INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **29**: 515–533 (1999)

CONVERGENCE ACCELERATION OF SEGREGATED ALGORITHMS USING DYNAMIC TUNING ADDITIVE CORRECTION MULTIGRID STRATEGY

FRANZ ZDRAVISTCH[a,1], CLIVE A.J. FLETCHER[a,*,2] AND MASUD BEHNIA[b,3]

[a] *Centre for Advanced Numerical Computation in Engineering and Science (CANCES), The University of New South Wales, Sydney, NSW 2052, Australia*
[b] *School of Mechanical and Manufacturing Engineering, The University of New South Wales, Sydney, NSW 2052, Australia*

SUMMARY

A convergence acceleration method based on an additive correction multigrid–SIMPLEC (ACM-S) algorithm with dynamic tuning of the relaxation factors is presented. In the ACM-S method, the coarse grid velocity correction components obtained from the mass conservation (velocity potential) correction equation are included into the fine grid momentum equations before the coarse grid momentum correction equations are formed using the additive correction methodology. Therefore, the coupling between the momentum and mass conservation equations is obtained on the coarse grid, while maintaining the segregated structure of the single grid algorithm. This allows the use of the same solver (smoother) on the coarse grid. For turbulent flows with heat transfer, additional scalar equations are solved outside of the momentum–mass conservation equations loop. The convergence of the additional scalar equations is accelerated using a dynamic tuning of the relaxation factors. Both a relative error (RE) scheme and a local Reynolds/Peclet (ER/P) relaxation scheme methods are used. These methodologies are tested for laminar isothermal flows and turbulent flows with heat transfer over geometrically complex two- and three-dimensional configurations. Savings up to 57% in CPU time are obtained for complex geometric domains representative of practical engineering problems. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: multigrid; convergence acceleration; segregated algorithms; finite volume method; dynamic tuning

1. INTRODUCTION

The application of computational fluid dynamics in design mode to practical engineering problems requires accurate algorithms, while keeping memory storage and CPU time to a minimum. In the case of incompressible flow applications, the use of segregated methods in which a set of decoupled equations is solved iteratively is usually preferred since they have lower storage requirements. However, when a large number of grid points are used, the rate of convergence of segregated iterative methods deteriorates as the solution progresses to smaller residual levels. This occurs because iterative solvers are very efficient in reducing the high

* Correspondence to: Centre for Advanced Numerical Computation in Engineering and Science (CANCES), The University of New South Wales, Sydney, NSW 2052, Australia.
[1] Tel.: +61 2 93855744; fax: +61 2 96627792; e-mail: franzz@cluster.cances.unsw.edu.au
[2] Tel.: +61 2 93855745; fax: +61 2 96627792; e-mail: c.fletcher@unsw.edu.au
[3] Tel.: +61 2 93854253; fax: +61 2 96631222; e-mail: m.behnia@unsw.edu.au

CCC 0271–2091/99/050515–19$17.50

Received June 1997
Revised September 1997

frequency Fourier modes of the error, but the low frequency modes of the error are reduced very slowly [1]. Multigrid methods assist in overcoming this problem by solving a set of associated equations on coarser grid levels so the whole spectrum of error frequency components is reduced at a similar rate.

Multigrid methods have achieved significant efficiency improvements in well-behaved problems. For example, a factor of four is reported [2] when using a 257×257 uniform grid for laminar flow in a driven cavity. However, in this paper the constraint that practical application problems, like power station boilers, usually involve turbulent flow in complex geometries and distorted non-uniform grids is accepted. In this practical situation, which methods are effective and what is a realistic gain in efficiency? Perhaps it is sufficient to increase the robustness of the method, i.e. it is reliable across a broad class of problems, with only minor gains in efficiency.

