Residual\| \leqslant 10^{-7}$									
DSG[1]	30	20	4	1	1	1.790	1.802	0.571	0.965
DSG[2]	24	30	12	2	2	1.726	1.610	0.785	1.033
DSG[3]	40	23	16	1	7	0.8542	1.782	0.800	0.860
SI[1]	224	5	—	1	—	0.8542	—	0.454	0.501
SI[2]	180	6	—	2	—	0.8542	—	0.079	0.583
128×64 cells; $\|Residual\| \leqslant 10^{-4}$									
DSG[2]	45	45	29	11	11	1.875	1.826	0.920	0.788
DSG[3]	50	29	22	6	1.911	1.911	1.859	0.903	0.731
SI[1]	200	98	—	3	—	0.992	—	0.426	0.031
128×64 cells; $\|Residual\| \leqslant 10^{-7}$									
DSG[2]	231	33	15	2	5	1.973	1.840	0.954	0.622

the first. Thus, the convergence properties of the DSG method depend not only on the properties of the pressure-correction equation but also on the external iterative method. Figure 13 shows the convergence history of the SIMPLE method, in which all the Poisson equations (pressure corrections and temperatures) are calculated by the SOR iterative procedure. The GA provides curves of convergence history without fluctuations in this figure, and similar convergence histories are obtained by the DSG method.

4.2. The SSOR method

In the SSOR method the linear system associated with Poisson's pressure-correction equation is solved twice at each internal iteration: first, $u^{n+1/2}$ is computed by means of the SOR method, then $u^{n-1/2}$ is computed by means of the backward SOR method. If the spectral radius of the matrix LU satisfies

$$\mathbf{S}(\mathrm{LU}) \leqslant 0.25 \tag{24}$$

then a good value of ω_{pb} is given by [62]

$$\omega_{pb} = \frac{2}{1+\sqrt{2(1-M(B))}}$$

DOI: 10.1002/fld

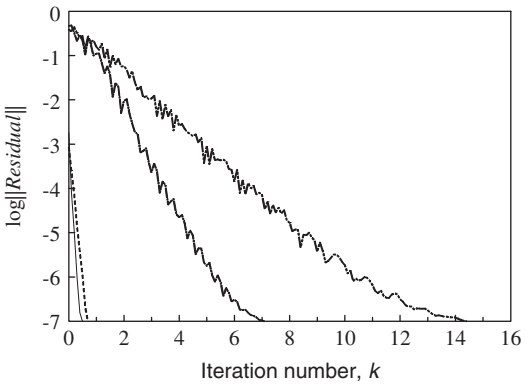

Figure 13. Convergence history of the SIMPLE method with the SOR iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; ·-· 64×32 cells, SI; and ·-·- 128×64 cells, SI.

Table VII. Spectral radius of the matrix LU.

Grid	$\mathbf{S}(\mathrm{LU})^1$	ω_{pb^1}	$\mathbf{S}^1$	N_{p^1}	$\mathbf{S}(\mathrm{LU})^2$	ω_{pb^2}	$\mathbf{S}^2$	N_{p^2}
64×32	0.0934	1.9795	0.9869	96	0.0612	1.8542	0.9243	12
128×64	0.0985	1.9876	0.9938	159	0.1171	1.9186	0.9584	23
256×128	0.1083	1.9909	0.9954	314	0.1324	1.9542	0.9769	41
512×256	0.0774	1.9966	0.9982	583	0.1383	1.9695	0.9846	64

with this ω_{pb} the spectral radius $\mathbf{S}$ of the satisfies

$$\mathbf{S}\leqslant\left(1-\sqrt{\frac{1-M(B)}{2}}\right)\left(1+\sqrt{\frac{1-M(B)}{2}}\right)$$

As indicated in Table VII, the SSOR condition (24) is satisfied; therefore, the SSOR method can be used effectively, although it has been argued that the rate of convergence of the SSOR method is relatively insensitive to the choice of ω (Figure 14). The calculation time strongly depends from on the ω, but the optimal value is 1.8701 and differs from the calculated asymptotic value (for 128×64 grids and $N_{p^1}:N_{p^2}=7.5$). A similar result was obtained by Ashcraft and Grmes [65], the relationship between iteration number and ω for the SSOR preconditioning is roughly linear between $\omega=1$ and its optimal value, ω; thus, the additional speed of iterating is achieved at the cost of more iterations. Figure 15 presents the convergence history of the SSOR iterative procedure.

