JOURNAL OF COMPUTATIONAL PHYSICS **124,** 1–13 (1996)
ARTICLE NO. 0041

Comparison of Pressure-Based and Artificial Compressibility Methods for Solving 3D Steady Incompressible Viscous Flows

PANOS TAMAMIDIS,* GUOQING ZHANG,† AND DENNIS N. ASSANIS†

Department of Mechanical Engineering, Aristotle University of Thessaloniki, Greece 54006; and †Department of Mechanical Engineering and Applied Mechanics, University of Michigan, Ann Arbor, Michigan 48109

Received February 27, 1995; revised July 3, 1995

Pressure-based and artificial compressibility methods for calculating three-dimensional, steady, incompressible viscous flows are compared in this work. Each method is applied to the prediction of three-dimensional, laminar flows in strongly curved ducts of square and circular cross sections. Numerical predictions from each method are compared with available experimental data and previously reported predictions using a multigrid numerical method. The accuracy, grid independence, convergence behavior, and computational efficiency of each method are critically examined. Hence, some of the merits and demerits of each method are revealed in this study. © 1996 Academic Press, Inc.

1. INTRODUCTION

Three-dimensional, steady, incompressible flows are commonly found in a wide range of industrial devices. One of the many challenges in computational fluids dynamics (CFD) of incompressible flows comes from the weak coupling of the velocity and pressure fields. This coupling has to be accomplished in such a way as to ensure that the divergence of the velocity vanishes. Among commonly used methods for handling the velocity–pressure coupling in incompressible flows are pressure-based methods, and methods based on the concept of artificial compressibility.

The pressure-based method (PBM), which is perhaps the most widely used for incompressible flows, was introduced by Harlow and Welch [1] for the calculation of unsteady flows. The extension of this method to steady flows has been detailed by Patankar and Spalding [2] and Patankar [3]. The basic idea is to formulate a Poisson equation for pressure corrections, and then to update the pressure and velocity fields until a divergence-free velocity field is obtained. Two different grid arrangements have been used in implementations of the PBM method: (i) staggered grids [4] with different control volumes for velocities and pressure and (ii) collocated grids [5] with the same control volume for all variables. However, the use of staggered grids introduces significant complexities in code development, increases the number of storage allocations, and requires intense interpolations. Thus, ever since the collocated grid arrangement was proposed [5], staggered grids have seldomly been used, while collocated grids are being increasingly applied to recent studies [6–15]. Nevertheless, there are certainly some critical issues that require attention when using the collocated grid arrangement, for instance, one might need artificial damping terms, or a special cell face interpolation technique to avoid the checkerboard problem [5].

The artificial compressibility method (ACM) for calculating steady flows was proposed by Chorin [16]. In this method, an artificial compressibility term is introduced in the continuity equation, and the unsteady terms in the momentum equations are retained. Hence, the system of equations becomes hyperbolic and many of the methods developed for hyperbolic systems can be applied. Several variations of the ACM method have been reported in the literature. Steger and Kutler [17], Kwak *et al.* [18], and Rogers *et al.* [19] used an implicit approximate factorization scheme by Beam and Warming [20]. Rogers *et al.* [21] used higher-order flux splitting techniques. Rogers and Kwak [22] used a flux-difference split approach. Ramshaw and Mousseau [23] accelerated convergence of the ACM method by introducing an artificial bulk viscosity to dissipate the artificial sound waves more rapidly. Further, Turkel [24] introduced artificial time derivatives in the momentum equations to allow for faster convergence. Recently, Rosenfeld and Kwak [25] extended the ACM method to solve unsteady problems.

At this point, it must be emphasized that the PBM and ACM methods are two different approaches to solve the system of nonlinear algebraic equations that result when the steady state governing equations are discretized. However, each method can be used with either a finite difference (FD) or a finite volume (FV) formulation for the spatial discretization. Ideally, a comparison of the two methods should be carried out for the same discretization, preferably the FV formulation which is inherently more conservative [26]. However, historically, the ACM method

0021-9991/96 $18.00

has been implemented with FD discretizations so as to simplify code development. Hence, to our knowledge, the only published comparative evaluation of the ACM and PBM methods has been carried out in the context of a FD formulation, and has addressed only two-dimensional flows [27]. The objective of this work is to carry out a critical comparison of accuracy, grid independence, convergence behavior, and computational efficiency of two representative pressure-based and artificial compressibility methods for calculating three-dimensional, steady, incompressible, laminar flows.

The PBM approach is implemented based on the work of Tamamidis and Assanis [13], using a finite volume (FV) formulation on a collocated grid arrangement. The ACM approach is based on the widely used work of Rogers *et al.* [28] and is implemented with a finite-difference (FD) formulation. This particular ACM method has been selected since it uses a single artificial parameter, rather than two parameters which are used in some of the competing ACM methods [24]. Both the PBM and ACM methods use generalized body-fitted coordinates (BFC) and the third-order QUICK discretization scheme [29]. While the PBM is implemented with the more conservative FV formulation and the ACM is implemented with a FD formulation, a comparison of the alternative approaches is still useful to the CFD community, as it can establish some of the performance and accuracy characteristics of each method in three-dimensional problems. In order to quantify the error introduced by implementing the ACM method with a FD rather than a FV discretization, a comparison of the truncation errors produced by each technique will also be conducted and its effect on overall accuracy will be assessed. The benchmark evaluation of the two methods will be performed through calculations of laminar flows in 90° curved ducts of square and circular cross sections. Solutions obtained using the two methods will be compared with available experimental data and the well-resolved numerical predictions of Smith and Vanka who used the multigrid technique [15].

2. MATHEMATICAL FORMULATION

2.1. *Pressure-Based Method*

The transformed governing equations for steady, viscous, incompressible flows can be written in the following format [30]:

$$\frac{\partial \mathbf{E}^{\xi}}{\partial \xi}+\frac{\partial \mathbf{E}^{\eta}}{\partial \eta}+\frac{\partial \mathbf{E}^{\zeta}}{\partial \zeta}=\frac{\partial \mathbf{E}_v^{\xi}}{\partial \xi}+\frac{\partial \mathbf{E}_v^{\eta}}{\partial \eta}+\frac{\partial \mathbf{E}_v^{\zeta}}{\partial \zeta}+\mathbf{P}, \tag{1}$$

where

$$\mathbf{E}^{\xi_i}=\rho\begin{bmatrix} U^{\xi_i} \\ U^{\xi_i}u \\ U^{\xi_i}v \\ U^{\xi_i}w \end{bmatrix},$$

$$\mathbf{E}_v^{\xi_i}=\mu g^{\xi_i\xi_i}\begin{bmatrix} 0 \\ \dfrac{\partial u}{\partial \xi_i} \\ \dfrac{\partial v}{\partial \xi_i} \\ \dfrac{\partial w}{\partial \xi_i} \end{bmatrix}, \quad \mathbf{P}=Ja\begin{bmatrix} 0 \\ -\dfrac{\partial p}{\partial \xi_i}\dfrac{\partial \xi_i}{\partial x} \\ -\dfrac{\partial p}{\partial \xi_i}\dfrac{\partial \xi_i}{\partial y} \\ -\dfrac{\partial p}{\partial \xi_i}\dfrac{\partial \xi_i}{\partial z} \end{bmatrix}+\mathbf{S}_{\mathrm{no}}. \tag{2}$$

In the above equations U^{ξ_i} is the scaled ξ_i-component of the contravariant velocity vector, and $g^{\xi_i\xi_j}$ are metric components introduced from the transformation of the equations from the physical (x, y, z) to the computational (ξ, η, ζ) space:

$$U^{\xi_i}=Ja\left(u\frac{\partial \xi_i}{\partial x}+v\frac{\partial \xi_i}{\partial y}+w\frac{\partial \xi_i}{\partial z}\right). \tag{3}$$

Expressions for Jacobian of the transformation and the various metric terms can be found in [30]. The term $\mathbf{S}_{\mathrm{no}}$ represents the viscous terms due to the nonorthogonality of the grid (i.e., the cross-derivative terms) [30].