The additive correction multigrid (ACM) [3,4] strategy is a multigrid-type formulation that has been proposed for the solution of the discretised incompressible Navier–Stokes equations in complex geometries. Previous work [5,6] presented convergence acceleration methodologies based on the ACM formulation in which segregated solvers are used on the fine grid, while direct solvers were used on the coarse grids. In the present paper, an ACM–SIMPLEC (ACM-S) procedure with dynamic tuning of the relaxation coefficients is proposed. In the ACM-S procedure [7], a velocity potential method is used as a solver on the fine grid, while the corrections from the velocity potential correction equation are directly introduced into the momentum correction equations on the coarse grid. This procedure presents several practical advantages: there is no need for discretisation and grid information storage on the coarse grids, the coarse grid equations already include the boundary conditions information and maintain the conservative form of the original fine grid equations, and the segregated structure of SIMPLEC-type algorithms is preserved on the coarse grid. The convergence of additional scalar equations for turbulence parameters and temperature is accelerated using variable relaxation coefficients. Two methods, a relative error scheme (RE) and a local Reynolds/Peclet relaxation scheme (ER/P) [8] are used. The effectiveness and robustness of this methodology is demonstrated for two- and three-dimensional laminar and turbulent flows with heat transfer simulations using stretched body-fitted grids.

2. NUMERICAL METHOD

2.1. Governing equations

The steady state Reynolds-averaged Navier–Stokes equations, retaining only dominant terms, can be written as:

$$\frac{\partial}{\partial x_i}(\rho u_i) = 0, \tag{1}$$

$$\frac{\partial}{\partial x_i}(\rho u_i u_j + \delta_{ij} p_j - \tau_{ij} + \rho \overline{u_i' u_j'}) = 0, \tag{2}$$

$$\frac{\partial}{\partial x_i}\left(\rho C_p u_i T - \kappa_{\text{eff}} \frac{\partial T}{\partial x_i}\right) = 0. \tag{3}$$

Here, the standard nomenclature found in the literature is used [2]. The term κ_{eff} is the effective thermal diffusivity, which includes the turbulent eddy diffusivity, and accounts for the

conductive heat transfer exchange in laminar or turbulent regimes. Equation (2) contains Reynolds stresses that are modelled via the $k-\varepsilon$ turbulence model [9]. A two-layer wall function [10] is used to obtain the solution and boundary conditions in the vicinity of solid surfaces.

2.2. Discretisation method and single grid solution algorithm

The governing equations can be written symbolically as

$$\frac{\partial F^{i\ell}}{\partial x_i} = S^{\ell}, \tag{4}$$

where the flux $F^{i\ell}$ is defined as

$$F^{i\ell} = \rho u^i \psi^{\ell} - \mu \frac{\partial \psi^{\ell}}{\partial x_i}, \tag{5}$$

and where ψ^{ℓ} is a general scalar, like u in the momentum equation. Discretising Equation (4) directly in physical space using the concept of area vectors [8] results in

$$\sum_{n=1}^{6} F_n^{i\ell} \cdot A_n = |\underline{\boldsymbol{J}^{-1}}| \langle S^{\ell} \rangle, \tag{6}$$

where A_n represents the area vector of the nth face, $|\underline{\boldsymbol{J}^{-1}}|$ is the effective volume of the control volume and $\langle S^{\ell} \rangle$ is the source term evaluated at the centre of the control volume, while the scalar product $F_n^{i\ell} \cdot A_n$ is evaluated at each face. Using Equation (5), one component of $F_n^{i\ell} \cdot A_n$ becomes

$$F_n^{i\ell} \cdot A_n = \{\rho \underline{u} \cdot A^{(i)} \psi^{\ell} - \mu A^{(i)} \cdot \nabla \psi^{\ell}\}_n, \tag{7}$$

where $\underline{u} \cdot A^{(i)}$ is the normal flux through the nth surface. The second term in Equation (7) can be evaluated in physical or generalised co-ordinates. The latter approach is preferred, resulting in

$$A^{(i)} \cdot \nabla \psi^{\ell} = \frac{A^{(i)} \cdot A^{(j)}}{|\underline{\boldsymbol{J}^{-1}}|} \frac{\partial \psi^{\ell}}{\partial \xi^j}. \tag{8}$$

