Assessments of SIMPLE and ASIMPLE Algorithms Based on Buoyancy-Driven Cavity Flows

Jian Qin[1], Huachen Pan[1], Zefei Zhu[1], Xiaoqing Tian[1], and M. M. Rahman[*][1]
[1]Hangzhou Dianzi University, School of Mechanical Engineering, 310018 Hangzhou, China

Abstract:- A comparative assessment between SIMPLE and artificial SIMPLE (ASIMPLE) algorithms is conducted based on two-dimensional buoyancy-driven incompressible cavity flows, using a cell-centered finite-volume formulation on a non-orthogonal collocated grid. Both methods are characteristically pressure-based; however, the ASIMPLE scheme additionally combines the concept of artificial compressibility with the pressure Poisson equation, provoking density perturbations that account for the transformation between primitive and conservative variables. An improved non-linear momentum interpolation scheme is employed at the cell face in discretizing the continuity equation, suppressing pressure oscillations effectively. A range of values is considered for the thermal Grashof number; excellent consistency is obtained between results available in the literature and numerical solutions adhering to both SIMPLE and ASIMPLE solvers. Numerical experiments in reference to buoyancy-driven cavity flows dictate that both contrivances (e.g., SIMPLE and ASIMPLE) execute a residual smoothing enhancement, facilitating an avoidance of the velocity/pressure under-relaxation (UR). However, compared with the SIMPLE approach, included benefits of the ASIMPLE method are the use of larger Courant numbers, enhanced robustness and convergence. Both procedures adopt an unfactored pseudo-time integration scheme and provide identical results.

Keywords:- SIMPLE Algorithm, Artificial Compressibility, Sound Speed, Under-Relaxation (UR) Factor, Convergence and Robustness.

➤ *Nomenclature*

C = artificial sound speed
CFL = Courant number
F, G = flux vectors in x and y directions
Gr = Grashof number
K = viscous flux or dissipation coefficient
L = characteristic length
$\dot{M}^*$ = fictitious mass source
p = static pressure
p' = pressure correction or artificial density-change
p^* = tentative pressure
Pr = Prandtl number
Q = source term
S = cell-face area
t = time
T = temperature

u, v = velocity components in x and y directions
u', v' = velocity corrections u^*, v^* = tentative velocities
U, V = contravariant or dimensionless velocity components
W = conservative variable vector
x, y = Cartesian coordinates
Γ = diffusion coefficient
η = cell-face damping factor
θ = dimensionless temperature
ν = kinematic viscosity
ρ = density
φ = scalar transport variable
$\forall$ = cell volume

Subscript
ref = reference value
nb = neighboring grid point

I. INTRODUCTION

Considerable research has been devoted to formulating Navier-Stokes (NS) equations, with low Mach number and incompressible flows wherein ideas of artificial compressibility (AC) are exploited [1–6]. The density-based (e.g., compressible flow) and the pressure-based (e.g., primitive variable formulation) methods are usually used in solving the NS equations. In principle, the AC is usually well-known as a preconditioned compressible flow scheme [1].

Conventionally, the AC approach applies a time-derivative of pressure with the continuity equation together with a multiplicative parameter recognized as an AC [6], allowing the mass continuity to be prosecuted in a time-marching manner, approximately analogous to the momentum equations. However, the AC term plays the key role in the evaluation of a credible success and hence, the choice of AC variable related to the artificial speed of sound has a significant impact on accuracy, stability and convergence of numerical methods. In essence, an enhancement in the convergence rate of numerical solver, requires the AC parameter to be automated in the computational domain.

On the other hand, the pressure-based method, assigned to the pressure-velocity coupling for incompressible flows assembles a number of consecutively progressive algorithms [7-20]. The methodology herein is to devise a second-order Poisson equation for the pressure-correction resorting to the continuity and momentum equations, updating afterwards the pressure and velocity fields until the mass continuity is satisfied. This

deterministic coupling together with the pressure or velocity under-relaxation (UR) is presumably the main reason for a slow convergence associated with the SIMPLE (Semi-Implicit Methods for Pressure Linked Equations) algorithm and its enhanced variants [10, 21, 22]. However, the physical motivation of pressure-based method is to muster assertively an excluded decoupling of pressure-velocity fields. Essentially, the major task herein is to introduce an accurate amount of cell-face dissipation to eliminate the destabilizing effect arising from the pressure checker-boarding. A lot of innovative formulations using primitive variables have been constructed to surpass the difficulties embedded with collocated grid arrangements [17-25].

The SIMPLE algorithm as a pressure-correction method is the most common algorithm; it has been used in most commercial software, put forwarded by Patankar and Spalding [7, 8]. Patankar formulated the SIMPLER (SIMPLE REVISED SIMPLE) algorithm [9]; it uses the velocity field obtained from the hypothetical or previous iteration and an intermediate pressure field directly, instead of assuming the pressure field. Van Dooormaal proposed SIMPLEC (SIMPLE Consistent) algorithm [10], which partially accounts for the influence of surrounding velocity nodes, and thus obtaining the improved velocity correction equation. Spalding [11], and Markatos and Pericleous [12] proposed the SIMPLEST algorithm; Issa et al. [13] and Issa [14] proposed PISO algorithm, which includes one prediction step and two corrector steps. In addition, ZhiGuo proposed the CLEAR (Coupled and linked equation algorithm) [15, 16] and Cheng proposed CLEARER algorithm [17]. Rahman and Siikonen proposed SIMPLE algorithm accompanied by an AC where the continuity equation adopts AC form [18]. Rahman et al. [19, 20] also improved the pressure-correction method. Compared with the traditional SIMPLE algorithm, the robustness and convergence of modified SIMPLE algorithm are greatly improved.

In particular, on a staggered/collocated grid arrangement, the SIMPLE-like procedure commonly considers the nodal momentum contributions to design influence and cell-face dissipation coefficients in the pressure-correction equation. These formulas may introduce relatively low weights in the influence coefficients, together with an inappropriate mass imbalance, which often overestimates the pressure-correction in pressure-based algorithms identified by acronyms such as SIMPLE, SIMPLER, SIMPLC, etc [7–10]. To counteract this adverse situation, the pressure-based method eventually confronts pressure or velocity UR, indulging in a progressively worse convergence-rate in the case of flows with strong source terms. Nevertheless, with the finite-volume Δ -formulation, this well-recognized slow convergence aspect associated with the SIMPLE approach could be reduced to a greater extent. In this formulation, the implicit stage conserves the explicit density residual $\Delta\rho$ (e.g., corrected mass imbalance in the SIMPLE method) as a consequence of applying the primitive rather than conservative variables. Intuitively, $\Delta\rho$ is explored in order to smoothen momentum residuals, resulting in presumably an avoidance of UR factor in the SIMPLE algorithm.

To this end, it must be stressed that an additional improvement is accounted for by com-bining the pressure Poisson equation of SIMPLE method with an AC which is conducive to magnifying the diagonal dominance of influence coefficients. The artificial SIMPLE (ASIM-PLE) scheme envisages density perturbations, appearing exclusively at the transformation between primitive and conservative variables. A revised nonlinear cell-face interpolation scheme [20] is applied to eliminate nonphysical pressure oscillations. Governing equations are solved in a segregated fashion, relying on diagonal dominance for convergence. The overall architecture dramatically induces a residual smoothing enhancement, thereby facilitating an avoidance of velocity or pressure UR. Buoyancy-driven cavity flows could be appropriate numerical experiments [26–30] to validate the efficacy and accuracy of both SIMPLE and ASIMPLE algorithms.

The ASIMPLE formula is an improved version of SIMPLE algorithm developed by Rahman and Siikonen [18] wherein the impact of AC is limited to momentum equations and the cell-face velocity interpolation technique devised by Rahman et al. [20] is modified to suppress the local extrema into the tentative cell-face velocity; entailing essentially a prerequisite to controlling the velocity or pressure UR associated with the SIMPLE method. Note-worthily, such a SIMPLE algorithm has not, to the authors knowledge, to date been formulated to abandon UR factor when numerically computing flows with dominant source terms in the interest of convergence and robustness, pertaining to the SIMPLE scheme.

II. GOVERNING EQUATION

Two-dimensional (2D) convection-diffusion equations with an inclusion of a scalar variable ϕ can be represented in the following form:

$$\frac{\partial W}{\partial t} + \frac{\partial (F_{inv} - F_{ivs})}{\partial x} + \frac{\partial (G_{inv} - G_{vis})}{\partial y} = Q$$

where $W = (\rho, \rho u, \rho v, \rho\phi)^T$ and Q signifies the source term. Inviscid fluxes are:

$$F_{inv} = \begin{bmatrix} \rho u \\ \rho u^2 + \rho \\ \rho uv \\ \rho u\phi \end{bmatrix}, G_{inv}$$

$$= \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho v\phi \end{bmatrix} \qquad (2)$$

In Equation (2), ρ is the density, u and v are the Cartesian velocity components and p implies the pressure. Viscous fluxes can be expressed as:

$$F_{vis} = \begin{bmatrix} 0 \\ \Gamma(\partial u/\partial x) \\ \Gamma(\partial v/\partial x) \\ \Gamma(\partial \phi/\partial x) \end{bmatrix}, \qquad G_{vis}$$

$$= \begin{bmatrix} 0 \\ \Gamma\left(\frac{\partial u}{\partial y}\right) \\ \Gamma\left(\frac{\partial v}{\partial y}\right) \\ \Gamma\left(\frac{\partial \phi}{\partial y}\right) \end{bmatrix} \qquad (3)$$

Γ denotes the diffusion coefficient, e.g., either viscosity or thermal conductivity of the fluid.