The convection-diffusion terms are discretized as

$$\begin{aligned}(\tilde{\mathbf{E}}_i)_{\mathrm{cf}} &= (\mathbf{E}_i)_{\mathrm{cf}}+(\mathbf{E}_{i,v})_{\mathrm{cf}} \\ &= \rho(U^{\xi_i})_{\mathrm{cf}}\begin{bmatrix} 1 \\ u \\ v \\ w \end{bmatrix}_{\mathrm{cf}} - \mu(g^{\xi_i\xi_i})_{\mathrm{cf}}\frac{\partial}{\partial \xi_i}\begin{bmatrix} 0 \\ u \\ v \\ w \end{bmatrix}_{\mathrm{cf}},\end{aligned} \tag{4}$$

where $(\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3) = (\mathbf{E}, \mathbf{F}, \mathbf{G})$, $(\xi_1, \xi_2, \xi_3) = (\xi, \eta, \zeta)$, and cf stands for cell face. The vector of unknowns at a cell face is calculated based on the relation

$$q_{\mathrm{cf}}=q_{u1}+\frac{3q_{d1}-2q_{u1}-q_{u2}}{8}, \tag{5}$$

where $d1$, $d2$, $u1$, $u2$, and $u3$ are the first downstream, the second downstream, the first upstream, the second upstream, and the third upstream nodes, respectively, from the cell face, and the vector of the unknowns is $\mathbf{q} = [u\ v\ w]^{\mathrm{T}}$. As shown in [30], the above interpolation corresponds to the third-order QUICK scheme of Leonard [29].

The terms in parentheses can be viewed as antidiffusion terms and are treated explicitly [30]. The diffusion terms are discretized using second-order derivatives.

Grouping the various terms together we obtain a system of equations of the format [30]

$$\begin{aligned} a_P\mathbf{q}_P = a_{\mathrm{E}}\mathbf{q}_E + a_W\mathbf{q}_W + a_N\mathbf{q}_N \\ + a_S\mathbf{q}_S + a_F\mathbf{q}_F + a_R\mathbf{q}_R + \mathbf{b}, \end{aligned} \tag{6}$$

where **b** is the source term vector which contains the pressure terms, the diffusion terms due to grid nonorthogonality, and the antidiffusive terms. The coupling of the continuity and momentum equations is performed using the interpolation approach of Rhie and Chow [5]; more details can be found in [30].

The pressure field is obtained using the SIMPLEC algorithm [31]. In this algorithm, the final form of the pressure correction equation becomes [30]

$$\begin{aligned} a_P^{p'}p'_P = a_E^{p'}p'_E + a_W^{p'}p'_W + a_N^{p'}p'_N \\ + a_S^{p'}p'_S + a_F^{p'}p'_F + a_R^{p'}p' + b^{p'}, \end{aligned} \tag{7}$$

where the source term $b^{p'}$ is the mass imbalance in the control volume.

Equations (6) and (7) constitute a system of four nonlinear algebraic equations. The system is linearized by resorting to relaxation, where the coefficients in Eqs. (6) and (7) are evaluated using the values from the previous iteration level. First, Eq. (6) is solved for the three velocity components simultaneously. Then, the mass imbalance in every control volume is calculated, followed by the solution of Eq. (7). The pressure corrections are used to correct the pressure and the velocities. This predictor–corrector procedure constitutes one iteration. The solution is declared convergent when the maximum residual becomes smaller than a convergence criterion (10^{-5} or lower). Here, the residual of an equation is defined as

$$\mathbf{R} = \sqrt{\sum_{i=1}^{N} Er_i^2 \Big/ N} \Big/ \text{reference flux}, \tag{8}$$

where N is the number of nodes, Er_i is the error of an equation at a node

$$Er_i = a_i\mathbf{q}_i + \mathbf{b}_i - a_P\mathbf{q}_P, \tag{9}$$

and the reference flux is the flux of the respective variable into the domain. For the residual of the pressure correction equation, however, the error in Eq. (9) is equal to the mass imbalance in the control volume, and the reference flux is the mass flux entering the domain. It should be noted that Eqs. (6) and (7) are solved using the ADI method, where one sweep of the domain corresponds to the three sweeps in the three respective dimensions. Here, one sweep is performed by solving along lines in the two directions on a plane and then moving to the next plane. The solution along one line is achieved using a tridiagonal matrix solver which has been suitably vectorized for the Cray-Y/MP supercomputer.

Beyond tracking variable residuals throughout the convergence process, a mass error is computed to check how accurate, i.e., how close to divergence-free the converged velocity field is. The mass error is defined as the normalized root-mean square of the divergence of the velocity field.

2.2. *Artificial Compressibility Method*

In the artificial compressibility method, the time derivative terms are retained in the formulation. The governing equations in generalized coordinates are [28]

$$\begin{aligned} \frac{\partial\hat{\mathbf{Q}}}{\partial t} + \frac{\partial}{\partial\xi}(\hat{\mathbf{E}} - \hat{\mathbf{E}}_v) + \frac{\partial}{\partial\eta}(\hat{\mathbf{F}} - \hat{\mathbf{F}}_v) + \frac{\partial}{\partial\xi}(\hat{\mathbf{G}} - \hat{\mathbf{G}}_v) \\ = \text{cross terms}, \end{aligned} \tag{10}$$

where

$$\hat{\mathbf{Q}} = Ja\mathbf{Q} = Ja[p\;u\;v\;w]^{\mathrm{T}}, \tag{11}$$

$$\hat{\mathbf{E}} = Ja\begin{bmatrix} \beta U \\ uU + \xi_x p \\ vU + \xi_y p \\ wU + \xi_z p \end{bmatrix}, \quad \hat{\mathbf{F}} = Ja\begin{bmatrix} \beta V \\ uV + \eta_x p \\ vV + \eta_y p \\ wV + \eta_z p \end{bmatrix},$$

$$\hat{\mathbf{G}} = Ja\begin{bmatrix} \beta W \\ uW + \zeta_x p \\ vW + \zeta_y p \\ wW + \zeta_z p \end{bmatrix}, \tag{12}$$

$$\begin{aligned} &U = u\xi_x + v\xi_y + w\xi_z, \quad V = u\eta_x + v\eta_y + w\eta_z, \\ &W = u\zeta_x + v\zeta_y + w\zeta_z, \end{aligned} \tag{13}$$

$$\hat{\mathbf{E}}_v = Re^{-1}g^{\xi\xi}\begin{bmatrix} 0 \\ \dfrac{\partial u}{\partial\xi} \\ \dfrac{\partial v}{\partial\xi} \\ \dfrac{\partial w}{\partial\xi} \end{bmatrix}, \quad \hat{\mathbf{F}}_v = Re^{-1}g^{\eta\eta}\begin{bmatrix} 0 \\ \dfrac{\partial u}{\partial\eta} \\ \dfrac{\partial v}{\partial\eta} \\ \dfrac{\partial w}{\partial\eta} \end{bmatrix},$$

$$\hat{\mathbf{G}}_v = Re^{-1} g^{\zeta\zeta} \begin{bmatrix} 0 \\ \dfrac{\partial u}{\partial \zeta} \\ \dfrac{\partial v}{\partial \zeta} \\ \dfrac{\partial w}{\partial \zeta} \end{bmatrix}. \tag{14}$$

Note that the viscous terms have been simplified by treating the cross-terms explicitly.