Equation (7) can be expressed as

$$F_n^{i\ell} \cdot A_n = C_n^i \psi_n^{\ell} - D_n^{ij} \left[\frac{\partial \psi^{\ell}}{\partial \xi^j} \right]_n, \tag{9}$$

where

$$C_n^i = \{\rho \underline{u} \cdot A^{(i)}\}_n, \tag{10}$$

$$D_n^{ij} = \mu \frac{A^{(i)} \cdot A^{(j)}}{|\underline{\boldsymbol{J}^{-1}}|}, \tag{11}$$

are the convection and diffusion coefficients respectively. These coefficients are evaluated at the centre of each face, while the dependent variables ψ^{ℓ} are evaluated at the centre of each control volume. More details of the form of these terms discretisations can be found elsewhere [11].

On the solution grid, the discretised equations are solved sequentially at each iteration to obtain the dependent variables, using the modified strongly implicit (MSI) algorithm [12]. The solution for the velocity obtained from the momentum equations is denoted $\vec{q}^*$, where

$\vec{q}^{*T} = [u, v, w]$ and u, v and w are the velocity components in the x-, y- and z-directions respectively. To satisfy continuity, it is necessary that

$$\nabla\cdot(\rho\vec{q})^{n+1} = (\rho\vec{q})^{*} + (\rho\vec{q})^{c}. \tag{12}$$

The correction velocity field is obtained from the velocity potential

$$\vec{q^{c}} = \alpha\nabla\varphi. \tag{13}$$

The pressure and density can also be obtained from φ [13]. This leads to the following transport equation for φ,

$$\delta\frac{\partial}{\partial x_i}\left\{\frac{u_i^{*}\varphi}{(a^{*})^2}\right\} - \alpha\frac{\partial}{\partial x_i}\left\{\rho^{*}\frac{\partial\varphi}{\partial x_\iota}\right\} \equiv \nabla\cdot(\rho q^{*}) - \vec{\gamma}\frac{\partial^4\varphi}{\partial\xi^{i4}}. \tag{14}$$

The last term on the right-hand-side of Equation (14) is introduced to assist in damping of any pressure oscillations due to the fact that the discretisation algorithm is non-staggered. The terms α, δ and $\vec{\gamma}$ are relaxation factors and a^{*} is the sound speed.

2.3. The ACM applied to a single equation

The linear discretised equations obtained from Equations (6) and (14) on the solution grid *ijk* for incompressible flow have the general form:

$$a_{ijk}^{p}\Psi_{ijk}^{p} = \sum_{\mathrm{NB}} a_{ijk}^{\mathrm{nb}}\Psi_{ijk}^{\mathrm{nb}} + S_{ijk}, \tag{15}$$

where Ψ_{ijk} is the variable of interest, i.e. u for the x momentum equation, nb indicates the neighbouring points around point p and S_{ijk} is the source term. Assuming that $\tilde{\Psi}_{ijk}^{p}$ is the last updated value of Ψ_{ijk}^{p} on the fine grid, a correction δ_{lmn}, which is constant over the coarse grid block *lmn* where the control volume *ijk* is contained, is used to update Ψ_{ijk}^{p}, thus

$$\Psi_{ijk}^{p} = \tilde{\Psi}_{ijk}^{p} + \delta_{lmn}. \tag{16}$$

Introducing Equation (16) into (15), and adding all the equations in the coarse grid block *lmn* results in the following equation for the coarse grid block:

$$a_{lmn}^{p}\delta_{lmn}^{p} = \sum_{\mathrm{NB}} a_{lmn}^{\mathrm{nb}}\delta_{lmn}^{\mathrm{nb}} + \tilde{S}_{lmn}. \tag{17}$$