4.3. MILU preconditioner

The general problem of finding a preconditioner for a linear system (14) is to find a matrix M (preconditioner) such that M is a good approximation to A. A suitable choice of M can significantly accelerate the convergence of the method. In particular, Buleev [66], Stone [67], and Dupont *et al*. [68] proposed an approximate factorization method for elliptic problems. It is convenient for

DOI: 10.1002/fld

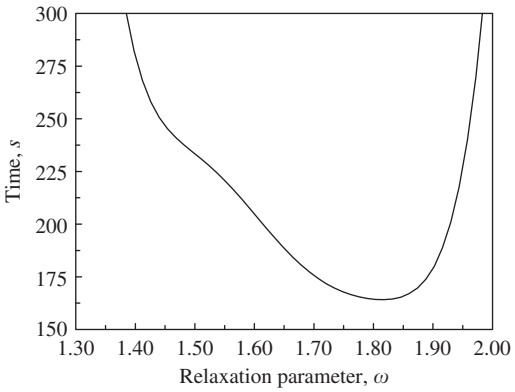

Figure 14. Calculation time *vs* SSOR relaxation parameter ω, 128×64 grid, $\|\text{Residual}\|\leqslant10^{-7}$.

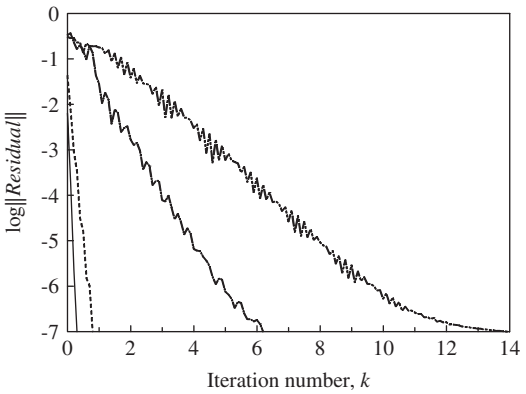

Figure 15. Convergence history of the SIMPLE method with the SSOR iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; -·- 64×32 cells, SI; and -···- 128×64 cells, SI.

problem (14) to compute an incomplete $LD^{-1}U$ or modified incomplete $MLD^{-1}U$ factorization of A, where L is the lower triangular, D is the diagonal, and U is the upper triangular. We use the modified ILU(0) (MILU(0)) preconditioner to a general matrix A due to Gustafsson [69]. The ratio between the iteration numbers for Poisson's first and second equations is ≈4 in the MILU(0) method, in which the first iteration number usually had been estimated to be ≈30. Under the constraint of $\|\text{Residual}\|\leqslant10^{-7}$, in SI method, the external iteration number and the computing time are, respectively, about 10 and 20 times larger than their values in the DSG method for 64×32 grids, and for a 128×64 grid the corresponding values of these parameters are about 18 and 8, respectively (Figure 16).

4.4. The CG

The CG method does not require internal relaxation parameters, and parameter stopping of the process may be prescribed. The calculated solution norm may be a parameter of the stopping

DOI: 10.1002/fld

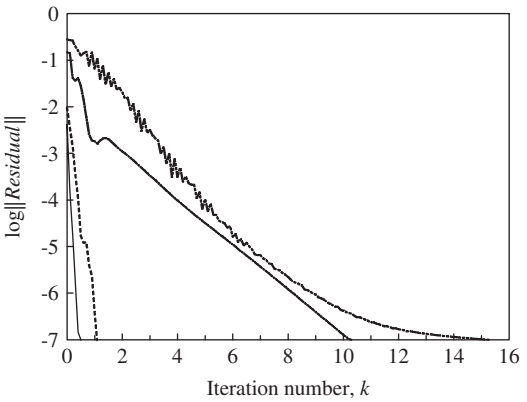

Figure 16. Convergence history of the SIMPLE method with the MILU0 iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; ·-· 64×32 cells, SI; and ·-·- 128×64 cells, SI.

process in internal iterations during the CG iteration process. The use of such criteria for the long external iterations could result in hundreds of iterations at the beginning of the iteration process and few iterations at the end. In order to estimate the convergence of the CG for one external iteration, we used a theorem that characterizes the approximation [70]. The approximate solution obtained from the mth (N_pth) step of the CG algorithm is an x_m, the exact solution is x_0. Then x_m is of the form

$$\|x_0-x_m\|_A\leqslant 2\left(\frac{\sqrt{k}-1}{\sqrt{k}+1}\right)^m\|x_0-x_m\|_A$$

and k is a spectral conditions number $k=\lambda_{\max}/\lambda_{\min}$. Table VIII presents some spectral properties of the A iteration matrix and N_{p^1} and N_{p^2} of the pressure-correction equations computed by the power and inverse power methods [63] for the factor $\varsigma=-1$. The data in Table VIII were calculated with residual of 10^{-14}, but the solution changed slowly after a few iterations. The tridiagonal matrix related to Lanczos's iteration from the coefficients of the CG algorithm was also used for the approximation of the largest and smallest eigenvalues of A. The largest approximation value of the eigenvalue $\lambda_{\max}$ of Poisson's first pressure-correction equation was 0.3380 and that for Poisson's second equation was 0.3373, although N_{p^1} and N_{p^2} were small (18 and 8); these values are in good agreement with the corresponding values for a 128×64 grid obtained from Table VIII. Experimentally it was found that the relationship $N_{p^1}:N_{p^2}\approx3$ is a suitable basis for a calculation on the DSG in the CG method for the annulus problem. This relationship, as calculated by the GS method, is less than the value of the relationship $N_{p^1}:N_{p^2}$ calculated by the SOR method, and the relationship calculated by a GA for a 128×64 grid was close to the asymptotic values. Table IX shows that the number of external iterations and computing time are wasted as a result of using the iterative method used for solving the pressure-correction equations; we must emphasize that these values were obtained with the SIMPLE method. Nevertheless under the constraint of $\|\text{Residual}\|\leqslant10^{-7}$ with the SI method the external iteration number and the computing time are, respectively, about 13 and about 3 times larger than their values with the DSG method

DOI: 10.1002/fld

Table VIII. Spectral properties of Poisson's pressure-correction equations, as solved by CG iteration procedure.