In the present implementation of the artificial compressibility method, Eq. (10) is advanced in time using the Beam and Warming [20] approximate factored finite-difference scheme. According to the Euler implicit scheme,

$$\hat{Q}^{n+1} = \hat{Q}^n + \Delta t \left(\frac{\partial \hat{Q}}{\partial t}\right)^{n+1} + O(\Delta t^2). \tag{15}$$

Using Taylor series expansion we obtain

$$\hat{\mathbf{E}}_i^{n+1} = \hat{\mathbf{E}}_i^n + \frac{\partial \hat{\mathbf{E}}_i^n}{\partial \mathbf{Q}} \Delta \mathbf{Q} + O(\Delta t^2), \tag{16}$$

where $(\hat{\mathbf{E}}_1, \hat{\mathbf{E}}_2, \hat{\mathbf{E}}_3) = (\hat{\mathbf{E}}, \hat{\mathbf{F}}, \hat{\mathbf{G}})$, and $\Delta \mathbf{Q} = \mathbf{Q}^{n+1} - \mathbf{Q}^n$. Substituting Eq. (16) into Eq. (15), and the latter into Eq. (10) yields

$$\left\{ I + \Delta t \frac{1}{Ja} \left[\frac{\partial}{\partial \xi_i} (A_i^n - \Gamma_i) \right] \right\} \Delta Q = R, \tag{17}$$

where

$$A_i = \frac{\partial \hat{\mathbf{E}}_i}{\partial Q}, \quad \Gamma_i = \frac{\partial \hat{\mathbf{E}}_{i,v}}{\partial Q} \tag{18}$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{19}$$

$$\mathbf{R} = -\frac{\Delta t}{Ja} \left[\frac{\partial \mathbf{E}_i^n}{\partial \xi_i} + \left(\frac{\partial \Gamma_i^n}{\partial \xi_i} \right) \mathbf{Q}^n \right] + \text{cross terms} \tag{20}$$

with $(\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3) = (\mathbf{A}, \mathbf{B}, \mathbf{C})$, and $(U_1, U_2, U_3) = (U, V, W)$.

Equation (17) is a finite difference scheme with first-order time accuracy. By approximate factorization, the three-dimensional problem is split into three one-dimensional problems. The convective terms are discretized using second-order central differences. The diffusion terms are calculated using second-order derivatives. To make the scheme stable, the fourth-order smoothing terms are added to Eq. (17). The use of central differences for the convective and diffusive terms leads to block-tridiagonal matrices. More details can be found in [18, 28], where a method similar to the one implemented in our work is described in detail. To make a fair comparison with the PBM method, the same third-order QUICK scheme of Leonard [29] was implemented following the deferred correction approach of Khosla and Rubin [32]. This approach gives rise to high order correction partial differences, which are moved into the RHS of the equations and lumped into separate high order source terms; the latter are then treated explicitly during the iteration procedure:

$$\left(\frac{\partial E}{\partial \xi}\right)^{n+1} = \left(\frac{\partial E}{\partial \xi}\right)^{n+1}_{\text{2nd}} + \left[\left(\frac{\partial E}{\partial \xi}\right)^{n}_{\text{QUICK}} - \left(\frac{\partial E}{\partial \xi}\right)^{n}_{\text{2nd}} \right]. \tag{21}$$

Both convergence and accuracy checks are defined for the solution, the $\Delta\mathbf{Q}$ residual, and the mass error. The $\Delta\mathbf{Q}$ residual, referred to as the variable residual, is defined as the root-mean-square value of the change of variable $\mathbf{Q}$ between two consecutive time steps. This value should decrease by at least two to three orders of magnitude for the solution to be declared convergent. The mass error, i.e., the normalized root mean square of the divergence of the velocity field, is defined and used as in the PBM code.

2.3. *Boundary Conditions*

Three types of boundary conditions are used in this work: inlet, exit, and wall boundary conditions. At the inlet, the values of all variables are prescribed. At the outlet, the streamwise gradient for each variable is prescribed to be equal to zero. At the walls, no slip and no penetration are assumed, while the pressure gradient normal to the wall is assumed to be zero.

3. COMPUTATIONAL EXPERIMENTS

3.1. *Flow in a* 90° *Square Duct*

The first test case deals with the prediction of the laminar flow in a 90° curved duct of square cross section. Numerical predictions generated by the present two methods are compared with the measurements of Humphrey *et al.* [34] and the previously reported multigrid predictions of Smith and Vanka [15]. Since the present studies employ high-order discretization schemes, moderate size grids are used, i.e., $52 \times 20 \times 20$ cells with the pressure-based method and $53 \times 21 \times 21$ nodes with the artificial compressibility method. Our grid, shown in Fig. 1a, was generated using Thompson's transfinite interpolation method [35]. The multigrid studies of Smith and Vanka [15] used a very

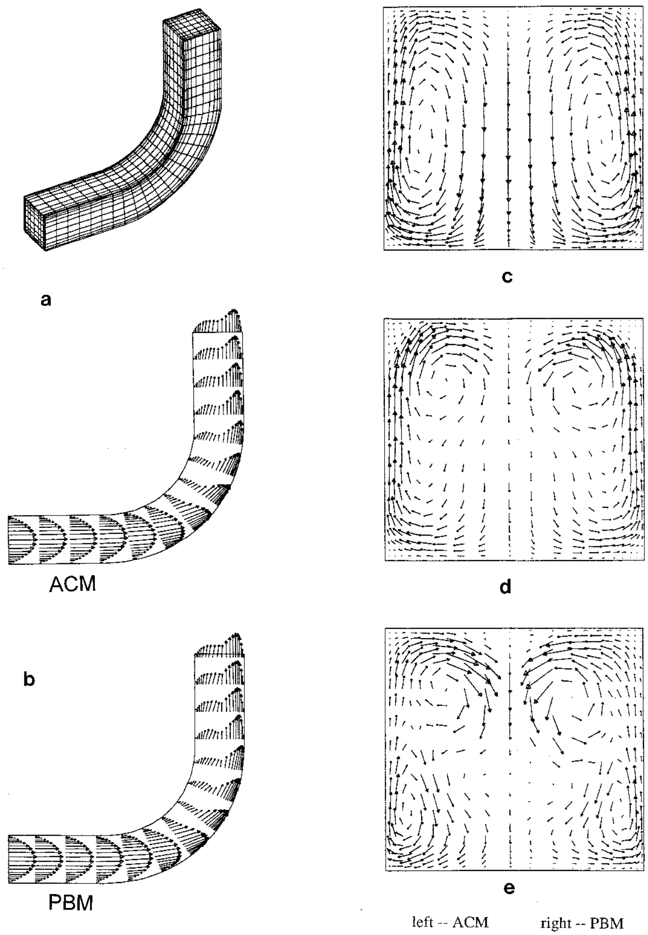

FIG. 1. Laminar flow predictions in a 90° curved duct of square cross-section: (a) perspective view of the grid topology; (b) flow in the vertical mid-plane; (c), (d), and (e) secondary flow at the 30°, 60°, and 90° cross sections.

dense fine grid (120 × 64 × 32) for half of the passage taking advantage of symmetry. A parabolic velocity profile for fully developed flow is prescribed at the inlet, which is located at a distance equal to two hydraulic diameters upstream of the bend. The downstream duct length is equal to two hydraulic diameters. The Re number based on the hydraulic diameter of the duct is 792, which corresponds to the flow of water with velocity equal to 0.0198 m/s. The dimensions of the duct at each cross section are 40 × 40 mm, and the radius of curvature of the suction side is 72 mm.

Figures 1b–e show the flow field at various cross-sectional planes, obtained by employing the PBM and the ACM methods. No significant differences between the results of the two methods can be seen. Figure 1b shows the flow development in the vertical midplane. The "core" fluid, defined as fluid of velocity greater than about 0.9 of the maximum at each streamwise station, is found progressively closer to the outer wall, with a correspondingly low velocity region adjacent to the suction side. This redistribution of streamwise velocity is caused by the secondary flow in the duct, as illustrated in Figs. 1c–e. At 30° (Fig. 1c) a pair of D-shaped counterrotating vortices has already been formed. At the 60° section the centers of the vortices have shifted considerably from their mean radius location at the 30° section towards the inner wall. Their previous D-shape has changed and now resembles a kidney. At the 90° section, two pairs of counterrotating vortices have been formed. Due to this secondary flow, core fluid is convected from the pressure surface along the sidewalls, and low momentum boundary layer fluid from the sidewalls is accumulated at the inner wall. This secondary motion is responsible for the enhanced mixing and heat transfer coefficients, as well as the higher pressure and frictional losses associated with curved ducts.