In this case, the coefficient a_{lmn}^{p} is computed by adding all the coefficients a_{ijk}^{p} and a_{ijk}^{nb} that belong to the *lmn* block, while the coefficients a_{ijk}^{nb} are obtained by adding the corresponding neighbouring coefficients on the *ijk* grid that lay outside the *lmn* block. Also, the source term $\tilde{S}_{lmn}$ has the form [3]:

$$\tilde{S}_{lmn} = \sum_{lmn}\left[S_{ijk} + \sum_{\mathrm{NB}} a_{ijk}^{\mathrm{nb}}\tilde{\Psi}_{ijk}^{\mathrm{nb}} - a_{ijk}^{p}\tilde{\Psi}_{ijk}^{p}\right], \tag{18}$$

where Σ_{lmn} is the sum over all the control volumes on the *ijk* grid that belong to the *lmn* block. The coarse grid block correction equation (17) presents a similar structure as Equation (15) on the fine grid, and the corrections are kept segregated for each individual equation. This allows the use of the same type of relaxation scheme for the coarse grid system of equations.

Work by Gjesdal [14] demonstrated that the ACM methodology is equivalent to the standard cell-centred multigrid algorithm [1] if the prolongation and restriction operators are based on a piecewise constant interpolation. Numerical experiments prove that grid-indepen-

dent convergence rate cannot be obtained using this method for diffusion-dominated problems [15]. However, the method is effective for problems with a large number of grid points. The efficiency of the method can be enhanced by using more accurate restriction and prolongation operators [14].

The standard ACM algorithm has been modified in the present work by using a volume-weighted interpolation of the correction during the prolongation to the solution grid (ACM-I). The objective is to provide a more smooth distribution of δ_{IJK} in control volumes that have neighbours which belong to an adjacent *IJK* block. The correction δ_{IJK} is considered constant in the *IJK* block when the coarse grid correction equations are formed, maintaining the fundamental characteristics of the ACM method. However, for control volumes located on an edge of an *IJK* block, the correction from the coarse grid for that point *ijk* is computed as

$$\delta_{ijk} = \frac{1}{\mathrm{NB}_{\max}} \sum_{\mathrm{NB}=1}^{\mathrm{NB}_{\max}} \frac{(\delta_{IJK} V_{ijk} + \delta_{\mathrm{NB}} V_{\mathrm{nb}})}{V_{ijk} + V_{\mathrm{nb}}} . \tag{19}$$

In this case, V_{ijk} is the volume of the control volume, NB refers to the number of adjacent face neighbours on the coarse grid *IJK*, and nb refers to the neighbouring control volume on the fine grid contained in the adjacent coarse grid block NB. Equation (19) is the average of the weighted interpolation between the point *ijk* and each neighbouring control volume in the adjacent *IJK* block. This interpolation allows for a smoother distribution of the corrections over the whole domain.

2.4. Results using the ACM-I for a single equation

The convergence history of the velocity potential equation for two test problems consisting of a two-dimensional laminar flow over a square wall-driven cavity and a two-dimensional