III. FINITE-VOLUME DISCRETIZATION WITH Δ-FORMULATION

The governing equations are discretized using a cell-centered finite-volume scheme, having an integral form

$$\frac{d}{dt}\int_{\forall} W\, dv + \int_s F(W)\cdot dS$$
$$= \int_{\forall} Q\, d\forall \qquad (4)$$

for an arbitrary region $\forall$ with a boundary S. Carrying out the integration for a computational cell i yields:

$$\forall_i \frac{dW_i}{dt}$$
$$= \sum_{faces} -S\hat{F}$$
$$+ \forall_i Q_i \qquad (5)$$

where the summation is taken over the computational cell faces; each surface has a unit normal vector **n**, defined by
$$\mathbf{n} = n_x i + n_y j$$
$$= \frac{S_x}{S} i$$
$$+ \frac{S_y}{S} j \qquad (6)$$
with the corresponding cell-face flux:
$$\hat{F}$$
$$= n_x F$$
$$+ n_y G \qquad (7)$$
where fluxes F and G are defined by Equations (2) and (3), respectively.

In principle, the development of an algebraic equation with the Δ-formulation by Rahman and Siikonen [18], linking the dependent variable at the cell-center to neighboring dependent variables is reproduced herein to accommodate better understanding of the current phenomena. Applying a first-order backward difference to the variation in W, the implicit time integration adhering to Equation (5) can be obtained as:

$$\forall_i \frac{\Delta W}{\Delta t} + H^{n+1}_{i+1/2} - H^{n+1}_{i-\frac{1}{2}} + H^{n+1}_{j+\frac{1}{2}} - H^{n+1}_{j-\frac{1}{2}}$$
$$= \forall_i Q_i \qquad (8)$$

where $H = S\hat{F}$, $\Delta W = W^{n+1} - W^n$ and the term H^{n+1} is linearized with respect to t as:

$$H^{n+1} \approx H^n + \frac{\partial H}{\partial W}\Delta W$$

Substituting Equation (9) into Equation (8) provides an implicit pseudo-time step relation after some manipulations:

$$\Delta \widetilde{W}_i + \frac{\Delta t_i}{V_i}(U\Delta\widetilde{W}S - K\partial\Delta\widetilde{W})^{i+\frac{1}{2}}_{i-\frac{1}{2}} + \frac{\Delta t_i}{\Delta\forall_i}(V\Delta\widetilde{W}S$$
$$- K\partial\widetilde{W})^{j+\frac{1}{2}}_{j-\frac{1}{2}} = \Delta\widetilde{W}_i^* \qquad (10)$$

In the above-mentioned relation:
$$\Delta\widetilde{W}_i^* = \frac{1}{\rho_i}(\Delta W_i - \widetilde{W}_i\Delta\rho_i), \qquad \Delta W_i = \frac{\Delta t_i}{\forall_i}R_i, \quad \widetilde{W} = (u, v, \phi)^T$$

$$R_i$$
$$= -\left(H^{n+1}_{i+\frac{1}{2}} - H^{n+1}_{i-\frac{1}{2}} + H^{n+1}_{j+\frac{1}{2}} - H^{n+1}_{j-\frac{1}{2}}\right)$$
$$+ \forall_i Q_i \qquad (12)$$

where $\partial i_{\pm 1/2}$ and $\partial j_{1\pm 1/2}$ are the first-order spatial difference operators and
$$U_{i\pm 1/2} = (un_x + vn_y)_{i\pm\frac{1}{2}} \qquad V_{j\pm\frac{1}{2}}$$
$$= (un_x + vn_y)_{j\pm\frac{1}{2}} \qquad (13)$$

are the contravariant velocity components in the i-direction and j-direction, respectively. The implicit stage preserves the explicit density residual Δρ (e.g., mass imbalance generated by corrected velocity fields) as a result of using the primitive rather than conservative variables. Evidently, Δρ is explored in order to linearize the residuals, resulting in presumably an avoidance of the velocity or pressure UR which is an expensive component in the SIMPLE algorithm. The elimination of UR factor dramatically improves numerical convergence and provides a significant reduction in the computational time.

In particular, a fully upwind second-order (FUS) difference is utilized to approximate the convective flux residual on the right-hand side of Equation (10). A stable solution with the FUS scheme is obtained since it applies only upwind terms to extrapolate the cell face value [30]. To account for the directional impact on the upwinding process, the inviscid flux at the cell face $(i + 1/2)$ associated with R_i in Equation (12) is evaluated using a rotational matrix [18]. A standard first-order upwind is employed to evaluate the coefficient matrix. Therefore, the convective flux on the cell face $(i + 1/2)$ for the implicit part can be given as:

$$(\dot{M}\Delta\widetilde{W})_{i+\frac{1}{2}}$$
$$= \dot{M}^+_{i+\frac{1}{2}}\Delta\widetilde{W}_i$$
$$- \dot{M}^-_{i+\frac{1}{2}}\Delta\widetilde{W}_{i+1} \qquad (14)$$

where $\dot{M}$ is the volume flow-rate across the cell face, defined as follows:

$$\dot{M}_{i+1/2} = (US)_{i+\frac{1}{2}}, \quad \dot{M}^{+}_{i+\frac{1}{2}} = max(\dot{M}_{i+\frac{1}{2}}, 0), \quad \dot{M}^{-}_{i+\frac{1}{2}} = max\left(-\dot{M}_{i+\frac{1}{2}}, 0\right) \quad (15)$$

Similar approximations are valid for other cell faces.

Collecting together relations like Equations (14) and (15) in Equation (10), a system of algebraic equations can be obtained as:

$$A_i \Delta \widetilde{W}_i = \sum_{nb} A_{nb} \Delta \widetilde{W}_{nb} + \Delta \widetilde{W}_i^* \quad (16)$$

with

$$A_i = 1 + \sum_{nb} A_{nb} + |\dot{M}_i| \quad (17)$$

$$\sum_{nb} A_{nb} = A_{i+1} + A_{i-1} + A_{j+1} + A_{j-1}$$

$$\dot{M}_i = \frac{\Delta t_i}{\forall_i}\left(M_{i+1/2} + M_{i-1/2} + M_{j+1/2} + M_{j-1/2}\right)$$

where

$$A_{i+1} = \frac{\Delta t_i}{\forall_i}\left(K_{i+\frac{1}{2}} + \dot{M}^{-}_{i+\frac{1}{2}}\right) \quad A_{i-1} = \frac{\Delta t_i}{\forall_i}\left(K_{i-\frac{1}{2}} + \dot{M}^{+}_{i-\frac{1}{2}}\right) \quad (18)$$

$$A_{j+1} = \frac{\Delta t_j}{\forall_j}(K_{j+\frac{1}{2}} + \dot{M}^{-}_{j+\frac{1}{2}}) \quad A_{j-1} = \frac{\Delta t_j}{\forall_j}(K_{j-\frac{1}{2}} + \dot{M}^{+}_{j-\frac{1}{2}})$$

$$K_{i\pm1/2} = \left(\frac{\Gamma S}{\rho \Delta n}\right)_{i\pm1/2}, \qquad \Delta n_{i\pm\frac{1}{2}} = \frac{\forall_i + \forall_{i\pm1/2}}{2S_{i\pm\frac{1}{2}}}$$

$$K_{j\pm1/2} = \left(\frac{\Gamma S}{\rho \Delta n}\right)_{j\pm1/2}, \qquad \Delta n_{j\pm\frac{1}{2}} = \frac{\forall_i + \forall_{i\pm1/2}}{2S_{j\pm\frac{1}{2}}}$$

where the diffusion term K is approximated using a symmetric linear profile, $\Delta n_{i\pm1/2}$ and $\Delta n_{j\pm1/2}$ are distances between cell centers in the i-and j-directions, respectively; nb stands for a run over neighboring nodes $(i+1)$, $(i-1)$, $(j+1)$, and $(j-1)$. After the implicit stage, the solution vector $\widetilde{W}$ is updated from:

$$\widetilde{W}_i^{n+1} = \widetilde{W}_i^n + \Delta \widetilde{W}_i \quad (19)$$

Equation (19) represents tentative velocity components with an arbitrary pressure field.

A. Artificial SIMPLE (ASIMPLE) method

In the SIMPLE algorithm, converged solutions to mass and momentum conservation laws can be achieved after repeated iterations for tentative velocity and pressure fields. The velocity and pressure fields are corrected as (for convenience, only u and p are considered):

$$u = u^* + u' \quad (20)$$

$$p = p^* + p' \quad (21)$$

where u' and p' represent incremental velocity and pressure, respectively. Quantities u^* and p^* are the tentative values. The velocity-correction can be linked to the pressure-correction on a curvilinear coordinate system by [18]:

$$u_i = u_i^* - \frac{\Delta t_i'}{\rho}\frac{(Sn_x\partial p')_i + (Sn_x\partial p')_j}{\forall_i} \quad (22)$$

with

$$\frac{(Sn_x\partial p')_i}{\forall_i} \approx \frac{(n_x p')_{i+\frac{1}{2}} - (n_x p')_{i-\frac{1}{2}}}{\Delta n_i}$$

$$\frac{(Sn_x\partial p')_j}{\forall_i} \approx \frac{(n_x p')_{j+\frac{1}{2}} - (n_x p')_{j-\frac{1}{2}}}{\Delta n_j} \quad (24)$$

where $\Delta t_i'$ is a fictitious time step (e.g., a weighting factor); Δn_i and Δn_j are the control-volume cell thicknesses in i and j directions, respectively. The cell face value of pressure-correction p' is obtained as an average from two adjacent nodal points.