To explore the ability of the two methods to predict accurate magnitudes of flow velocities, a quantitative comparison of our results with the measurements of Humphrey *et al.* [33] at several midspan locations along the duct is shown in Fig. 2. Although both methods are in good agreement with the experimental results at the 0° and 30° stations, some discrepancies between the PBM and ACM methods start appearing at the 60° section and become more pronounced at the 90° section. The cause of these discrepancies does not appear to be grid dependence, since similar discrepancies are experienced with the well-resolved calculations of Smith and Vanka [15]. It should be noted, however, that PBM numerical predictions are closer to the experimental data at the 90° section than either ACM or multigrid predictions. Overall, the comparison of PBM and ACM predictions with experimental data and multigrid predictions show that both methods under evaluation are capable of producing satisfactory results.

Figure 3 shows the variation of the pressure coefficient C_p inside the bend versus the distance X_h from the duct inlet, measured along the centerline of the duct and nondimensionalized with respect to the hydraulic diameter of the duct. A radial pressure gradient is present even at two hydraulic diameters upstream. Initially a severe, adverse pressure gradient develops on the pressure side of the bend, while a weaker, favorable pressure gradient is evident in the suction side. Their combination is responsible for the initial inward shift of the core flow. Up to 25° into the bend, the agreement between the two methods is excellent. After about 45°, a favorable pressure gradient rapidly develops in the suction side of the bend, and slight differences between the predictions of the methods appear. These slight differences continue in the section downstream of the bend.

The computational performance of each of the two algorithms in the square duct test is investigated further by varying user-specified, method parameters that control

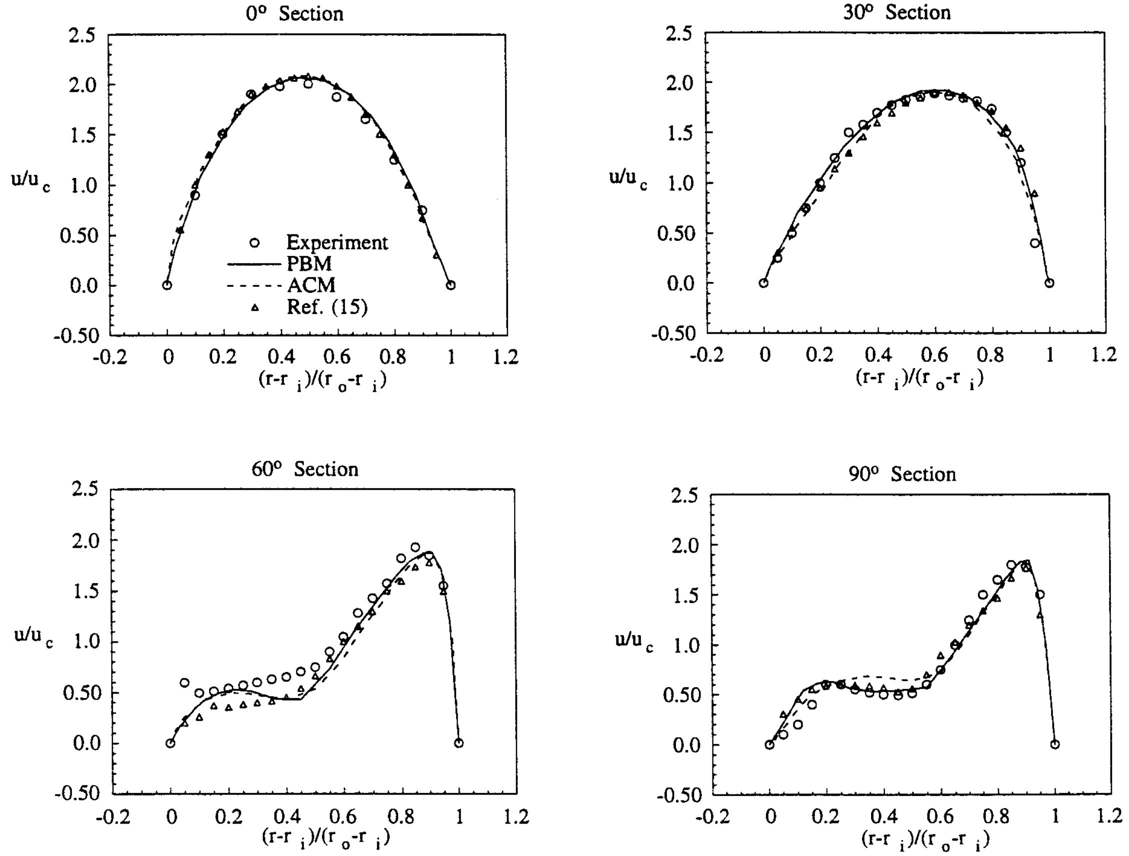

FIG. 2. Comparison of calculations with experimental data for laminar flow in a curved duct of square cross section.

convergence. Figure 4a shows the number of iterations required by SIMPLEC for convergence as a function of the pressure underrelaxation factor ω_p; the velocity underrelaxation factor ω_u was used as a parameter. We should note here that one sweep of the domain for the momentum and pressure equations was performed at each iteration. For most values of ω_u (between 0.3 and 0.7), there exists a range of ω_p values that can be used without considerably affecting the convergence rate. When virtually no underrelaxation is applied to the velocities ($\omega_u = 0.9$), the required number of iterations is considerably reduced, but convergence becomes more sensitive to the values of ω_p used. Based on the evidence shown in this test problem, it appears that a value of $\omega_p = 0.7$ is nearly optimal for all values of ω_u used. This leaves only one parameter (ω_u) to play with in order to minimize the number of iterations required for convergence. In general, while the optimum underrelaxation factor(s) may depend on the specifics of a given geometry and flow field, a safe guess can always be to set both ω_p and ω_u equal to 0.5 and expect reasonable convergence behavior.

On the other hand, the artificial compressibility method uses one user-specified parameter to control convergence, namely the artificial compressibility factor β. The effect of this parameter on the number of iterations is shown in Fig. 4b. Clearly, very small or large values of β slow down solution convergence significantly and may even cause divergence. Using the optimum value of β can markedly reduce the required number of iterations for convergence. In this study, the optimum value of lies between 4 and 6. Note, however, that this optimum value may depend, among other factors, on geometry, grid density, and flow field. A more detailed discussion concerning the selection of the optimum value of this parameter will be presented in Section 4.3.

3.2. *Flow in a* 90° *Circular Duct*

Further evaluation of the two methods in a different geometry is achieved by comparing the predictions of the laminar flow in a 90° curved duct of circular cross section against the laser Doppler measurements of Enayet *et al.* [36] and the computations of Smith and Vanka [15] using the multigrid method. The grid used for our computations consists of 53 × 20 × 20 cells for the PBM method, and 54 × 21 × 21 nodes for the ACM method, as shown in

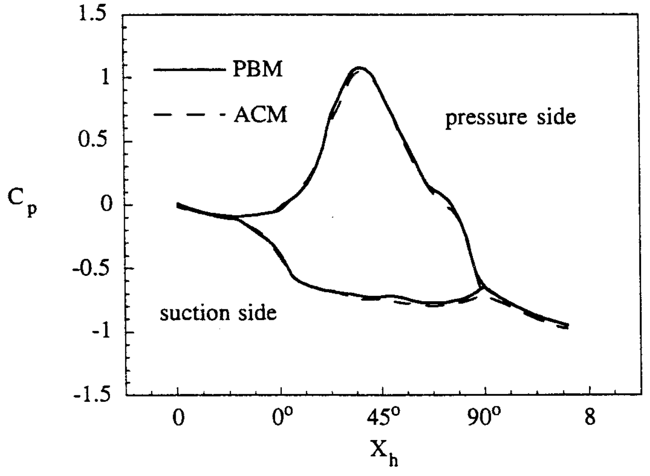

FIG. 3. Wall static pressure coefficient for laminar flow in a curved square duct.