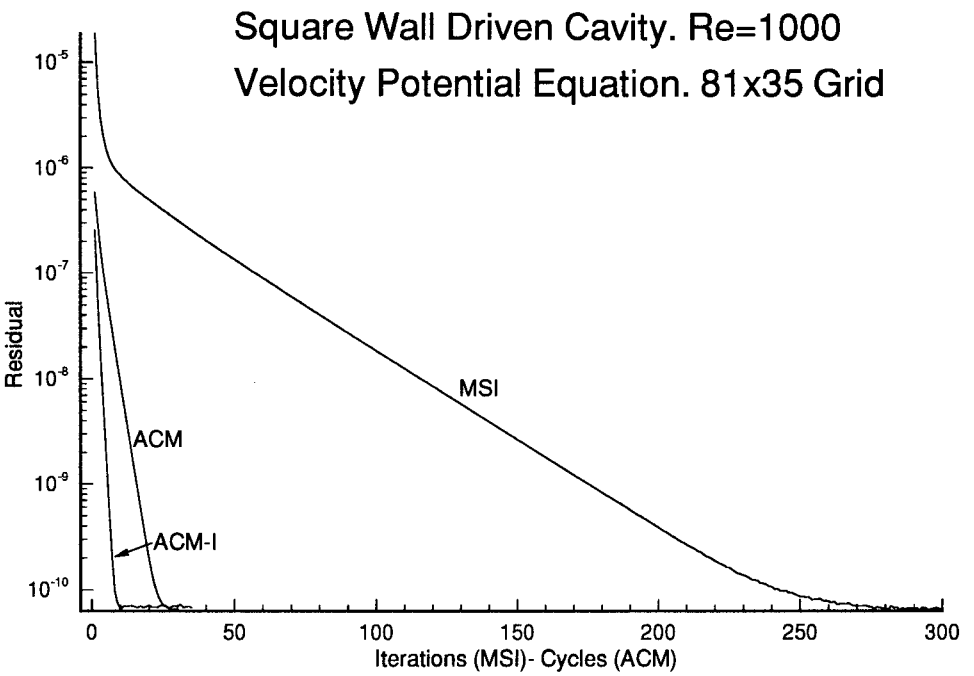

Figure 1. Convergence history.

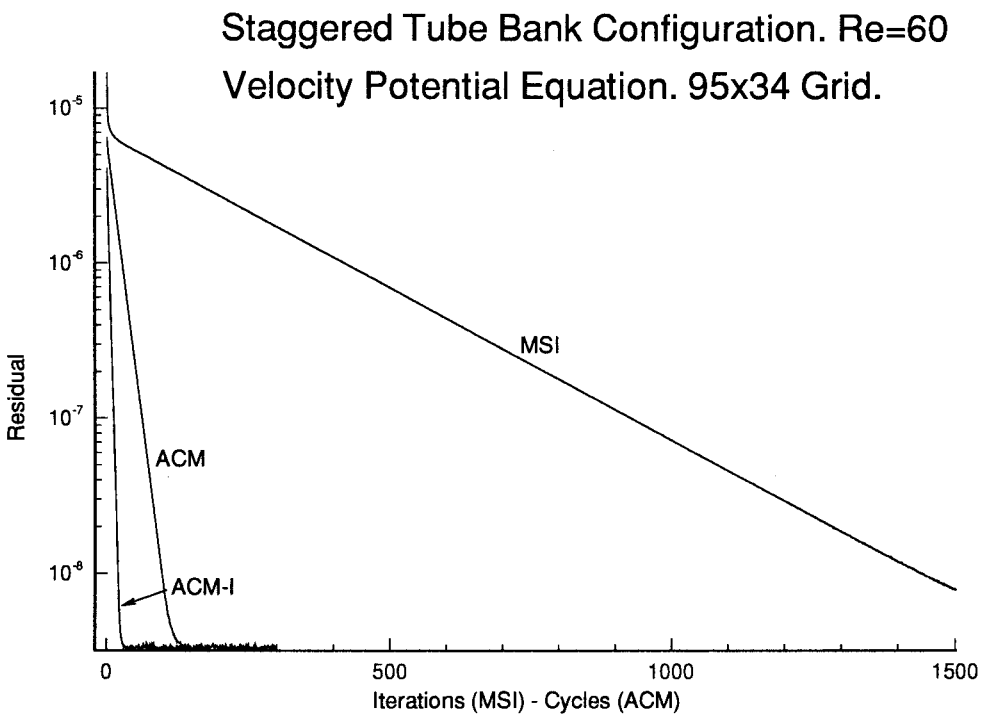

Figure 2. Convergence history.

laminar flow over a staggered tube bank configuration are presented in Figures 1 and 2. To obtain these results, the iteration process of the SIMPLEC cycle is frozen after 15 iterations and the velocity potential equation normalised residual is converged to round-off error levels. In this case, the objective is to assess the effectiveness of the algorithm for an individual equation. The x-axis represents the number of iterations using MSI and the number of cycles using the standard ACM and the ACM with an interpolated prolongation operator (ACM-I). For both test cases, three grid levels with a coarsening of 2×2 were used with the V-adaptable cycle, which were found to be the most numerically efficient. Each ACM cycle is computationally more expensive (typically four to six times) than each MSI iteration, resulting in a reduction in CPU time of the order of 75% for both cases. The results show that the use of interpolated prolongation increases the efficiency of the algorithm substantially at a very small additional computational cost.