Within the framework of traditional AC method, the continuity equation is modified by introducing an artificial time-derivative of pressure [18]:

$$\frac{1}{C^2}\frac{\partial p}{\partial t} + \rho\frac{\partial u}{\partial x} + \rho\frac{\partial v}{\partial y} = 0 \quad (25)$$

where C is the artificial sound speed, optimized numerically as :

$$C = \sqrt{max\left[(u^2 + v^2 + \beta), \frac{1}{2}U_{ref}^2\right]} \quad (26)$$

where $\beta = 10$ and U_{ref} is the reference velocity.

The finite-volume discretization of Equation (25) results in:

$$\frac{\forall_i}{C_i^2\Delta t_i}p_i' + (\rho U^*S)_{i-\frac{1}{2}}^{i+\frac{1}{2}} + (\rho V^*S)_{j-\frac{1}{2}}^{j+\frac{1}{2}} = 0 \quad (27)$$

Substituting relations like Equation (22) into Equation (27) leads to a truncated pressure-correction relation:

$$B_i p_i' = \sum_{nb} B_{nb} p_{nb}' - \dot{M}_i^* \quad (28)$$

where

$$B_i = \frac{\forall_i}{C_i^2 \Delta t_i} + \sum_{nb} B_{nb} \quad \sum_{nb} B_{nb}$$

$$= B_{i+1} + B_{i-1} + B_{j+1} + B_{j-1} \qquad (29)$$

$$= \left(\frac{\Delta t' S}{\Delta n}\right)_{i+\frac{1}{2}}$$

$$B_{i-1} = \left(\frac{\Delta t' S}{\Delta n}\right)_{i-\frac{1}{2}} \quad B_{j+1} = \left(\frac{\Delta t' S}{\Delta n}\right)_{j+\frac{1}{2}} \qquad B_{j-1}$$

$$= \left(\frac{\Delta t' S}{\Delta n}\right)_{j-\frac{1}{2}}$$

where $\dot{M}_i^*$ represents the fictitious mass source, given by:

$$\dot{M}_i^*$$

$$= \dot{M}_{i+\frac{1}{2}}^* - \dot{M}_{i-\frac{1}{2}}^* + \dot{M}_{j+\frac{1}{2}}^* + \dot{M}_{j-\frac{1}{2}}^* \qquad (30)$$

$$\dot{M}_{i+\frac{1}{2}}^* = (\rho U^* S)_{i+\frac{1}{2}} \quad \dot{M}_{i-\frac{1}{2}}^* = (\rho U^* S)_{i-\frac{1}{2}}$$

$$\dot{M}_{j+\frac{1}{2}}^* = (\rho U^* S)_{j+\frac{1}{2}} \quad \dot{M}_{j-\frac{1}{2}}^* = (\rho U^* S)_{j-\frac{1}{2}}$$

Note that cross-diffusion fluxes in Equation (28) are neglected with a thin-layer-type approximation, frequently used due to maintaining diagonally dominant coefficient matrices [18, 19]. Obviously, the physical relevance of the first term on the left-hand side in Equation (25) is to add more weight on the diagonal of nodal coefficients, allowing the use of a higher CFL number and thereby, enhancing the convergence acceleration. Conspicuously, the SIMPLE algorithm is obtained with an exclusion of the first term from Equation (29).

B. Time step and weighting factor

Time step Δt define is evaluated from the relation [18]:

$$\Delta t_i = min \left[\frac{CFL \forall_i}{(|U|S)_i (|V|S)_j + C_i \sqrt{S_i^2 + S_j^2}}, \frac{L^2}{2k v_i} \right] \quad L$$

$$= min(\Delta n_i, \Delta n_j) \qquad (31)$$

where $k = 2$. Grid properties at (i, j) are interpolated from cell faces. Values of CFL in the range 1-20 are recommended by the current implicit solvers, accommodating better convergence to the steady state at which the fictitious/artificial mass conservation is enforced.

The weighting factor $\Delta t'$ involved in the velocity-pressure correction equation may be evaluated in many ways [19]. In the present study, $\Delta t'$ is determined from Equation (17) as:

$$\Delta t_i' = min(1, CFL) \frac{\Delta t_i}{A_i} \qquad A_i$$

$$= \sum_{nb} A_{nb} \qquad (32)$$

where the coefficient A_i emerges naturally from a consequence of the momentum equations. The choice of

$\Delta t'$ is an additional criterion to eradicate the presence of UR factor in the SIMPLE-like algorithm.

C. Cell face velocity

On a collocated grid, a linear approximation for the cell-face velocity induces a checkerboard pressure mode, reflecting the pressure-velocity decoupling. Unfortunately, the discrete form of momentum equations cannot detect the unrealizable components, remaining until the iterative process converges. To avoid this occurrence, an improved Rhie-Chow [24] interpolation method developed by Rahman and Siikonen [20] is adopted. To revive the formulation explicitly, the calculation of $u_{i-1/2}^*$ is discussed herein. Using Equation (17) the velocity components u_{i-1}^*, u_i^* and $u_{i-1/2}^*$ comply with the discretized momentum equations:

$$u_{i-1}^* + \frac{\forall_{i-1}}{A_{i-1}} \left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-1} = \frac{\sum_{i-1} A_{nb} u_{nb}^*}{A_{i-1}}$$

$$u_i^* + \frac{\forall_i}{A_i} \left(\frac{\partial p^*}{\partial x} - f^u\right)_i = \frac{\sum_i A_{nb} u_{nb}^*}{A_i}$$

$$u_{i-1/2}^* + \frac{\forall_{i-1}}{A_{i-1/2}} \left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-1/2} = \frac{\sum_{i-1/2} A_{nb} u_{nb}^*}{A_{i-1/2}}$$

where $f_i^u, f_{i-1/2}^u$ and f_{i-1}^u represents the non-pressure gradient source terms. The right-hand side of Equation (35) can be obtained using a linear interpolation of corresponding terms in Equations (33) and (34) as follows :

$$u_{i-1/2}^* = \bar{u}_{i-1/2}^* + \frac{1}{2}\left[\frac{\forall_{i-1}}{A_{i-1}}\left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-1} + \frac{\forall_i}{A_i}\left(\frac{\partial p^*}{\partial x} - f^u\right)_i\right] - C_{i-\frac{1}{2}}\left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-\frac{1}{2}} \quad (36)$$

with

$$u_{i-1/2}^* = \frac{1}{2}(u_{i-1}^* + u_i^*) \quad C_{i-1/2} = \frac{\forall_{i-1/2}}{A_{i-1/2}}$$

$$= \frac{1}{2}\left(\frac{\forall_{i-1}}{A_{i-1}} + \frac{\forall_i}{A_i}\right)_{i-1}$$

At point $(i - 1/2)$, Taylor series expansion can be used to obtain the following formulas:

$$\left(\frac{\partial p^*}{\partial x}\right)_i$$

$$= \left(\frac{\partial p^*}{\partial x}\right)_{i-1/2} + \frac{\Delta x_{i-1/2}}{2}\left(\frac{\partial^2 p^*}{\partial x^2}\right)_{i-1/2}$$

$$+ \frac{\Delta x_{i-1/2}^2}{8}\left(\frac{\partial^3 p^*}{\partial x^3}\right)_{i-1/2} + \cdots \quad (37)$$

$$f_i^u = f_{i-1/2}^u + \frac{\Delta x_{i-1/2}}{8}\left(\frac{\partial f^u}{\partial x}\right)_{i-1/2} + \cdots$$

(38)

$$\left(\frac{\partial p^*}{\partial x}\right)_{i-1}$$

$$= \left(\frac{\partial p^*}{\partial x}\right)_{i-1/2} + \frac{\Delta x_{i-1/2}}{2}\left(\frac{\partial^2 p^*}{\partial x^2}\right)_{i-1/2}$$

$$+ \frac{\Delta x_{i-1/2}^2}{8}\left(\frac{\partial^3 p^*}{\partial x^3}\right)_{i-1/2} + \cdots \quad (39)$$

$$f_{i-1}^u$$

$$= f_{i-1/2}^u - \frac{\Delta x_{i-\frac{1}{2}}}{8}\left(\frac{\partial f^u}{\partial x}\right)_{i-\frac{1}{2}}$$

$$+ \cdots \qquad (40)$$

Multiplying Equations (37) and (38) by $\forall_i / A_i$, and Equations (39) and (40) by $\forall_{i-1} / A_{i-1}$ yields:

$$\frac{1}{2}\left[\frac{\forall_{i-1}}{A_{i-1}}\left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-1} + \frac{\forall_i}{A_i}\left(\frac{\partial p^*}{\partial x} - f^u\right)_i\right]$$

$$- C_{i-\frac{1}{2}}\left(\frac{\partial p^*}{\partial x} - f^u\right)_{i-\frac{1}{2}}$$

$$= -\frac{\Delta x_{i-\frac{1}{2}}}{4}\left(\frac{\forall_i}{A_i} - \frac{\forall_{i-1}}{A_{i-1}}\right)\left(\frac{\partial f^u}{\partial x}\right)_{i-\frac{1}{2}}$$

$$+ \left|\frac{\Delta x_{i-\frac{1}{2}}^2 C_{i-\frac{1}{2}}}{8}\left(\frac{\partial^3 p^*}{\partial x^3}\right)_{i-\frac{1}{2}}\right. \qquad (41)$$

assuming that the pressure at the cell face is obtained from a linear interpolation between grid nodes. Therefore, the impact of the following term

$$\frac{\Delta x_{i-\frac{1}{2}}}{4}\left(\frac{\forall_i}{A_i}\right.$$

$$\left. - \frac{\forall_{i-1}}{A_{i-1}}\right)\left(\frac{\partial^2 p^*}{\partial x^3}\right)_{i-\frac{1}{2}}$$

on the overall scheme is insignificant. Combining Equation (41) with Equation (36) provides for:

$$u_{i-1/2}^*$$

$$= \bar{u}_{i-1/2}^* - \left(K\frac{\partial f^u}{\partial x}\right)_{i-\frac{1}{2}}$$

$$+ \left(R\frac{\partial^3 p^*}{\partial x^3}\right)_{i-\frac{1}{2}} \qquad (42)$$

where

$$K_{i-1/2} = \frac{\Delta x_{i-\frac{1}{2}}}{4}\left(\frac{\forall_i}{a_i} - \frac{\forall_{i-1}}{A_{i-1}}\right), \quad R_{i-1/2} = \frac{\Delta x_{i-\frac{1}{2}}^2 C_{i-\frac{1}{2}}}{8}$$