Grid	$\lambda^1_{\max}$	k^1	N_{p^1}	$\lambda^2_{\max}$	k^2	N_{p^2}
64×32	1.2306	5.5×10^3	62	1.2259	5.5×10^2	19
128×64	0.3741	8.1×10^3	76	0.3707	9.8×10^2	26
256×128	0.1094	1.0×10^4	84	0.1094	1.8×10^3	35
512×256	0.0275	1.9×10^4	116	0.0274	2.8×10^3	44

Table IX. Computational results for natural convection in annulus.

	$\|\text{Residual}\|\leqslant10^{-4}$				$\|\text{Residual}\|\leqslant10^{-7}$			
	In	Ct(s)	Rin	Rct	In	Ct(s)	Rin	Rct
64×32 grid								
SOR	157	7	3.2	2.0	524	24	14.2	7.2
SSOR	159	7	6.2	3.4	400	18	17.0	9.1
MILU0	171	10	4.7	2.2	461	21	19.5	10.0
CG	222	14	4.0	2.3	479	31	13.2	5.1
CG-SSOR	167	8	5.9	3.0	462	22	11.7	5.8
CG-MILU0	177	8	4.1	2.5	556	31	16.6	7.3
128×64 grid								
SOR	218	45	5.86	3.42	736	153	20.1	10.5
SSOR	214	44	5.83	2.89	883	167	15.9	8.6
MILU0	256	57	5.25	2.47	830	184	18.5	8.2
CG	265	95	7.18	3.32	1099	307	15.9	5.8
CG-SSOR	271	54	3.09	1.74	887	181	18.3	9.9
CG-MILU0	256	57	3.20	1.65	873	187	16.1	9.4

In is a number of iterations in the DSG method; Ct(s) is a computing time (s); Rin is a relationship between number of iterations in SI and number of iterations in DSG; Rct is a relationship between computing time in the SI method and that in DSG.

for 64×32 grids; the corresponding values of these parameters for a 128×64 grid are about 16 and 6, respectively (Figure 17).

4.5. CG with the preconditioner

The spectral conditions k of the matrix A are large (Table VIII); therefore, the convergence with the CG method is much worse than with the other methods presented in Table IX. The iterative method with enhanced spectral properties is the CG method with preconditioner. Oliveira and Issa [71] gave one version of the CG solver with incomplete Cholesky preconditioner, for computations of buoyancy-driven flows in the rectangular cavity. The number of inner iterations required to solve the set of linear equations varies according to the specified tolerance for relative decay of the residuals. The number of internal iterations needed to solve the energy equation was greater than that for the momentum equations; this is also attributed to the differing types of boundary conditions for temperature and velocity components. In this study we consider CG with SSOR and MILU(0) preconditioner.

DOI: 10.1002/fld

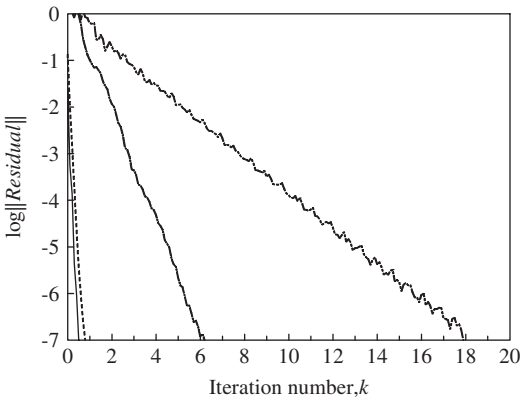

Figure 17. Convergence history of the SIMPLE method with the CG iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; ·-· 64×32 cells, SI; and ·-·- 128×64 cells, SI.

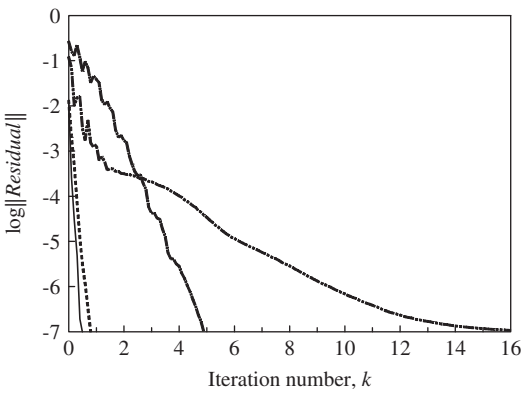

Figure 18. Convergence history of the SIMPLE method with the CG-SSOR iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; ·-· 64×32 cells, SI; and ·-·- 128×64 cells, SI.