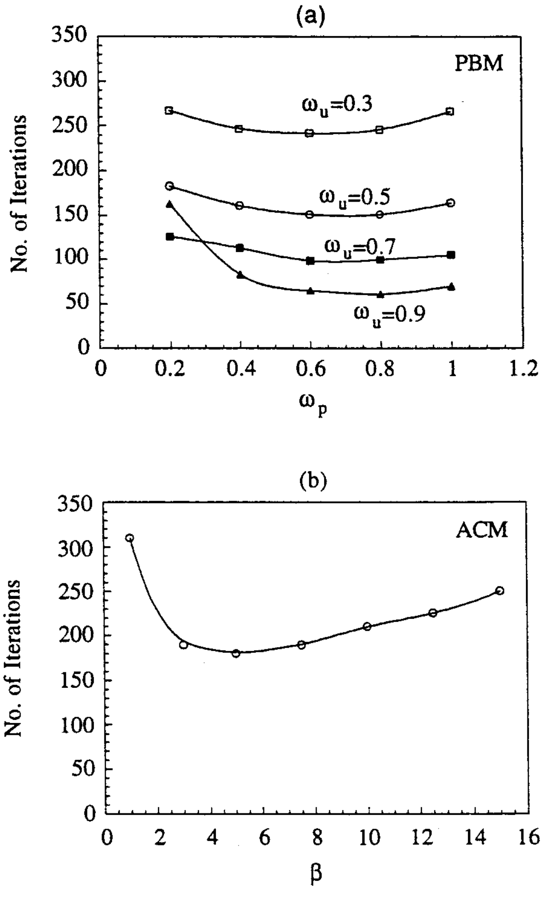

FIG. 4. Number of iterations required for convergence versus user-specified parameters for (a) pressure-based method and (b) artificial compressibility method.

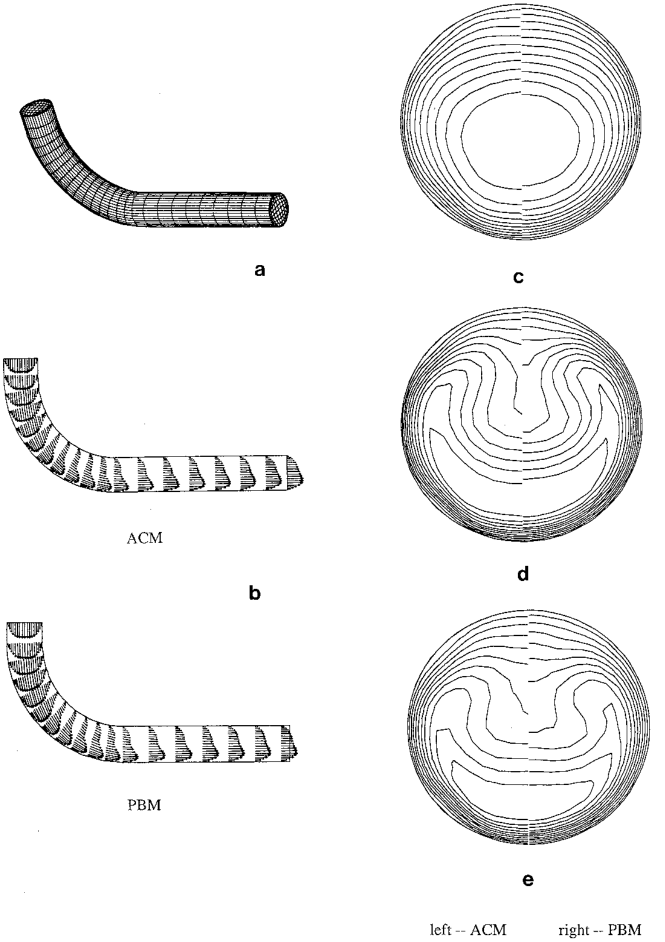

FIG. 5. Laminar flow predictions in a 90° curved duct of circular cross-section: (a) perspective view of the grid topology; (b) flow in the vertical mid-plane; (c), (d), and (e) secondary flow at the 30°, 60°, and 1-d_h cross sections.

Fig. 5a. This grid was generated numerically, following the elliptic grid generation procedure of Thompson *et al.* [34]. Again, our computations are compared with corresponding results reported by Vanka and Smith [15] for their finest grid (120 × 64 × 32 cells for half of the duct). The inlet velocity profile is prescribed by fitting the experimental data at a distance equal to 0.58 hydraulic diameters upstream of the bend. The downstream duct length is equal to five diameters. The Re number based on the diameter of the duct is 500, which corresponds to the flow of water with velocity equal to 0.0105 m/s. The diameter of the duct is 48 mm, and the radius of curvature of the suction side is 110.4 mm.

Figure 5b shows the flow development at the vertical midplane. Similar to the square duct case, the maximum streamwise velocities are found progressively closer to the outer wall. Contours of the streamwise velocity in three cross-stream planes are presented in Figs. 5c–e; the con-

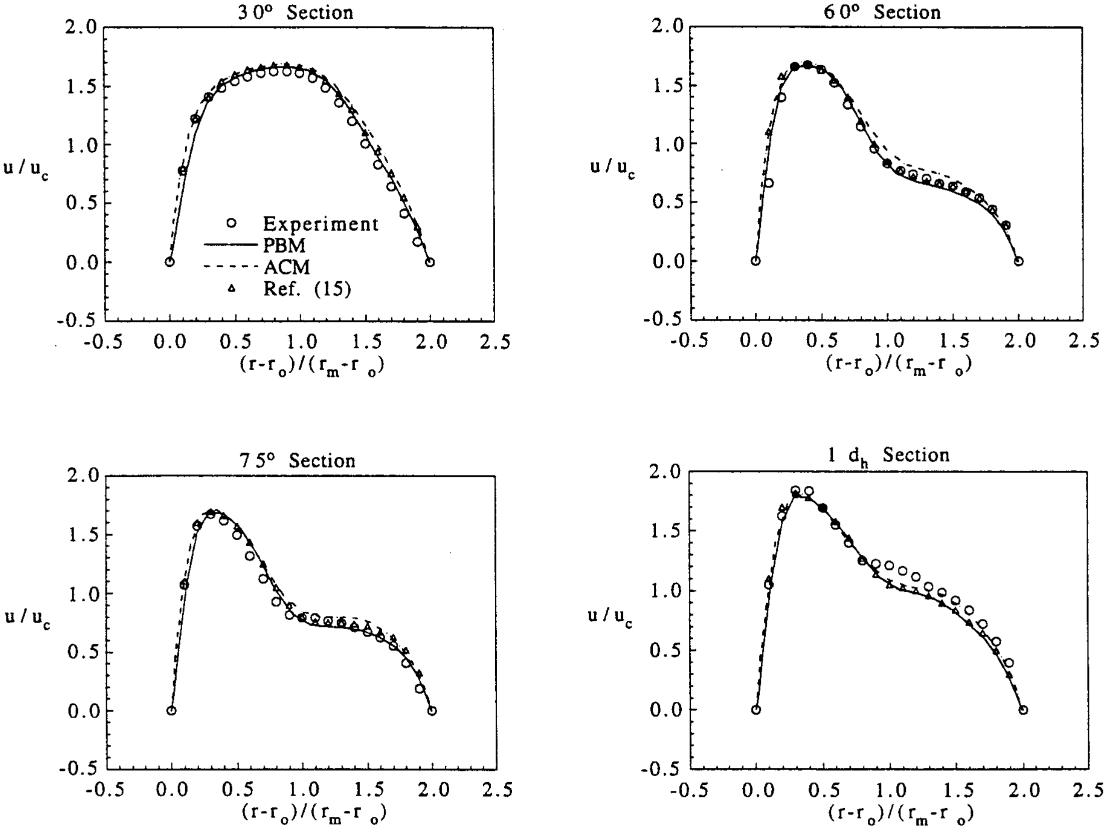

FIG. 6. Comparison of calculations with experimental data for laminar flow in a curved duct of circular cross section.

tours show the effects of strong secondary flow which displaces the region of maximum velocity to the outside of the bend. The secondary flow develops gradually; in the 30° plane the only visible effect is a thickening of the shear layer on the inside of the bend. By 60° the secondary flow has become almost fully developed, as shown by the distortion of the contours. The third set of contours, in the plane located one diameter downstream of the bend, shows that the secondary motion persists without significant decay to this distance.