In the current research, Equation (42) is modified that guarantees no local extrema into the cell-face velocity. This is appropriately preserved by the following explicit reconstruction:

$$u_{i-1/2}^* = \bar{u}_{i-1/2}^* - \eta_{i-\frac{1}{2}}\left\{K_{i-\frac{1}{2}}\frac{\partial f^u}{\partial x}|_{i-\frac{1}{2}} - C_{i-\frac{1}{2}}\left[\frac{1}{2}\left(\frac{\partial p^*}{\partial x}|_{i-1} + \frac{\partial p^*}{\partial x}|_i\right) - \frac{\partial p^*}{\partial x}|_{i-\frac{1}{2}}\right]\right\} \qquad (43)$$

where

$$\eta_{i-1/2} = \frac{\zeta_{i-\frac{1}{2}}^2}{1 + \zeta_{i-\frac{1}{2}}^2}, \qquad \zeta_{i-\frac{1}{2}} = \frac{\Delta x_{i-\frac{1}{2}}\bar{u}_{i-\frac{1}{2}}^*}{\nu_{i-1/2}}$$

where ν is the kinematic viscosity. Discernibly, the damping term (can actually be interpreted as a correction to the central difference scheme) involved in Equation (43) has several desirable attributes when compared with the Rhie-Chow [24] cell-face interpolation scheme: (a) it is not dependent on UR factors; (b) the frame-work constitutes a compact formulation; (c) the included non-pressure

gradient source term influences the stability of solution [20] and (d) the magnitude of cell-face dissipation is bounded due to the presence of damping coefficient η. Thus, the formulation has a compatible competency in eliminating the checkerboard pressure mode in fluid flow and heat transfer problems with strong source terms.

D. Solution vector fields

Incorporating an AC parameter in the pressure-correction equation, invokes modified-correction phenomena in the actual SIMPLE algorithm, resulting in an ASIMPLE method. This is constructed such as to expedite an enhancement in the residual smoothing properties. Conservative corrections can be recovered by allowing an artificial change in density subjected to:

$$\rho'$$

$$= \frac{p'}{C^2} \qquad (44)$$

which leads to the following linearizations:

$$\Delta\rho u^* = \rho u' + \rho' u^*, \quad \Delta\rho v^*$$

$$= \rho v' + \rho' v^* \qquad (45)$$

Remarkably, the artificial density change ρ' implicitly accounts for the effect of fictitious mass imbalance. Corrected primitive variables are achieved as follows:

$$(u, v)^T$$

$$= (u^*, v^*)^T$$

$$+ \frac{1}{\rho}(\Delta\rho u^*, \Delta\rho v^*)^T \qquad (46)$$

Undoubtedly, p' can be recognized as the density preconditioning or perturbation to the incompressible limit, convoking an explicit definition of the fluid constitutive relation which leads to a transformation between conservative and primitive variables. Consequently, a consistent control over changes in dependent variables is obtained, resulting in an enhanced robustness of the solver. Comparing Equation (20) with Equation (46) advocates that the velocity corrections are modified; however, as the solution converges to the steady state, the density perturbation disappears. The physical rationale with the ASIMPLE approach is that it imitates the characteristics of pseudo-compressibility to solve the incompressible equations.

To this end, it is appropriate to emphasized herein that the system is pressure based. Compared with the ASIMPLE method proposed by Rahman and Siikonen , the current ASIMPLE algorithm involves the artificial density change with momentum equations only. Since the participation of p' in the scalar equation entangles non-physical scalar results when considering flow regimes ranging from highly diffusive to convection dominated; for instance, viscosity-driven and buoyancy-driven cavity flows.

E. Boundary conditions

The boundary-treatment must be consistent with the numerical scheme; since boundary conditions are very critical/sensitive to accuracy and robustness of any numerical scheme. Two layers of ghost cells are set to

boundaries and boundary conditions are provided in ghost cells. However, actual boundary conditions are recovered when a central difference scheme is applied. For a boundary cell face $(i − 1/2)$, Dirichlet and Neumann conditions are approximated as follows:

$$\widetilde{W}_{i-1} = 2\widetilde{W}_{i-1/2} - \widetilde{W}_i, \qquad \widetilde{W}_{i-1}$$
$$= \widetilde{W}_i - \left(\Delta n \frac{\partial \widetilde{W}}{\partial n}\right)_{i-\frac{1}{2}} \qquad (47)$$

where $(i − 1)$ represents a ghost cell nodal point and $\widetilde{W}_{i-1/2}$ or $(\partial \widetilde{W}/\partial n)$ is the specified boundary value. The wall viscous flux is determined using a second-order one-sided formula and the wall pressure is evaluated using a second-order extrapolation from the computational domain. In the initial iteration step, $\Delta \widetilde{W}_{i-1} = 0$. Boundary conditions of the pressure-correction are treated implicitly as follows. In the ghost cell, Equation (28) becomes:

$$B_{i-1}p'_{i-1}$$
$$= B_i p'_i \qquad (48)$$

where Dirichlet and Neumann boundary conditions assume that $B_i = 0$ and $B_{i-1} = B_i$, respectively. This strategy accelerates the convergence speed of algorithm.

F. Steps of algorithm

Equations are solved with a tridiagonal matrix algorithm (TDMA); different steps of the solver can be summarized as follows:

➤ Guess pressure, velocity, and scalar fields.
➤ Solve momentum equations and calculate tentative velocity fields u^* and v^* using Equation (19).
➤ Calculate $\widetilde{U}^*_{i\pm 1/2}$, using relations like Equation (43) to form the mass imbalance $\dot{M}^*_i$ and solve Equation (28) for p'.
➤ Construct conservative correction terms using Equation (45).
➤ Update velocity, pressure and scalar fields.
➤ Repeat steps 2-5 until convergence is attained.

IV. NUMERICAL EXPERIMENTS

In order to make a realistic assessment between SIMPLE and ASIMPLE algorithms, numerical experiments are carried out for buoyancy-driven flows in a square cavity and a half-concentric annulus, respectively. Both cases are probably the most conventional computational experiments, retaining strong source terms to validate the predictive performance of a numerical formulation. Since there are no analytic solutions available to these problems, literature data and results of the control-volume-based finite-element method (CVFEM) with vorticity stream function formulation are considered as standards for estimating the accuracy of computations obtained by the proposed methods.

The computational system has a characteristic length of L with insulated top and bottom walls. The hot and cold vertical walls are both isothermal, having temperatures T_h and T_c, respectively. The fluid is incompressible with constant properties except for the density variation in the buoyant force; the usual Boussinesq approximation is introduced. The overall system of equations, comprising continuity, X-momentum, Y-momentum and energy can be written in Cartesian coordinates for a two-dimensional case as follows:

$$\frac{\partial U}{\partial \tau} + \frac{\partial V}{\partial Y}$$
$$= 0 \qquad (49)$$

$$\frac{\partial U}{\partial \tau} + \frac{\partial}{\partial X}\left(UU - \frac{\partial U}{\partial X}\right) + \frac{\partial}{\partial Y}\left(VU - \frac{\partial U}{\partial Y}\right)$$
$$= -\frac{\partial P}{\partial X} \qquad (50)$$

$$\frac{\partial V}{\partial \tau} + \frac{\partial}{\partial X}\left(UV - \frac{\partial V}{\partial X}\right) + \frac{\partial}{\partial Y}\left(VV - \frac{\partial V}{\partial Y}\right)$$
$$= Gr\theta - \frac{\partial P}{\partial Y} \qquad (51)$$

$$\frac{\partial \theta}{\partial \tau} + \frac{\partial}{\partial X}\left(U\theta - \frac{1}{Pr}\frac{\partial \theta}{\partial X}\right)$$
$$+ \frac{\partial}{\partial Y}\left(V\theta - \frac{1}{Pr}\frac{\partial \theta}{\partial Y}\right) = 0 \qquad (52)$$

In the above-mentioned Equations (49-52), all variables are non-dimensionalized using dimensionless parameters, defined as:

$$X = \frac{x}{L}; \qquad Y = \frac{y}{L}; \qquad U = \frac{uL}{\nu}$$
$$V = \frac{\mu L}{\nu}; \qquad P$$
$$= \frac{pL^2}{\rho\nu^2}; \qquad \tau = \frac{\nu t}{L^2} \qquad (53)$$
$$\theta = \frac{T - T_c}{T_h - T_c}; \qquad Gr = \frac{g\beta(T_h - T_c)L^3}{\nu^2}; \qquad Pr = \frac{\nu}{\alpha}$$

The selected problems are governed by two dimensionless parameters: Grashof number Gr and Prandtl number Pr. The parameters g, ν, α and γ are the gravitational acceleration, kinematic viscosity, thermal diffusivity and coefficient of thermal expansion, respectively. In the current computations, Pr is set to unity. The dimensional reference velocity can be evaluated as $\Psi = \sqrt{g\gamma(T_h - T_c)L} = \nu\sqrt{Gr}/L$. Therefore, the dimensionless reference velocity defined herein is $U_{ref} = \Psi L/\nu = \sqrt{Gr}$. In principle, the buoyancy-driven flow in a square cavity/half-concentric annulus represents a good test case for the general applicability of a solver, retaining both highly diffusive central and convective near-wall flows.