4.5.1. CG with SSOR preconditioner (CG-SSOR). The SSOR preconditioner is defined as

$$M=(D-\omega E)D^{-1}(D-\omega F)$$

Spectral conditions number of the matrix $M^{-1}A$, defined from the tridiagonal matrix related to Lanczos's iteration, is ≈900, which is lower than the A spectral conditions number in the CG method (Table VIII, 128×64 grid). Comparison of the results of the calculations by the CG-SSOR method presented in Table IX and Figure 18 shows that although the calculation time and number of iterations needed in this method are less than in the CG method, using of the SSOR method without acceleration for the pressure-correction equations provides a greater improvement of the convergence rate of the SIMPLE-like method.

DOI: 10.1002/fld

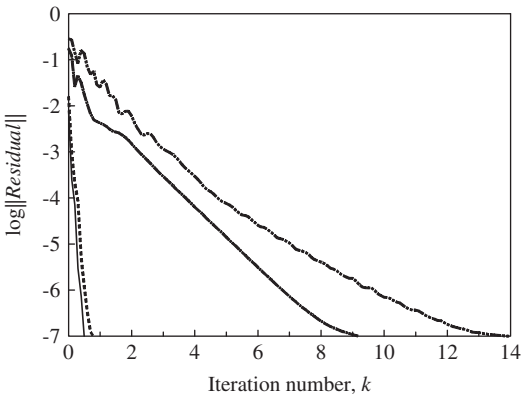

Figure 19. Convergence history of the SIMPLE method with the CG-MILU0 iteration process: - 64×32 cells, DSG; - - 128×64 cells, DSG; ··· 64×32 cells, SI; and ···· 128×64 cells, SI.

The ratio between the iteration numbers for Poisson's first and second equations is approximately 2; the first iteration number usually had been estimated to be ≈10. Under the constraint $\|\text{Residual}\|\leqslant10^{-7}$, for the SI method the external iteration number and the computing time are, respectively, about 12 and 6 times larger than their values in the DSG method for a 64×32 grid; the corresponding values of these parameters for a 128×64 grid are about 18 and 10, respectively.

4.5.2. CG with the MILU(0) preconditioner (CG-MILU0). The MILU(0) preconditioner discussed previously was used with the CG procedure. Spectral conditions number of the matrix $M^{-1}A$, defined from the tridiagonal matrix related to Lanczos's iteration, is ≈237, which is significantly lower than the spectral conditions number in the CG method (Table VIII, grid 128×64). Results of the calculation of CG with MILU(0) preconditioner (CG-MILU) are presented in Table IX and Figure 19. Use of the CG-MILU(0) for solving of the pressure-correction equations offers no advantages over MILU(0) in terms of the number of external iterations or in calculation time. Furthermore, under the constraint of $\|\text{Residual}\|\leqslant10^{-7}$, the CG-MILU(0) method with a 64×32 grid is less effective than the MILU(0) method. This highlights the finding that conjugate acceleration of the SSOR and MILU0 methods for the solution of Poisson's pressure-correction equations does not increase the convergence rate the SIMPLE-like method.

Under the constraint of $\|\text{Residual}\|\leqslant10^{-7}$ for the SI method, the external iteration number and computing time are, respectively, about 16 and 7 times larger than their values with the DSG method for a 64×32 grid, the corresponding values of these parameters for a 128×64 grid are about 16 and 9, respectively.

4.6. Non-orthogonal grids

A set of 64×32 grids displaced as shown in Figure 20 was built by using a stretched function s (Equation (12)), $0\leqslant P\leqslant0.9$; $P=1.0$ is the orthogonal grid presented in Figure 10(c); under condition $P\leqslant0.5$ the ξ co-ordinate crosses an inner cylinder. The SOR method was applied to all Poisson's equations in the SIMPLE-like method in a non-orthogonal grid. The calculation time in the DSG method increased with increasing non-orthogonality (Table X and Figure 21); for $P>0.6$ the relaxation parameters α_p and $\alpha_{\mathbf{u}}$ have higher values, but for $P\leqslant0.6$ the calculation time

DOI: 10.1002/fld

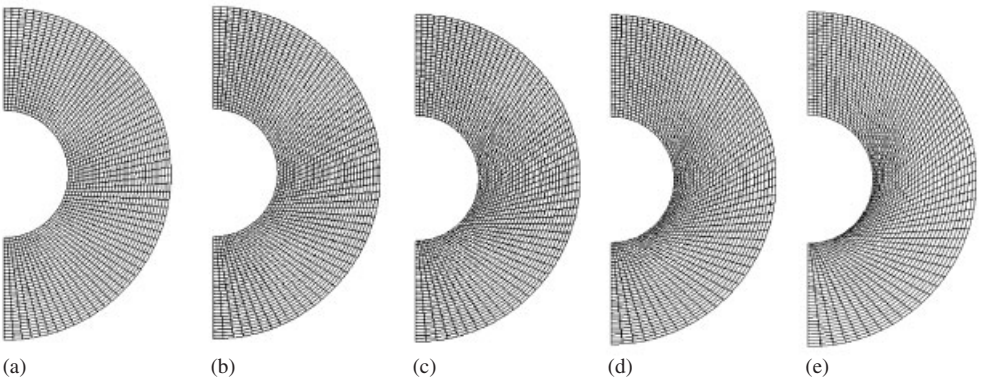

Figure 20. Non-orthogonal grids for natural convection in annuli: (a) $P=0.9$; (b) $P=0.8$; (c) $P=0.7$; (d) $P=0.6$; and (e) $P=0.5$.