The distributions of the calculated streamwise velocities are plotted against the measured ones [36] and previous numerical predictions [15] at various midspan locations along the duct (Fig. 6). The velocities have been normalized with respect to the inlet velocity. The results indicate good quantitative agreement of PBM predictions with both measurements and multigrid predictions. On the other hand, ACM predictions are in slightly less satisfactory agreement with measurements and multigrid predictions, especially in the 30° and 60° sections of the duct. The slight discrepancy between PBM predictions and experimental measurements that exists at the downstream section of the duct has also been noticed by Smith and Vanka [15] on their very fine grid and, thus, cannot be attributed to the grid sizes used here.

ACM and PBM predictions of the pressure coefficient in the symmetry plane of the duct are compared in Fig. 7. There is a moderate adverse pressure gradient along the pressure side, which becomes favorable after approximately 40°. On the suction side of the duct, the pressure

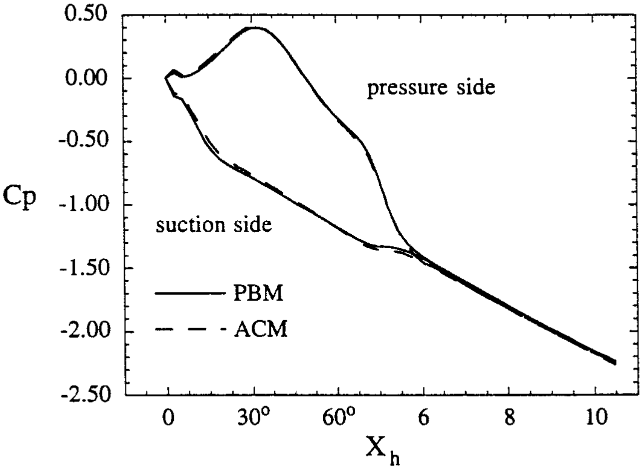

FIG. 7. Wall static pressure coefficient for laminar flow in a curved circular duct.

gradient is favorable from the inlet to the duct. Only slight differences can be seen between 15° and 30°, and between 60° and one diameter downstream from the exit of the 90° bend. On the other hand, there is almost no difference along the pressure side. Note also that in the downstream section of the duct, the C_p-lines generated by PBM and ACM are very close.

Studies of grid independence and convergence of flow solutions were performed for the circular duct geometry using a series of grid sizes. With the PBM method, a coarse grid ($16 \times 6 \times 6$ cells), a moderate grid ($28 \times 10 \times 10$ cells), and a relatively dense one ($53 \times 20 \times 20$ cells) were used. The corresponding three grids for the ACM method employed $17 \times 7 \times 7$ nodes, $29 \times 11 \times 11$ nodes, and $54 \times 21 \times 21$ nodes. The grid independence of the results is explored in Fig. 8 for the two methods. Using the PBM method, differences in predictions from the moderate and dense size grids are relatively minor, indicating that grid independent results have been obtained with the moderate size grid. At the same time, the ACM method also achieves grid independence with the moderate size grid. This demonstrates the equivalency of the two discretization approaches.

The convergence behavior of the PBM and ACM methods is compared in Fig. 9. For the PBM method, convergence is established when the maximum residual drops below the specified strict criterion. In the ACM method, variable residual indicates both velocity and pressure residuals since the continuity and momentum equations are solved in a fully coupled manner. Recall that the convergence criterion for the ACM method is that the variable residual drops by at least three orders of magnitude. Convergence of all residuals could be achieved by both methods without difficulty. In the PBM method, the coarse grid required 200 iterations for convergence, while the moderate and dense size ones converged after 350 and 650 iterations, respectively. While ACM exhibited a similar rate of convergence with PBM for the coarse and moderate grids, the ACM method converged faster (within 300 iterations) for the dense grid; the latter characteristic is perhaps the greatest advantage of the ACM method when using finer grids. As expected for both methods, the solutions on coarser grids converge faster, but are less accurate than dense grids solutions (see Fig. 9b). Note also that, for the denser grid, the ACM mass error is 100 times larger than the corresponding PBM mass error upon convergence of residuals. A detailed discussion of this discrepancy is presented in the following section.

4. DISCUSSION

4.1. *Accuracy*

Our computations on the curved duct geometries have demonstrated that both PBM and ACM methods are capable of generating results that compare satisfactorily with experimental measurements. However, in general, the PBM method compares slightly more favorably with measurements and the multigrid method than the ACM method. For example, in the downstream section of the ducts the ACM method captures the velocity distributions less accurately than the PBM method. Our observations are consistent with the findings of Pentaris *et al.* [27]. Upon convergence, the mass errors of the PBM code is 100 times lower than the mass error of the ACM code. This discrepancy does not appear to be related to grid density since results for each method have been found to be grid independent. Furthermore, both methods use the same high order discretization scheme—the third-order QUICK scheme [29]. In addition, the artificial compressibility term (which is of the same order as the variable residuals) is 1000 times smaller than the overall mass error in the ACM code upon convergence, as indicated by Figs. 9a and 9b. Hence, the larger mass error in the ACM code should be attributed to (i) the fourth-order explicit smoothing term used to stabilize the central difference scheme in the particular implementation of the ACM method adopted here and (ii) the fact that the ACM method uses the less conservative FD formulation rather than the FV formulation which is used to implement the PBM method.

Further, to quantify the contribution of the discretization formulation on the overall discrepancy, an estimation of the truncation errors produced by FD and FV formulations has been made. As shown by Leonard [26] for a one-dimensional case, the excess truncation error (TE) of the FD formulation over the FV formulation for the second or higher order discretization of the convection term is given by

$$(\mathrm{TE})_{\mathrm{FD}} - (\mathrm{TE})_{\mathrm{FV}} = \frac{1}{24}\frac{\partial^3 \phi}{\partial x^3}(\Delta x)^2 + \frac{1}{1920}\frac{\partial^5 \phi}{\partial x^5}(\Delta x)^4 + \cdots. \tag{21}$$

After the solution from the ACM method has converged (the residual has been dropped more than three orders of magnitude) and has reached grid independence, Eq. (21) was used to estimate the difference in mass errors that would be produced by FD and FV formulations in the ACM code. We found that the mass error could decrease by approximately 4% if we choose to implement the FV formulation in the ACM code.

Following the same methodology, a similar conclusion can be reached for the discretization of the diffusion terms [37]. Hence, it can be deduced that the PBM code can achieve a better mass conservation than the ACM code,