It is worth noting that the SIMPLE method evaluates the time step Δt from Equation (31) with an exclusion of the artificial sound speed C and calculates the weighting factor $\Delta t'$ from Equation (32) with $CFL = 1$. An important parameter to judge the convergence of SIMPLE algorithm is the average mass residual in the computational domain, which can be expressed as:

$$\Delta \rho^*$$
$$= \sqrt{\frac{\sum_{i=1}^{NP} |\dot{M}^*_i|^2}{NP}} \le \zeta_t \qquad (54)$$

where NP is the number of computational cell, and ζ_t is the user-defined tolerance limit. For the ASIMPLE algorithm, the cell mass imbalance $\dot{M}_i^*$ in Equation (31) is replaced by the artificial mass imbalance p_i'/C_i^2. For each computational grid, the maximum possible CFL number is approximated, enhancing the fastest convergence. Each computational test case uses a variable grid spacing to resolve the sharp gradients in near-wall regions. Grid densities (not shown) are varied to guarantee the grid independence of numerical results; numerical errors due to the grid size is estimated as less than 2%, based on grid dependence tests for both velocity and temperature fields.

Since the buoyancy source term $f^V = Gr\,T$ in Equation (51) is positive; its linearization is not possible. However, the stiffness caused by f^V can be reduced by applying the pseudo-linearization:

$$\frac{\partial f^V}{\partial V} = -\frac{f^V}{|\Delta V_{max}|} \tag{55}$$

Thus the maximum change in V induced by f^V is limited to $|\Delta V_{max}|$. The maximum change relating to $|\Delta V_{max}|$ can be estimated by utilizing the current value of V as:

$$|\Delta V_{max}| = \frac{V}{C_T} \tag{56}$$

where the value of C_T is set to $\max(1; CFL)$ after test calculations. The local Nusselt number can be evaluated as follows:

$$Nu = -\frac{\partial T}{\partial X}|_{X=0} \tag{57}$$

where $\partial T/\partial X$ is approximated using a three-point one-sided difference. To this end, it must be stressed that both SIMPLE and ASIMPLE algorithms uproot the well-recognized velocity or pressure UR to correct the respective fields.

A. Buoyancy-driven flow in square cavity

Figure 1 shows a schematic diagram of the natural convection in a square cavity. The characteristic length is L with adiabatic top and bottom walls; left and right vertical walls are hot and cold, respectively, having constant temperatures T_h and T_c. The overall system of equations in non-dimensional form for the two-dimensional case is given earlier in Equations (49-52). Boundary conditions are as follows: U = V = 0 on all walls, θ = 1.0 on the left wall, θ = 0 on the right wall and ∂θ/∂Y = 0 on top and bottom walls.
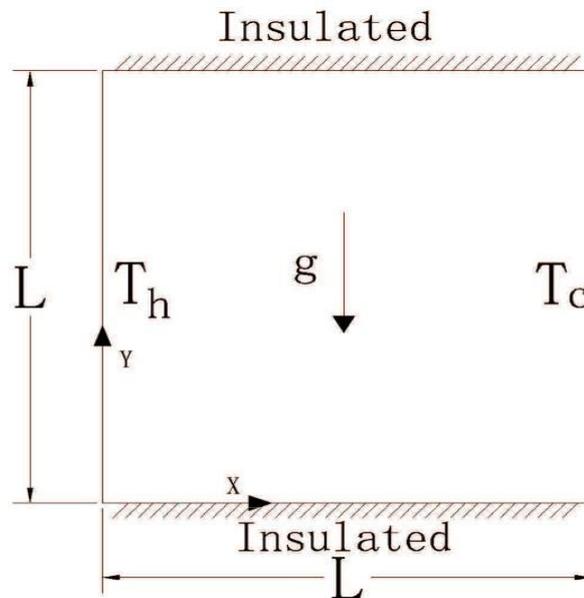

Fig 1:- Physical model of buoyancy-driven cavity flow.

32×32 non-uniform grid for Gr = 10^5, 40×40 for Gr = 10^6 and 60×60 for Gr = 10^7, retaining finer grid points near the walls than in the core, are employed for computations. This range of flows is selected with a view to evaluating the performance of proposed algorithms on highly diffusive to convection dominated flow regimes. Plots of vertical velocity and temperature profiles at the horizontal centerline of cavity are presented at different Gr's in Figures 2 and 3, respectively under same initial conditions. The semi-implicit solutions based on the characteristic-based split algorithm by Massarrotti et al. [30] and CVFEM computational data [31, 32] are also included in the plots. As is observed, comparisons exhibit an encouraging qualitative agreement. Figure 4 depicts local Nusselt number distributions along the hot vertical wall for various Grashof numbers. A careful inspection shows that the SIMPLE method is in excellent agreement with the ASIMPLE algorithm.
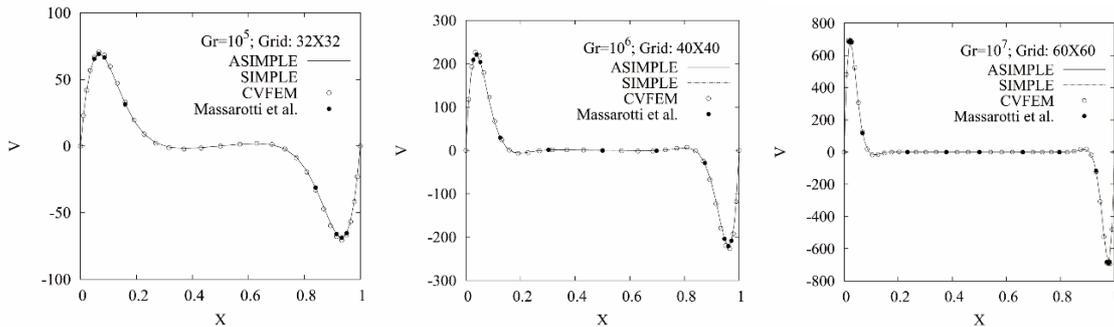
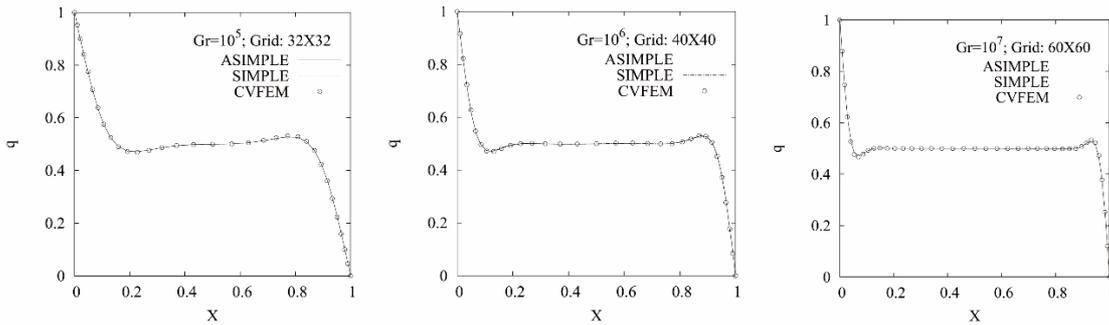Fig 2:- Velocity profiles on horizontal midplane for buoyancy-driven cavity flow.

Fig 3:- Temperature profiles ($\theta = q$) on horizontal midplane for buoyancy-driven cavity flow.
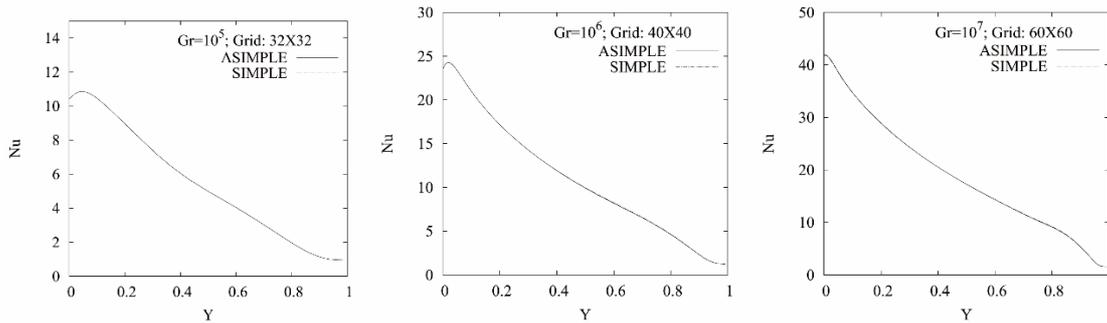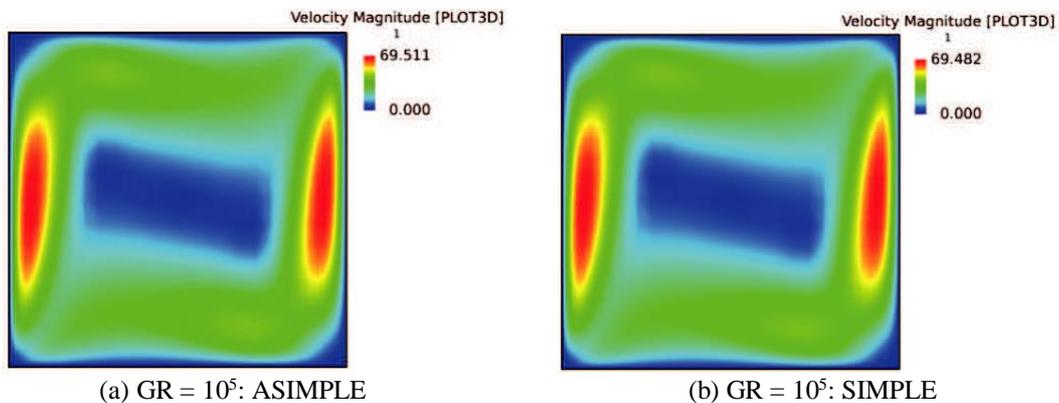

Fig 4:- Local Nusselt number along hot wall for buoyancy-driven cavity flow.