Table X. Computational results for natural convection in annuli with a non-orthogonal 128×64 DSG grids.

P	In	Ct(s)	N_{p^1}	N_{p^2}	α_p	$\alpha_{\mathbf{u}}$
1.0	736	153	58	24	0.923	0.831
0.9	866	176	46	26	0.910	0.836
0.8	903	177	57	14	0.907	0.845
0.7	960	191	46	26	0.870	0.741
0.6	2008	379	32	9	0.508	0.781
0.5	3974	768	57	27	0.328	0.488

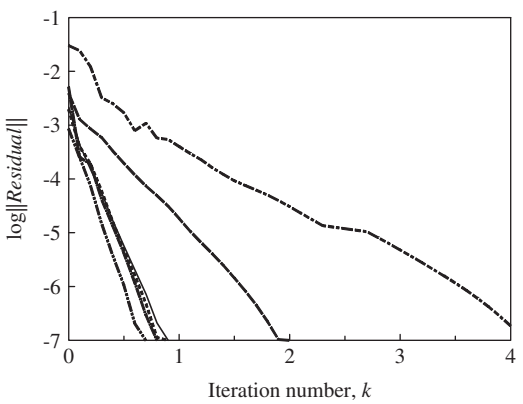

Figure 21. Convergence history of the SIMPLE-like algorithm with the SOR method with the annular non-orthogonal 128×64 grid in DSG method: -··- $P=0.9$; -·- $P=0.8$; – $P=0.7$; - - $P=0.6$; and – -· $P=0.5$.

DOI: 10.1002/fld

increased abruptly and the relaxation parameters α_p and α_u decreased. Under the constraint of $P<1.0$, the SI method calls for very small iteration parameters that lead to unacceptable calculation times.

5. CONCLUSION

A methodology has been presented for the numerical solution of two-dimensional convection–diffusion problems for arbitrary solution domains [1]. Use of this methodology was illustrated by comparison with the benchmark solutions for natural convection in a squeezed cavity and a hot cylinder in a duct. The DSG method as presented extends the solution of natural convection in concentric and eccentric horizontal cylindrical annuli. The numerical results with the orthogonal and non-orthogonal curvilinear co-ordinates were found to be in good agreement with published experimental results.

The convergence properties of the DGG method were compared with those of the widely used 'SI' method for the problem of natural convection in annuli. A GA developed to solve problems of numerical optimization of calculation time provided a means of making an unbiased comparison between the DSG and the SI methods. A variety of iteration algorithms were used for solving Poisson's equation for pressure correction: the SOR method, the SSOR method, the incomplete factorization preconditioner method, the CG method, and CG methods with preconditioners. The best calculation time was achieved with the SOR and the SSOR iteration procedures. The CG method with the SSOR and MILU0 preconditioners for the solution of Poisson's pressure-correction equations did not increase the convergence rate of the SIMPLE-like method. Use of iterative parameters such as pressure-correction relaxation and internal velocity relaxation realized higher values both on orthogonal grids and on the strongly non-orthogonal DSG, but the SI method calls for very small iteration parameters that lead to unacceptable calculation times on the strongly non-orthogonal grids.

DSG can be treated as a type of overlapping grids, both of which discretize the common domain (or sub-domain). In the numerical examples, grid boundaries are displaced at a domain boundary; interfaces between shifted MAC grids are calculated only at this boundary. In the context of convergence rate with the DSG, we would like to refer to some intuitive reason to improve the convergence by partitioning the working domain into a fixed collection of subset domains (the ADI method), according to Forsythe and Wasow [72]: 'During any linear iterative process, the error $\|E_k\|$ tends to settle into a certain 'rut'; i.e. if there is a dominant eigenvalues λ^1, eventually $E_k = c\lambda_k^1 X_1$ where X_1 is the corresponding eigenvalue. For a significant problem, one usually finds that $|\lambda_1|$ is only slightly less than 1, so that $\|E_k\|$—is making only very slow progress to zero; this is a 'rut'. Now, if at this stage one changes the iterative process, one will ordinary acquire a new set of eigenvectors. At the moment of change, the vector X will not be an eigenvector of the new process, and one may expect several steps to take place before $\|E_k\|$ settles into a new rut. Meanwhile, one may hope decreased $\|E_k\|$ considerably'. The calculation process on the DSG comprises sequences of changing of boundary conditions (Dirichlet, Neumann) and calculated domains (first grid, second grid); these switches, according to the 'intuitive reason' of Forsythe and Wasow lead to a higher convergence rate with the DSG than with the SI method. Use of DSG is of interest in general domain decomposition methods as, for example, when one of the sub-domains has a non-orthogonal grid. This represents the next step in implementing DSG in a domain decomposition method.