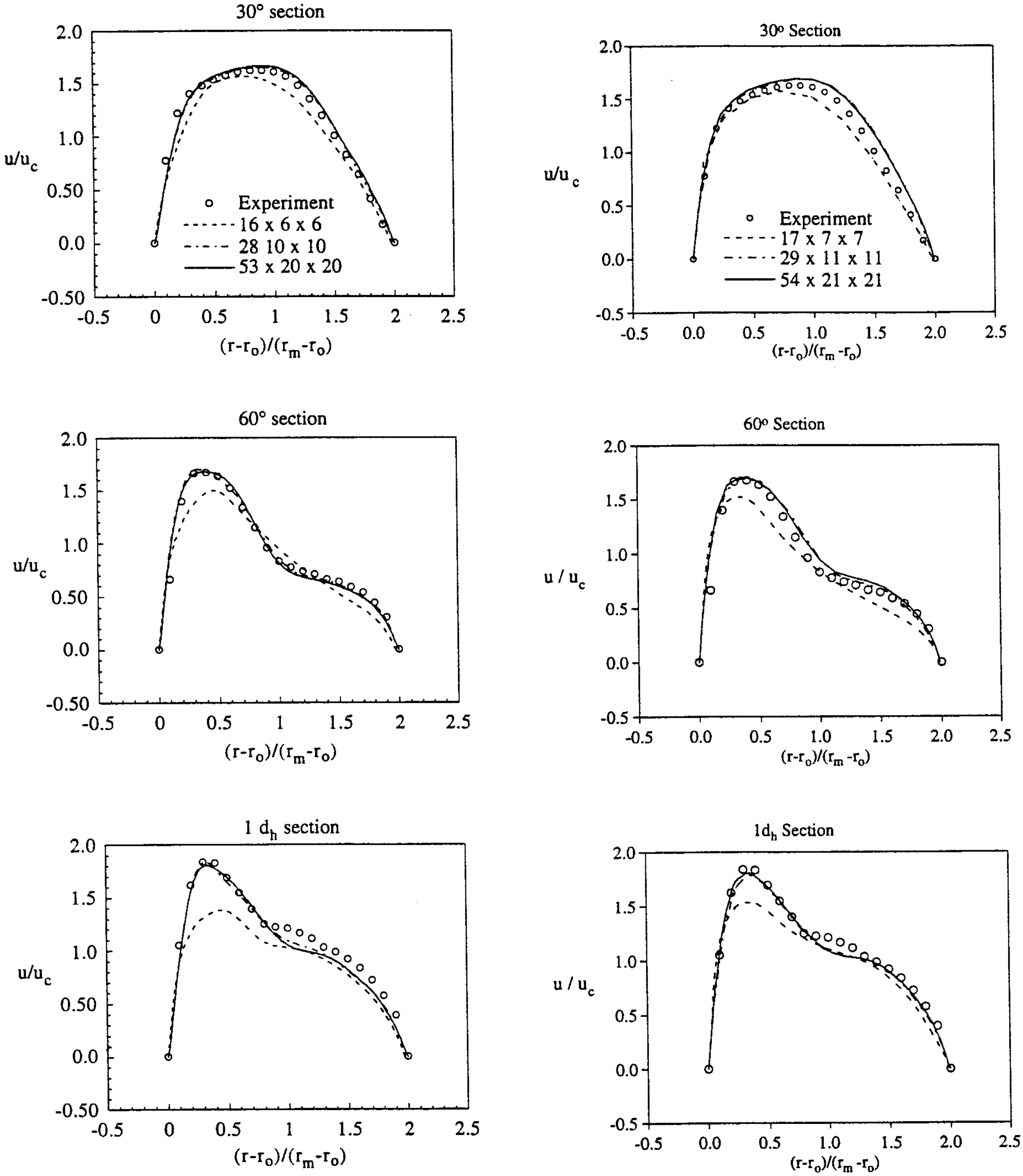

FIG. 8. Comparison of PBM (left) and ACM (right) predictions with experimental results on several grids.

even if a FV formulation is implemented with the latter method.

4.2. *Convergence Behavior*

Perhaps the most attractive feature of the ACM method is that it converges faster because it solves the continuity and momentum equations in a fully coupled manner instead of the uncoupled approach of the PBM method [27, 35]. For instance, for the dense grid calculation, the variable residual of the ACM code can be dropped by three orders of magnitude in 300 iterations; correspondingly, the maximum residual of the PBM code can be reduced to 10^{-6} in 550 iterations. However, it should be noted that, for this illustration, the PBM code is specifically vectorized and optimized for the CRAY vector platform. Due to the vectorization of the tridiagonal matrix solver, the PBM code requires more iterations to converge since the coefficients are updated at a slower

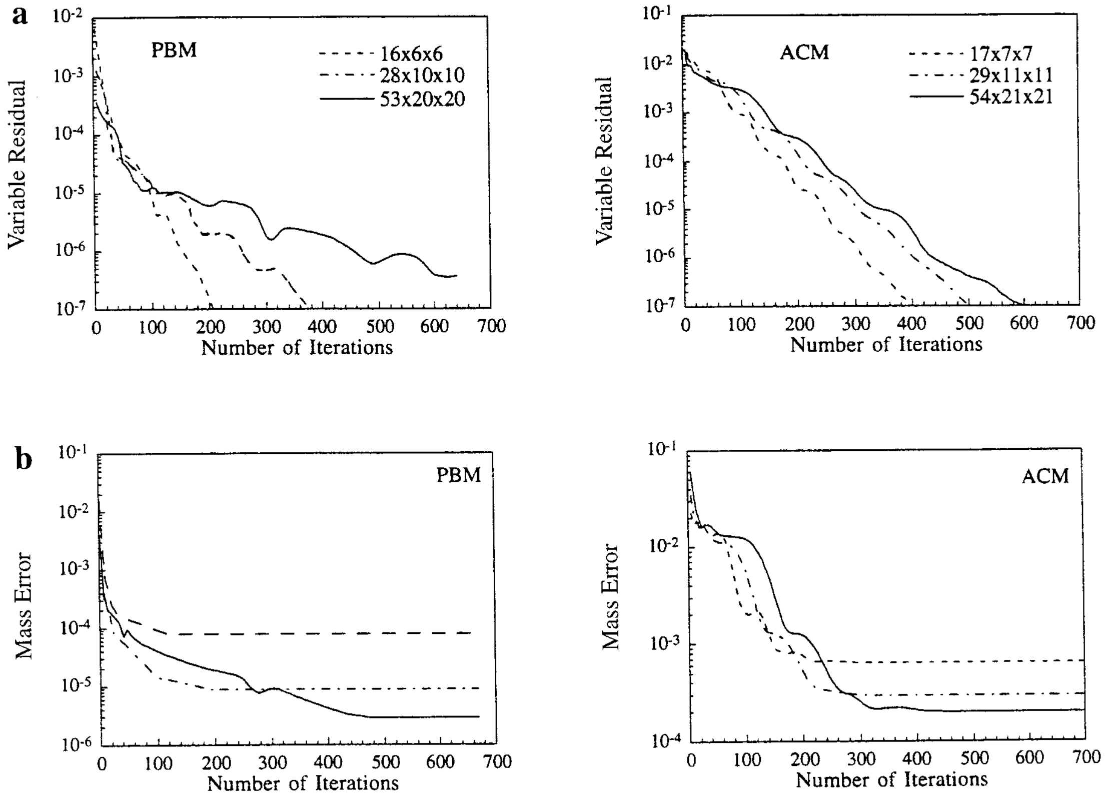

FIG. 9. Comparison of convergence characteristics of PBM (left) and ACM (right) methods on various grid sizes: (a) variable residuals; (b) mass errors.

rate. It has been estimated that the convergence rate of the PBM method could be speeded-up by up to 25% if the code were unvectorized.

4.3. *User Specified Parameters*

For laminar flow computations, there are two basic user-specified parameters in the PBM method, namely the velocity and pressure underrelaxation factors, versus one parameter in the ACM method, i.e., the artificial compressibility factor. These numerical parameters depend on the geometry, grid size, and flow field details. In general, while the optimum underrelaxation factor(s) of the PBM method may depend on the specifics of a given geometry and flow field, a safe guess can always be to set both ω_p and ω_u equal to 0.5 and expect reasonable convergence behavior. Since little intuition is required to select the underrelaxation factors, the PBM method is reasonably user-friendly.

On the other hand, the optimum artificial compressibility factor can assume a wide range of values from problem to problem. In previous studies, Rogers and Kwak [22] investigated values of β between 0.001 and 100 and recommended a value of 0.1 to be optimum for the problem of a backward-facing step. Pentaris *et al.* [27] used a value of $\beta = 1$ for two-dimensional flow problems. In our study, a value of 5 was shown to be optimum for three-dimensional laminar computations in a curved duct. While Kwak *et al.* [18] have suggested a criterion to guide the selection of the parameter β, its evaluation depends on estimates of characteristic lengths for vorticity waves, pressure waves, geometry, and Reynolds number based on eddy viscosity, many of which are not known a priori. Unfortunately, the stability and convergence are sensitive to the value of the parameter β [35]. Therefore, although the ACM method uses a single parameter to control convergence, its selection requires an experienced user and considerable intuition.