Shown in Figure 5 are the velocity contours for two algorithms at different Grashof numbers. Consistent with velocity and temperature profiles as demonstrated in Figures 2 and 3, boundary-layer edges move toward the wall and the area of central flow-cavity increases with increasing Gr. Interestingly, the influence of secondary flows is very clear when investigating the reasonable distortion near upper left and lower right corners of velocity plots at a higher Gr. Evidently, for these particular grid refinements, solutions from both algorithms are favorably comparable to each other; differences are almost indistinguishable.

(a) GR = 10^5: ASIMPLE   (b) GR = 10^5: SIMPLE

(c) GR = 10^6: ASIMPLE

(d) GR = 10^6: SIMPLE

(e) GR = 10^7: ASIMPLE
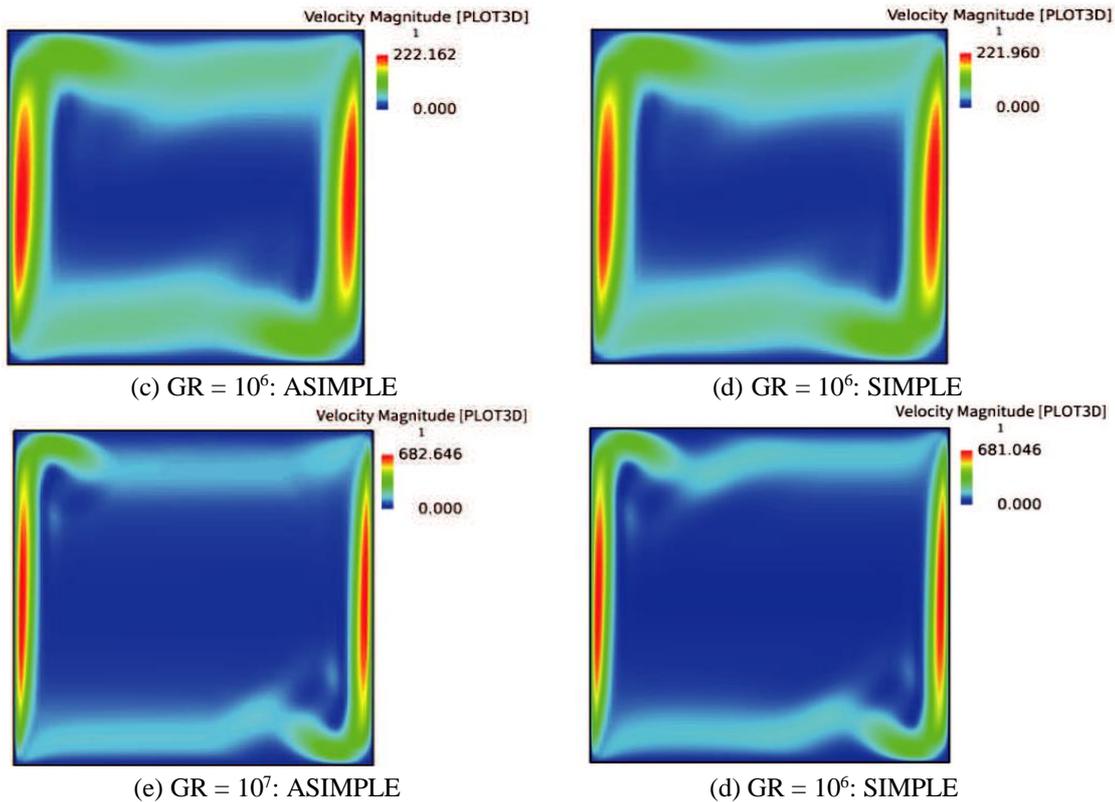
(d) GR = 10^6: SIMPLE

Fig 5:- Velocity contours for buoyancy-driven cavity flow.

Figure 6 displays the convergence histories of fictitious/artificial mass residuals at different Grashof numbers with the same initial conditions. Observed fluctuations in the mass imbalance profiles could be presumably due to the pseudo-linearization of positive strong buoyant source with the CFL numbers. However, mass residuals of both schemes drop to a lower state for a given number of iteration cycles. Although, the mass imbalance has different interpretations in SIMPLE and ASIMPLE algorithms, it is obvious that the ASIMPLE method can consume a higher CFL number than that of the SIMPLE scheme to achieve faster convergence. Convergence plots unambiguously confirm the conclusion that both methods work effectively toward avoiding the unwanted oscillations even at a higher Gr without employing velocity and pressure UR factors.
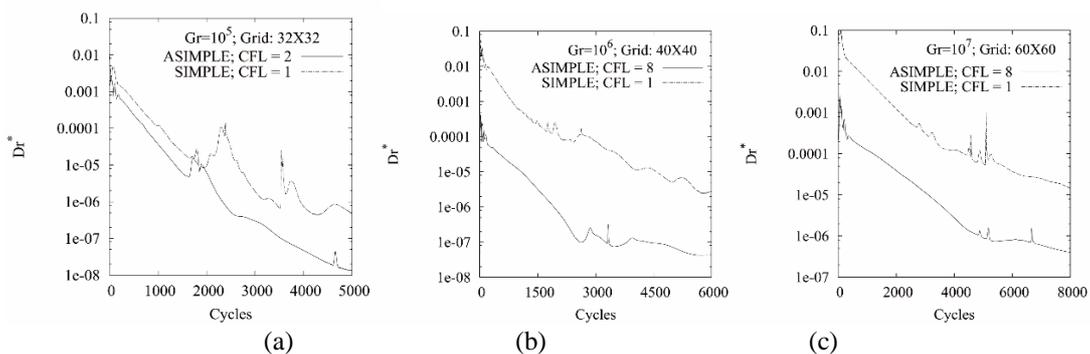

(a)   (b)   (c)

Fig 6:- Convergence histories of mass residuals $(\Delta\rho^* = Dr^*)$ for buoyancy-driven cavity flow.

B. Buoyancy-driven flow in an annulus

A schematic of the half-concentric annulus is shown in Figure 7. The radii of inner and outer surfaces are designated by R_i and R_o, respectively, with $Ro/Ri = 3$. The origin of Cartesian coordinate is situated at the center of circles. The geometry has a characteristic length $L = Ro − Ri$ with insulated top and bottom vertical walls. The inner and outer curved surface temperatures are held at T_h and $T_{c, respectively}$ with $T_h > T_c$. Governing equations and boundary conditions are equivalent to those applied for the buoyancy-driven square-cavity flow.
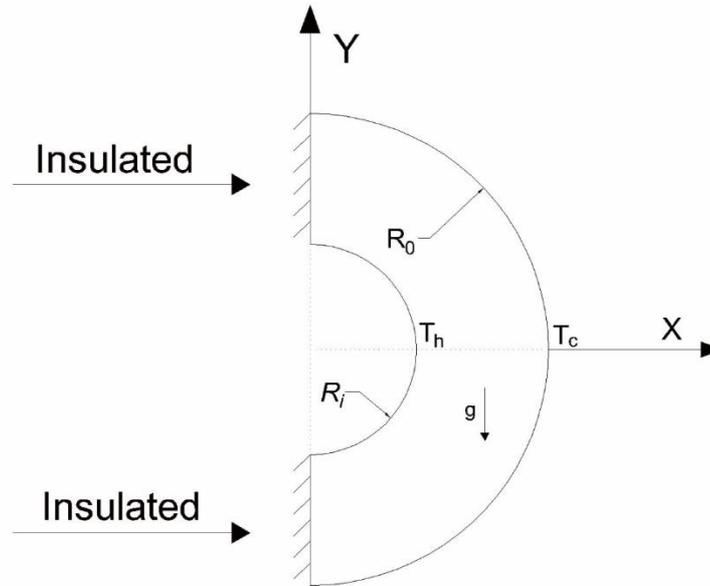
Fig 7:- Physical model of buoyancy-driven flow in annulus.

Tamamidis et a. [23] have made an assessment between the SIMPLE approach with the Rhie-Chow (RC) cell-face interpolation scheme [34] and one variant of the artificial compressibility (AC) method based on steady incompressible viscous-flow computations for 3D curvilinear grids. Overall, predictions from the SIMPLE method compare more favorably with experiments than those of well-resolved computations provided by the AC method. Since the current SIMPLE is analogous to the RC scheme used in Reference [23], its computations can be used as a reference for comparisons, although the present test case is not a well-documented benchmark.