DOI: 10.1002/fld

NOMENCLATURE

C_1, C_2, C_3, C_4	transformed diffusion coefficients
e_v	eccentricity along vertical axis (positive upwards)
J	Jacobian of the transformation
$\mathbf{g}$	matrix transformation
g	gravity constant
l	length
Nu	Nusselt number
p	pressure
Pr	Prandtl number
Ra	Rayleigh number
Re	Reynolds number
R_i	inner cylinder radius
R_o	outer cylinder radius
T	temperature
$\mathbf{u}$	velocity vector
u, v	Cartesian velocity components
U, V	contravariant velocity components
u	unknown vector in the matrix equation $Au=b$
x, y	Cartesian co-ordinate system
α	angle
λ	eigenvalue
ξ, η	general curvilinear co-ordinates
ρ	density
ϕ	general scalar field
Ψ	stream function
ω	vorticity, relaxation parameter

Superscripts

1, 2	first and second primitive variables fields
$'$	field correction
$\mathbf{u}$	velocity
u, v	velocity component

Subscripts

E, P, N, S, W	values associated with center of neighbor control volumes
e, n, s, w	values associated with control volumes faces
p	pressure
x, y, ξ, η	partial derivatives
Ω	two-dimensional domain
Γ	domain boundary

REFERENCES

1. Shklyar A, Arbel A. Numerical method for calculation of the incompressible flow in general curvilinear co-ordinates with double staggered grid. *International Journal for Numerical Methods in Fluids* 2003; **41**:1273–1294.

DOI: 10.1002/fld

2. Chorin AJ. A numerical method for solving incompressible viscous problems. *Journal of Computational Physics* 1967; **2**:12–26.
3. Harlow FH, Welsh JE. Numerical calculations of the time-dependent viscous incompressible flow with free surface. *Physics of Fluids* 1965; **8**:2182–2189.
4. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computations* 1968; **22**:745–762.
5. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. Hemisphere: New York, 1980.
6. Van Doormaal JP, Raithby FD. Enhancements of the SIMPLE method for predicting incompressible fluid flows. *Numerical Heat Transfer* 1984; **7**:147–163.
7. Tang HS. Study on a grid interface algorithm for solutions of incompressible Navier–Stokes equations. *Computers and Fluids* 2006; **35**:1372–1383.
8. Faghri M, Sparrow EM, Prata AT. Finite-difference solutions of convection–diffusion problems in irregular domains, using a non-orthogonal coordinate transformation. *Numerical Heat Transfer* 1984; **7**:183–209.
9. Shyy W. A numerical study of annular dump diffuser flows. *Computer Methods in Applied Mechanics and Engineering* 1985; **53**:47–65.
10. Shyy W, Tong SS, Correa SM. Numerical re-circulating flow calculation using a body-fitted coordinate system. *Numerical Heat Transfer* 1985; **8**:99–113.
11. Vanka SP, Chen BC-J, Sha WT. A semi-implicit calculation procedure for flow described in body-fitted coordinate systems. *Numerical Heat Transfer* 1980; **3**:1–19.
12. Hirt CW, Amsden A, Cook JL. An arbitrary Lagrangian–Eulerian computing method for all flow speed. *Journal of Computational Physics* 1974; **14**:227–253.
13. Amsden AA, Ramshaw JD, O'Rourke, Dukowicz JK. KIVA: a computer program for two- and three-dimensional fluid flows with chemical reactions and fuel sprays. *LA-10245-MS*, 1985.
14. Fortin M, Peyret R, Temam R. Calcul des ecoulements d'un fluid visqueux incompressble. *Journal de Mécanique* 1971; **10**:357–390.
15. George A, Huang LC, Tang WP, Wu YD. Numerical simulation of unsteady incompressible flow ($Re \leqslant 9500$) on the curvilinear half-staggered grid. *SIAM Journal on Scientific and Computing* 2000; **21**:2331–2351.
16. Huang L. Numerical solution of the unsteady incompressible Navier–Stokes equations on the curvilinear half-staggered mesh. *Journal of Computational Mathematics* 2000; **18**:521–540.
17. Demirdžić I, Gosman AD, Issa RI. *A Finite-Volume Method for the Prediction of Turbulent Flow in Arbitrary Geometries*. Lecture Notes in Physics, vol. 141. Springer: Berlin, 1980; 144–150.
18. Yang HQ, Yang KT. Buoyant flow calculations with non-orthogonal curvilinear co-ordinates for vertical and horizontal parallelepiped enclosures. *International Journal for Numerical Methods in Engineering* 1998; **25**: 331–345.
19. Ikohagi T, Shin BR, Daiguji H. Application of an implicit time-marching scheme to a three-dimensional incompressible flow problem in a curvilinear coordinate systems. *Computers and Fluids* 1992; **21**:163–175.
20. Demirdžić I, Gosman AD, Issa RI, Perić M. A calculation procedure for turbulent flow in complex geometries. *Computers and Fluids* 1987; **13**:251–273.
21. Koshizuka S, Oka Y, Kondo S. A staggered differencing technique on boundary-fitted curvilinear grids for compressible flows along curvilinear of slant walls. *Computational Mechanics* 1990; **7**:123–136.
22. Oosterlee CW, Wesseling P, Segal A, Brakkee E. Benchmark solution for the incompressible Navier–Stokes equations in general co-ordinates on staggered grids. *International Journal for Numerical Methods in Fluids* 1993; **17**:301–321.
23. Zijlema M, Segal A, Wesseling P. Invariant discretization of the k–ε model in general co-ordinates for prediction of turbulent flow in complicated geometries. *Computers and Fluids* 1995; **24**:209–225.
24. Zijlema M, Segal A, Wesseling P. Finite volume computation of incompressible turbulent flows in general co-ordinates on staggered grids. *International Journal for Numerical Methods in Fluids* 1995; **20**:621–640.
25. Rosenfeld M, Kwak D. Time-dependent solutions of viscous incompressible flows in moving co-ordinates. *International Journal for Numerical Methods in Fluids* 1991; **13**:1311–1328.
26. Rosenfeld M, Kwak D, Vinokur M. A fractional-step method for unsteady incompressible Navier–Stokes equations in generalized coordinate system. *Journal of Computational Physics* 1991; **94**:102–1037.
27. Rosenfeld M. Multigrid acceleration of a fractional-step solver in generalized curvilinear coordinate systems. *AIAA Journal* 1993; **31**:1792–1800.
28. Kiris C, Kwak D. Numerical solution of incompressible Navier–Stokes equations using a fractional-step approach. *Computers and Fluids* 2001; **30**:829–851.
29. Kiris C, Kwak D. Aspects of unsteady incompressible flow simulations. *Computers and Fluids* 2002; **31**:627–638.