4.4. *Computational Performance*

When either the PBM or ACM method is implemented in a computer code, the overall computational performance will depend on several factors. While the number of iterations is important, the ultimate discriminator is cost; the latter depends on CPU time and memory requirements of the selected algorithms. In turn, the CPU time depends on number of iterations, number of operations per iteration, and floating point operations per second (MFLOPS). Furthermore, for any given machine, the MFLOPS count will depend on the efficiency with which the selected algorithm utilizes the computer architecture. Hence, the discussion presented below is only meant to compare the computational performance of the two codes on a representative vector supercomputer and a scalar workstation.

On a vector machine, such as the Cray-Y/MP system, the PBM code achieves very efficient utilization (190–240 MFLOPS) of the computer architecture. In particular, the inversion of the vectorized tridiagonal matrices, which consumes about 35% of the total CPU time, runs at approximately 70–140 MFLOPS (depending on grid density.) On the other hand, the 4 × 4 block-tridiagonal matrices resulting from the implicit scheme which was implemented in the ACM code do not vectorize efficiently, yielding performance in the neighborhood of 30 MFLOPS. Overall, while the number of iterations required for convergence by the PBM code is more than double the number required by the ACM code, the former requires considerably less CPU time than the latter. For example, on a relatively dense grid size (54 × 21 × 21), the PBM code consumes 1 min of CPU time, while the ACM code takes 5 min of CPU time to compute the 3D flow in a duct of circular cross section.

When the PBM and the ACM codes are run on a scalar machine, the number of operations involved in inverting tridiagonal or block-tridiagonal matrices should be comparable. As a result, the number of iterations required for convergence by each code becomes the paramount factor in determining overall CPU time. On an IBM Risc 6000 workstation running at a clock speed of 62.5 MHz, the ACM code consumes about 42 min, while the PBM code takes about 100 min for the circular duct problem on the same grid (54 × 21 × 21). Note, however, that no effort was made either to unvectorize the tridiagonal matrix solver, or to rewrite other portions of the PBM code which were developed with vectorization in mind. Furthermore, the ACM code would require approximately 16 times more dynamic storage than the PBM code since it inverts 4 × 4 block-tridiagonal matrices. The overall cost of running any of the two codes on a workstation platform would depend on the trade-off between speed and memory.

5. CONCLUSIONS

In this paper, two alternative methods for calculating three-dimensional, steady, incompressible viscous flows are critically compared. While both pressure-based and artificial compressibility methods have merits and demerits, the following conclusions can be drawn from our evaluation:

(i) Both methods produce grid independent results at the same grid resolution. Overall, both methods are capable of producing results in reasonable agreement with experimental data. However, predictions from PBM compare slightly more favorably with measurements and other well-resolved computations than the results of ACM.

(ii) Perhaps the most attractive feature of the ACM method is that it converges faster because it solves the continuity and momentum equations in a fully coupled manner. However, PBM produces converged solutions with better mass conservation than ACM. This deficiency is mainly attributed to the smoothing term which was used in the particular implementation of the ACM method that was adopted here.

(iii) The ACM method requires the use of only one adjustable numerical parameter to control convergence. However, the method's stability and convergence are sensitive to this parameter, and its selection requires an experienced user. While the PBM method typically requires two underrelaxation factors, the latter can be selected with little intuition, thus making the method more user-friendly.

(iv) The computational performance of the present codes which implemented the PBM and ACM methods depends on the available computer architecture. On a vector machine, the PBM code would run approximately five times faster due its efficient inversion of the tridiagonal matrices. On a scalar machine, the ACM code would run faster due to its reduced number of iterations required for convergence. Note however, that the ACM code would require 16 times more memory in both machines.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign for providing computer time on the CRAY-Y/MP system for this study.

REFERENCES

1. F. H. Harlow and J. E. Welch, *Phys. Fluids* **8,** 2182 (1965).
2. S. V. Patankar and D. B. Spalding, *Int. J. Heat Mass Transfer* **15,** 1787 (1972).
3. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow* (Hemisphere, Washington, DC, 1980.
4. A. Arakawa, *J. Comput. Phys.* **1,** 119 (1966).
5. C. M. Rhie and W. L. Chow, *AIAA J.* **11,** 1525 (1983).
6. C. M. Rhie, *Comput. Fluids* **13,** 443 (1985).
7. R. F. Kunz, C. M. Rhie, and R. E. Malecki, AIAA Paper 88-3005, 1988 (unpublished).
8. G. V. Hobson and B. Lakshminarayana, AIAA Paper 90-2122, 1990 (unpublished).
9. M. C. Melaaen, *Int. J. Numer. Methods Fluids* **15,** 895 (1991).
10. P. Tamamidis and D. N. Assanis, *Multidisciplinary Applications in CFD*, edited by O. Baysal ASME-FED, Vol. 129 (ASME, New York, 1991), p. 99.
11. P. Tamamidis and D. N. Assanis, ASME Paper 92-ICE-1, 1992 (unpublished).
12. P. J. Coelho and J. C. F. Pereira, *Int. J. Numer. Methods Fluids* **14,** 423 (1992).

13. P. Tamamidis and D. N. Assanis, *Numer. Heat Transfer, Part B* **24,** 57 1993.
14. P. Tamamidis and D. N. Assanis, *SAE Trans.: J. Engines* **102**(3), 1621 (1993).
15. K. M. Smith and S. P. Vanka, AIAA Paper 92-0096, 1992 (unpublished).
16. A. J. Chorin, *J. Comput. Phys.* **2,** 12 (1967).
17. J. L. Steger and P. Kutler, *AIAA J.* **15,** 581 (1977).
18. D. Kwak, J. L. C. Chang, S. P. Shanks, and S. R. Chakravarthy, *AIAA J.* **24,** 390 (1986).
19. S. E. Rogers, D. Kwak, and U. Kaul, *Appl. Math. Modelling* **11,** 35 (1987).
20. R. M. Beam and R. F. Warming, *J. Comput. Phys.* **22,** 87 (1976).
21. S. E. Rogers, D. Kwak, and C. Kiris, *AIAA J.* **29,** 603 (1991).
22. S. E. Rogers and D. Kwak, *Appl. Numer. Math.* **8,** 43 (1991).
23. J. D. Ramshaw and V. A. Mousseau, *Comput. & Fluids* **18**(4), 361–367.
24. E. Turkel, *J. Comput. Phys.* **72,** 277 (1987).
25. M. Rosenfeld and D. Kwak, *Intl. J. Numer. Methods Fluids* **13**(10), 1311 (1991).
26. B. P. Leonard, *Appl. Math. Modelling* **18** (1994).
27. A. Pentaris, K. Nikolados, and S. Tsangaris, *Int. J. Numer. Methods Fluids* **19,** 1013 (1994).
28. S. E. Rogers, D. Kwak, and J. L. C. Chang, *NASA Tech. Memo* 100012, 1987 (unpublished).
29. B. P. Leonard, *Comput. Methods Appl. Mech. Eng.* **19,** 59 (1979).
30. P. Tamamidis, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1993 (unpublished).
31. J. P. Vandoormal and G. D. Raithby, *Numer. Heat Transfer* **7,** 147 (1984).
32. P. K. Khosla and S. G. Rubin, *Comput. & Fluids* **83,** 207 (1974).
33. J. A. C. Humphrey, A. M. K. P. Taylor, and J. H. Whitelaw, *J. Fluid. Mech.,* **83,** 509 (1977).
34. J. F. Thompson, Z. V. A. Warsi, and C. W. Mastin, *Numerical Grid Generation, Foundations and Applications* (North-Holland, New York, 1985).
35. B. Lakshminarayana, *Trans. ASME J. Fluids Eng.* **113,** 315.
36. M. M. Enayet, M. M. Gibbson, A. M. K. P. Taylor, and M. Yianneskis, NASA Contractor Report 3551, 1982 (unpublished).
37. B. P. Leonard, NASA Tech. Memo 106402, 1993 (unpublished).