The employed non-uniform grid refinement, consisting of 40 radial and 30 circumferential line segments, is assumed to be sufficiently accurate to compute associated transport characteristics. Figures 8 and 9 illustrate the vertical velocity and temperature profiles, respectively, at the horizontal midplane for $Gr = 10^5 - 10^6$, where X is exactly measured from the inner curved surface. Remarkably, the correspondence of ASIMPLE with the SIMPLE is quite striking and both solutions produce an impressive agreement with CVFEM data. Nevertheless, some discrepancies between SIMPLE/ASIMPLE and CVFEM results appear after the boundary layers. The probable reason for this inconsistency is that the CVFEM estimates the nodal velocity and temperature by computing the weighted nodal velocity and temperature. This incongruity may be reduced to some extent using a finer grid resolution. Ostensibly, the local Nusselt number distributions in Figure 10 replicate identical solutions.
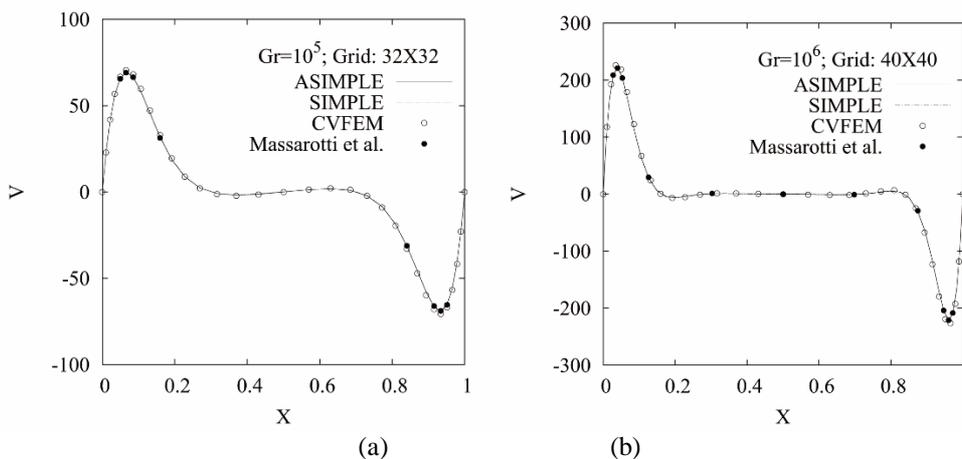

(a)   (b)

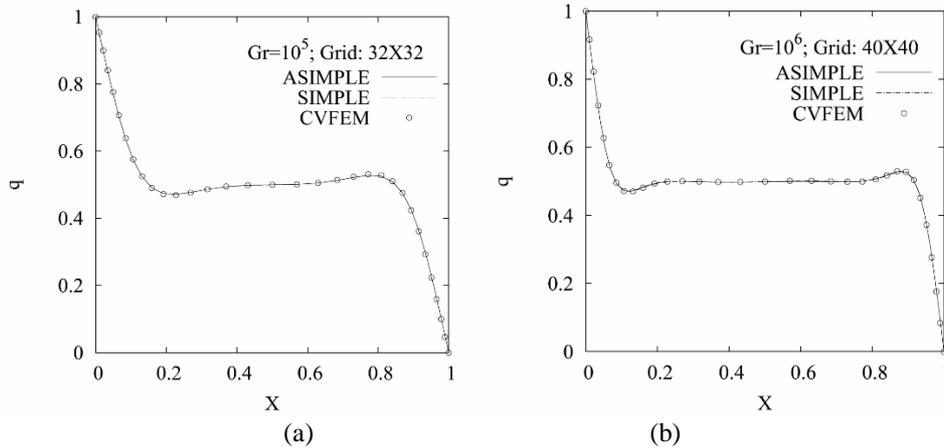Fig 8:- Velocity profiles on horizontal midplane for buoyancy-driven flow in annulus.

Fig 9:- Temperature profiles ($\theta = q$) on horizontal midplane for buoyancy-driven flow in annulus.
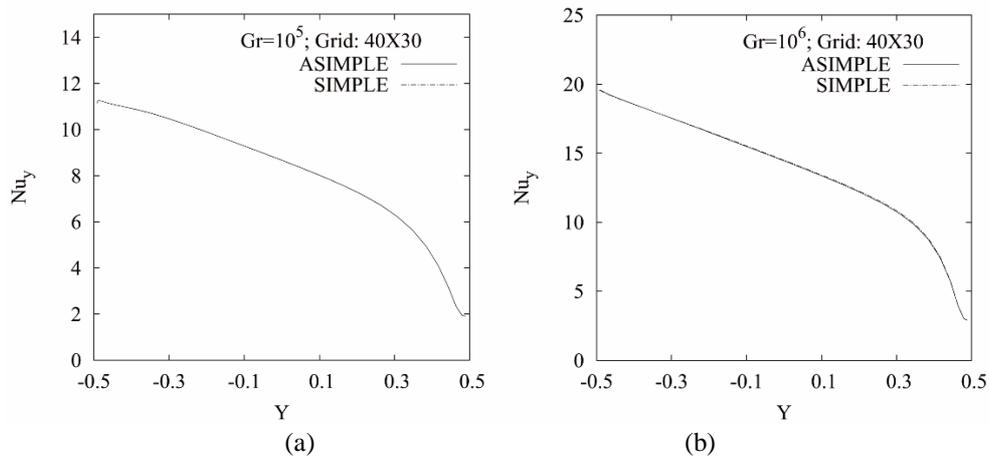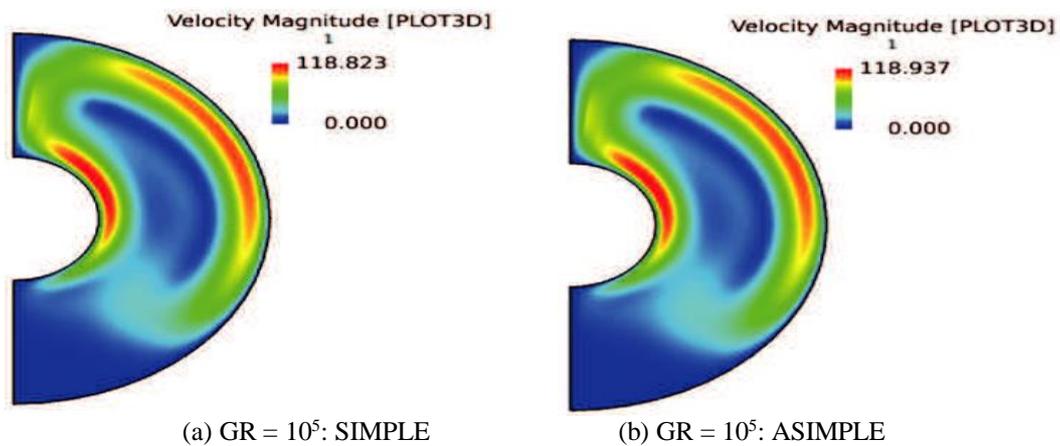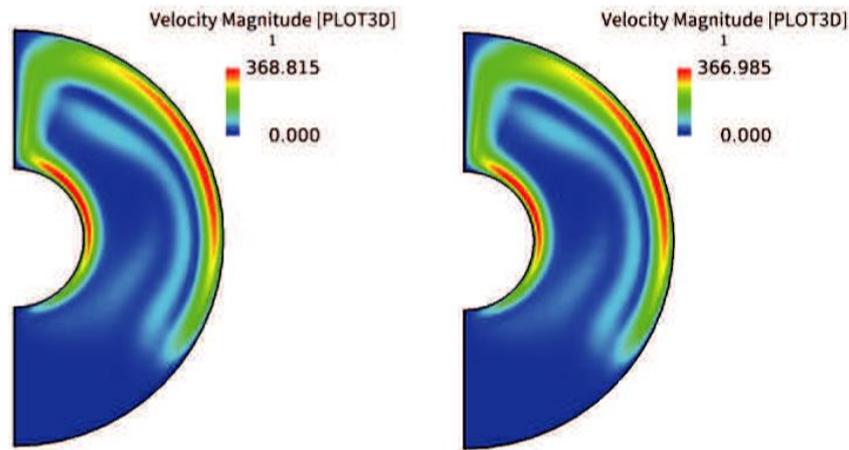

Fig 10:- local Nusselt number along hot wall for buoyancy-driven flow in annulus.

Figure 11 demonstrates the velocity contours, pertaining to SIMPLE and ASIMPLE algo-rithms at two different Grashof numbers. Explanations analogous to Figure 5 for the buoyancy-driven natural convection in a square cavity could be valid for these velocity contours and they are compatible with velocity and temperature profiles shown in Figures 8 and 9, respectively. The glaring difference of Figure 11 with those of Figure 5 is that with increasing Gr, central big vortexes are stretched along the curvature of annular cavity, augmenting the velocity magnitude in near-wall regions and temperature profiles get steeper therein.

(a) GR = 10^5: SIMPLE   (b) GR = 10^5: ASIMPLE

(c) GR = 10^6: SIMPLE   (d) GR = 10^6: ASIMPLE

Fig 11:- Velocity contours for buoyancy-driven flow in annulus.

Figure 12 exhibits the convergence histories for the mass imbalance/artificial density change at different Grashof numbers. Similar to the previous case, the convergence becomes faster in the ASIMPLE method with a higher CFL number. It is once more ensured that higher CFL numbers depend on the artificial sound speed C. CFL = 1 is a good compromise with the SIMPLE algorithm to accommodate better convergence in all computations.
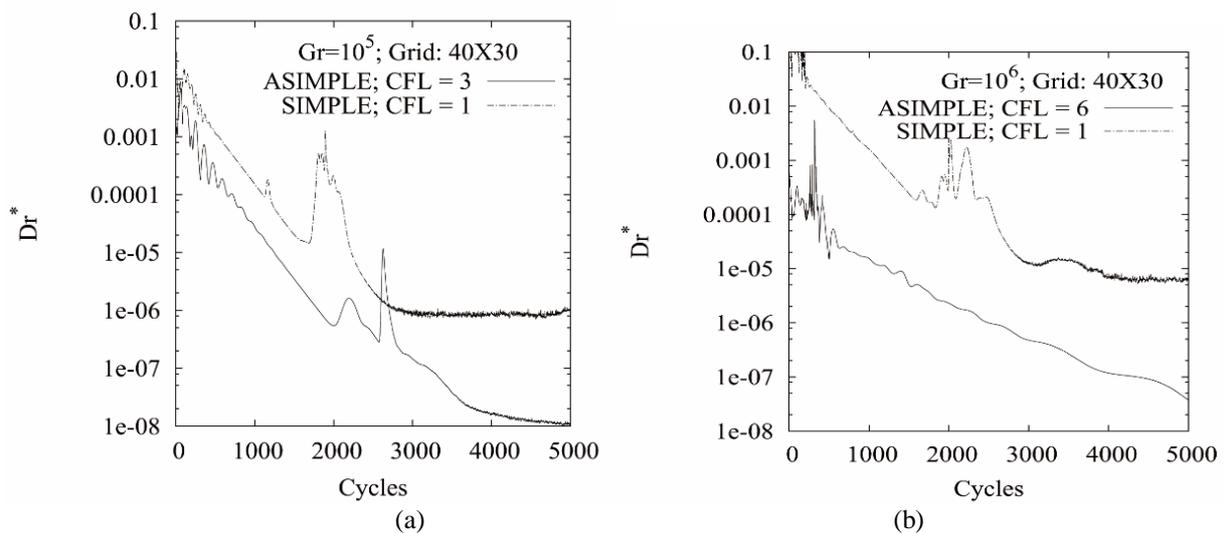

(a)   (b)

Fig 12:- Convergence histories of mass residuals ($\Delta\rho^* = Dr^*$) for buoyancy-driven flow in annulus.