DOI: 10.1002/fld

30. Wesseling P, Segal A, Kassels CGM. Computing flows on general three-dimensional nonsmooth staggered grids. *Journal of Computational Physics* 1999; **149**:332–362.
31. Karki KC. A calculation procedure for viscous flows at all speeds in complex geometries. *Ph.D. Thesis*, University of Minnesota, Minneapolis, MN, 1986.
32. Shklyar A, Arbel A, Sokolov M. Navier–Stokes analyses of a supersonic ejector. *Proceeding of the 27th Israel Conference on Mechanical Engineering*, Haifa, Israel, 1998; 301–303.
33. Shklyar A, Arbel A. An application of the turbulence model to predict air flow and heat transfer in a greenhouse. *Proceeding of the 26th Israel Conference on Mechanical Engineering*, Haifa, Israel, 1996; 309–311.
34. Hall CA, Peterson JC, Porsching TA, Sledge FR. The dual variable method for finite element discretization of Navier–Stokes equations. *International Journal for Numerical Methods in Engineering* 1985; **21**:883–901.
35. Nicolaides RA. Direct discretization of planar div-curl problems. *ICASE Report*, 1989; 76–89.
36. Perić M. Analysis of pressure velocity coupling on non-staggered grids. *Numerical Heat Transfer*, *Part B* 1990; **17**:63–82.
37. Cho MJ, Chung MK. New treatment of non-orthogonal terms in the pressure-correction equation. *Numerical Heat Transfer*, *Part B* 1994; **26**:133–145.
38. Wang Y, Komori S. Comparison of using Cartesian and covariant velocity components on non-orthogonal collocated grids. *International Journal for Numerical Methods in Fluids* 1999; **31**:1265–1280.
39. Wang Y, Komori S. On the improvement of the SIMPLE-like method for flow with complex geometry. *Heat and Mass Transfer* 2000; **36**:71–78.
40. Maliska CR, Raithby GD. A method for computing three-dimensional flows using non-orthogonal boundary-fitted co-ordinates. *International Journal for Numerical Methods in Fluids* 1984; **4**:519–537.
41. Lehnhäuser T, Schäfer M. Efficient discretization of pressure-correction equations on non-orthogonal grids. *International Journal for Numerical Methods in Fluids* 2003; **42**:211–231.
42. Napolitano M, Pascazio G, Quartapello L. A review of vorticity conditions in the numerical solution of the k–ε equations. *Computers and Fluids* 1999; **28**:139–185.
43. Demirdžić I, Lilek Z, Perić M. Fluid flow and heat transfer test problems for non-orthogonal grids: benchmark solutions. *International Journal for Numerical Methods in Fluids* 1992; **15**:329–354.
44. Fletcher CAJ. *Computational Techniques for Fluid Dynamics*. Springer: Berlin, 1988.
45. Gresho PM, Sani RL. On pressure boundary conditions for the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**:1111–1145.
46. Abdallah S. Numerical simulations for the pressure Poisson equations with Neumann boundary conditions using a non-staggered grid. 1. *Journal of Computational Physics* 1987; **70**:182–192.
47. Claeyssen JR, Platte RB, Bravo E. Simulation in primitive variables of incompressible flow with pressure Neumann condition. *International Journal for Numerical Methods in Fluids* 1999; **30**:1009–1026.
48. Kuehn TH, Goldstein RJ. An experimental study of natural convection in the annulus between horizontal concentric cylinders. *Journal of Fluid Mechanics* 1976; **74**:695–719.
49. Kuehn TH, Goldstein RJ. An experimental study of natural convection heat transfer in and eccentric annuli. *Journal of Heat Transfer* (ASME) 1978; **100**:635–640.
50. Guj G, Iannetta S, Moretti G. Experimental analysis thermal fields horizontal eccentric cylindrical annuli. *Experiments in Fluids* 1992; **12**:385–393.
51. Farouk B, Geceri SI. Laminar and turbulent natural convection in the annulus between horizontal concentric cylinders. *Journal of Heat Transfer* (ASME) 1982; **104**:631–636.
52. Cho CH, Chang KS, Park KH. Numerical simulation of natural convection in concentric and eccentric horizontal cylindrical annuli. *Journal of Heat Transfer* (ASME) 1982; **104**:624–630.
53. Projahn U, Rieger H, Beer H. Numerical analysis of laminar natural convection between concentric and eccentric cylinders. *Numerical Heat Transfer* 1981; **4**:131–146.
54. Shu C, Yao Q, Yeo KS, Zhu YD. Numerical analysis of flow and thermal fields in arbitrary eccentric annulus by differential quadrature method. *Heat and Mass Transfer* 2002; **38**:597–608.
55. Date AW. Numerical prediction of natural convection heat transfer in horizontal annulus. *International Journal of Heat and Mass Transfer* 1986; **29**:1457–1464.
56. Guj G, Stella F. Natural convection in horizontal eccentric annuli: numerical study. *Numerical Heat Transfer*, *Part A* 1995; **27**:89–105.
57. Karki KC, Patankar SV. Solution of some two dimensional incompressible flow problems using a curvilinear coordinate system based calculation procedure. *Numerical Heat Transfer* 1988; **14**:309–321.
58. Choi SK, Nam HY, Cho M. Use of staggered and non-staggered grid arrangements for incompressible flow calculations on nonorthogonal grids. *Numerical Heat Transfer*, *Part B* 1994; **25**:193–204.