Average Nusselt number	ASIMPLE		SIMPLE	
Gr	Square	Cylinder	Square	Cylinder
10^5	5.42	8.42	5.42	8.41
10^6	10.94	14.36	10.95	14.38
10^7	18.90		18.91	

Table 1:- Average Nusselt number at different Grashof numbers for buoyancy-driven flows.

Apparently, quantitative results presented in Table 1 show that ASIMPLE solutions match SIMPLE data very well. The table represents the values of average Nusselt number on the hot wall for both square and concentric annular cavities. The larger the average Nusselt number, the more obvious is the convection heat transfer.

V. CONCLUSIONS

A comparative assessment is prosecuted between SIMPLE and ASIMPLE algorithms within the framework of a pressure-based approach, using two-dimensional buoyancy-driven incompressible flows on a curvilinear non-staggered grid. Both schemes use a cell-centered finite-volume approximation, solving the flow equations in a segregated fashion with a pseudo-time integration attribute. The ASIMPLE includes an AC parameter with the pressure Poisson equation, amplifying the diagonal dominance of influence coefficients. The AC artifacts invoke artificial density perturbations, contributing assertively an intended linearization to tentative momentum residuals; the AC-generated sound speed provides just enough damping at the cell-face velocity via the time step, preventing the pressure-velocity decoupling. Relevant phenomena associated with both algorithms alleviate the need for UR factor even in the presence of a dominant source term in momentum equations. However, compared with the SIMPLE model, the ASIMPLE scheme has been additionally sensitized to enhanced convergence and robustness by including the time step Δt, parameterized with diffusive, convection and artificial sound speeds. In principle, the actual coding pertaining to SIMPLE and ASIMPLE methods is quite simple and straightforward.

The SIMPLE algorithm and its variant ASIMPLE perform well quite consistently over the different flows computed in this research. Results demonstrate that both methods compare favorably with literature and CVFEM data for a specified level of grid resolution. Compared with the SIMPLE formula, flow quantities namely the velocity, temperature and heat transfer-rate remain almost unaffected by the addition of AC term with the ASIMLE; however, convergence studies essentially reveal that the ASIMPLE scheme is capable of producing satisfactory stabilization for the iteration process. Note that observed oscillations in the convergence plot are due to the linearization of strong positive source term in the momentum equation with CFL numbers. Conclusively, predicted results of ASIMPLE are identical to SIMPLE with substantially augmented stability and convergence properties when convoking the AC parameter, resulting in a conversion of SIMPLE to ASIMPLE. Above all, the linearization of momentum residuals with mass imbalance $\Delta\rho$, and appropriate choices of damping factor η and weighting factor $\Delta t'$ influence the SIMPLE-like algorithm to avoid the presence of UR feature.

REFERENCES

[1]. Turkel E, Radespiel R, Kroll N: Assessment of preconditioning methods for Multi-dimensional Aerodynamics. Computers and Fluids, 26:613-634;1997.

[2]. Weiss JM, Smith WA: Preconditioning applied to variable and constant density Flows. AIAA Journal, 33:2050-2057:1995.

[3]. Rahman MM, Rautaheimo P, Siikonen T: Numerical Study of turbulent heat transfer from a confined impinging jet using a pseudo-compressibility method. 2nd International Symposium on Turbulence, Heat and Mass Transfer, Delft, The Netherlands, pp. 511-520;1997.

[4]. Rahman MM, Siikonen T: An artificial compressibility method for incompressible flows. Numerical Heat Transfer, Part B, 40:391-409;2001.

[5]. Rahman MM, Siikonen T: An artificial compressibility method for viscous incompressible and low Mach number flows. International Journal for Numerical Method in Engineering, 75:1320-1340;2008

[6]. Chorin AJ: A Numerieal method for solving incompressible viscous flow problems. Jour-nal of Computational Physics, 2:12-26;1967.

[7]. Patankar SV, Spalding DB: A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. International Journal of Heat and Mass Transfer, 15:1787-1806;1972.

[8]. Patankar SV: Numerical heat transfer and fluid flow. New York: Mc Graw-Hill, 131-134;1980.

[9]. Patankar SV: A calculation procedure for two-dimensional elliptic situations. Numerical Heat Transfer, 4:409-425;1981.

[10]. [10] van Dooormaal JP, Raithby GD: Enhancement of the SIMPLE method for predicting in-compressible fluid flow. Numerical Heat Transfer7:147-163;1984.

[11]. Spalding DB: A general purpose computer program for multi-dimensional one-and-two phase flow. Mathematics and Computers in Simulations23:267-276;1981.

[12]. Markatos NC, Pericleous KA: Laminar and turbulent natural convection in an enclosed cavity. International Journal of Heat and Mass Transfer27:755-772;1985.

[13]. Issa RI, Gosman AD, Watkins AP: The computation of compressible and incompressible recirculating flows by a non-iterative scheme. Journal of Computational Physics62:66-82;1986.

[14]. Issa RI: Solution of the implicit discretized fluid flow equations by operator splitting. Journal of Computational Physics62(1):40-65;1986.

[15]. ZhiGuo Qu, YaLin He: A new fully implicit algorithm for solving flow and heat transfer problems(A). Journal of Engineering Thermophysics, 26(1):125-127;2005.

[16]. ZhiGuo Qu, YaLin He: A new fully implicit algorithm for solving flow and heat transfer problems(B). Journal of Engineering Thermophysics, 26(1):128-130;2005.

[17]. Cheng YP, Lee TS, Low HT, Tao WQ: Improvement of SIMPLER algorithm for incompressible flow on collocated grid system. Numerical heat transfer B, 51(5):463-486;2007.

[18]. Rahman MM, Siikonen T: An Improved SIMPLE Method on a Collocated Grid, Numeri-cal Heat Transfer B, 38:177-201;2000.

[19]. Rahman MM, Siikonen T, Miettinen A: A Pressure correction method for solving fluid Flow Problems on a Collocated Grid. Numerical Heat Transfer B, 32:63-84;1997.

[20]. Rahman MM, Miettinen A, Siikonen T: Modified SIMPLE formulation on a collocated grid with an assessment of the simplified Quick scheme. Numerical Heat Transfer B, 30:291-314;1996.

[21]. Wen X, Ingham DB: A new method for accelerating the rate of convergence of SIMPLE-like algorithm. International Journal of Numerical Methods in Fluids, 17:385-400;1993.

[22]. Johansson P, Davidson L: Modified collocated SIMPLEC algorithm applied to buoyancy-affected turbulent flow using a multigrid solution procedure. Numerical Heat Transfer B, 28:39-57;1995.

[23]. Armfield SW: Finite difference solutions of the Navier-Stokes equations on staggered and nonstaggered Grids. Computers and Fluids, 20:1-17;1991.

[24]. Rhie CM, Chow WL: Numerical study of the turbulent flow past an airfoil with trailing edge separation. AIAA Journal, 21:15251532;1983.

[25]. Peric M: Analysis of pressure velocity coupling on nonorthogonal grids. Numerical Heat Transfer B, 17:63-82;1990.

[26]. Castrejon A, Spalding DB: Experimental and theoretical study of transient free-convection flow between horizontal convection cylinders. International Journal of Heat and Mass Transfer, 31:273-284:1998.

[27]. Fuchs L, Tillmark N: Numerical and experimental study of driven flow in a polar cavity. International Journal for Numerical Methods in Fluids, 5:311-329;1985

[28]. Kuehn TH, Goldstein RJ: Experimental and theoretical study of natural convection in the annulus between horizontal convection cylinders. Journal of Fluid Mechanics, 74:695-719:1976.

[29]. Wrobel WA, Fornalik-Wajs E, Szmyd JS: Analysis of the influence of a strong magnetic field gradient on convection process of paramagnetic fluid in the annulus between horizontal concentric cylinders. Journal of Physics Conference Series, 395:1;2012.

[30]. W. Shyy W: A study of finite difference approximations to steady-state convection dominated flow problems. Journal of Computational Physics, 15:415-438;1985.

[31]. Massarrotti N, Arpino F, Lewis RW, Nithiarasu P: Explicit and semi-implicit CBS proce-dures for incompressible viscous flows. International Journal for Numerical Methods in Engineering, 66:16181640;2006

[32]. [32] Rahman MM, Lampinen MJ: Numerical study of natural convection from a vertical sur-face due to combined buoyancies. Numerical Heat Transfer A, 28:409-429:1995.

[33]. Kettleborough CF, Husain SR, Prakash C: Solution of fluid flow problems with the vorticity-stream function formulation and the control volume based finite element method. Numerical Heat Transfer B, 16:31-58;1989.

[34]. Tamamidis P, Zhang G, Assanis DN: Comparison of pressure-based and artificial compressibility methods for solving 3D steady incompressible viscous flows. Journal of Computational Physics, 124:113:1996.