DOI: 10.1002/fld

59. Kobayashi MH, Pereira JCF. Calculations of incompressible laminar flow on a nonstaggered nonorthogonal grid. *Numerical Heat Transfer*, *Part B* 1991; **19**:243–262.
60. Ray S, Date AW. A calculation procedure for solution of incompressible Navier–Stokes equations on curvilinear non-staggered grids. *Numerical Heat Transfer*, *Part B* 2000; **38**:93–131.
61. Galpin PF, Raithby GD. Numerical solution of problem in incompressible fluid flow: treatment of the temperature–velocity coupling. *Numerical Heat Transfer* 1986; **10**:105–129.
62. Hageman LA, Young DM. *Applied Iterative Methods.* Academic Press: New York, 1981.
63. Golub GH, Van Loan C. *Matrix Computation*. The John Hopkins University Press: Baltimore, 1996.
64. Vuik C, Saghir A, Boerstoler GP. The Krylov accelerated SIMPLE(R) method for flow problems in industrial furnaces. *International Journal for Numerical Methods in Fluids* 2000; **33**:1027–1040.
65. Ashcraft CC, Grmes RG. On vectorization incomplete factorization and SSOR preconditioners. *SIAM Journal on Scientific and Statistical Computing* 1988; **9**:122–151.
66. Buleev NI. A numerical method for the solution of two-dimensional and three-dimensional equations of diffusion. *Mat. Sb.* (in Russian) 1960; **51**:227–238.
67. Stone HL. Iterative method of implicit approximation of multidimensional partial differential equations. *SIAM Journal on Numerical Analysis* 1968; **51**:530–558.
68. Dupont T, Kendall RP, Rachford HH. An approximate factorization procedure for solving self-adjoint elliptic difference equations. *SIAM Journal on Numerical Analysis* 1968; **5**:559–573.
69. Gustafsson I. A class of first-order factorization methods. *BIT* 1978; **18**:142–156.
70. Saad Y. *Iterative Method for Sparse Linear System*. PWS Publishing Company: New York, 1996.
71. Oliveira PJ, Issa RI. An improved PISO algorithm for the computation of buoyancy-driven flows. *Numerical Heat Transfer*, *Part B* 2001; **40**:473–493.
72. Forsythe GE, Wasow WR. *Finite-Difference Method for Partial Differential Equations*. Wiley: New York, 1960.

DOI: 10.1002/fld