These results indicate that the use of the ACM methodology is effective when the residual of discretised velocity potential equation is reduced to round-off or very low residual levels. However, the residuals of the individual equations are typically reduced only two or three orders of magnitude during each SIMPLEC cycle before the dependent variables are updated and the equation coefficients are recomputed. It can be seen from Figures 1 and 2 that for those levels of residual reduction, the MSI and ACM algorithms present comparable performance. The use of the ACM method on the individual pressure and energy equations has shown to improve the overall convergence of the SIMPLEC cycles for problems with large recirculation regions and with a line Gauss–Seidel iterative solver as smoother [16]. However, no significant advantages have been found in this work by using the ACM-I in the velocity potential and energy equations for the overall convergence of the SIMPLEC cycle with the

MSI as a smoother. This has motivated the extension of the ACM method to include the influence of the corrections to the velocity potential equation in the momentum correction equations on the coarse grid. This strategy will be presented in the next section.

2.5. The dynamic tuning ACM-S procedure

The objective of this strategy is to link the velocity potential and the momentum correction equations on the coarse grid and to maintain the iterative form of the SIMPLEC algorithm, allowing for the same smoother (MSI) to be used on the coarse grid. Therefore, this methodology can be interpreted as an 'add-on' convergence acceleration algorithm in which the fundamental structure of the solver is maintained.

The corrections to the velocity field from the velocity potential correction equation on the coarse grid are obtained using a similar approach. The discrete form of Equation (13) on the fine grid is:

$$\tilde{q}^c_{ijk} = \alpha\left(b^p_{ijk}\tilde{\varphi}^p_{ijk} + \sum_1^6 b^{\mathrm{nb}}_{ijk}\tilde{\varphi}^{\mathrm{nb}}_{ijk} \right), \tag{20}$$

where $\tilde{q}^c_{ijk}$ is the scalar component of the velocity correction for each component of the velocity field, b_{ijk} are coefficients obtained from geometrical and discretisation parameters, which are constant at each momentum–velocity potential equations cycle, and φ_{ijk} is the velocity potential. Defining $\delta\varphi_{lmn}$, the correction to φ_{ijk} from the coarse grid, the correction to the velocity components from the velocity potential equation on the fine grid is:

$$q^c_{ijk} = \tilde{q}^c_{ijk} + f^p_{ijk}(\delta\varphi_{lmn}), \tag{21}$$

where

$$f^p_{ijk}(\delta\varphi_{lmn}) = b^p_{ijk}\varphi^p_{lmn} + \sum_1^6 b^{\mathrm{nb}}_{ijk}\delta\varphi^{\mathrm{nb}}_{ijk}. \tag{22}$$

Introducing $f^p_{ijk}(\delta\varphi_{mn})$ as an explicit term into Equation (16), and adding all the contributions from the block *lmn*, the source term of the coarse grid correction equation (18) becomes:

$$S_{lmn} = \sum_{lmn}\left[S_{ijk} + \sum_{\mathrm{NB}} a^{\mathrm{nb}}_{ijk}(\tilde{\Psi}^{\mathrm{nb}}_{ijk} + f^{\mathrm{nb}}_{ijk}(\delta\varphi_{lmn})) - a^p_{ijk}(\tilde{\Psi}^p_{ijk} + f^p_{ijk}(\delta\varphi_{lmn})) \right]. \tag{23}$$

The procedure outlined above allows for the convergence acceleration of the momentum velocity potential equations loop. Only the velocity potential coarse grid correction function f_{ijk} is included in the momentum coarse grid correction equations, since the aim of the algorithm is to improve the momentum and continuity coupling on the coarse grid. The value of $\tilde{q}^c_{ijk}$ results from the linearised equation coefficients obtained at the previous SIMPLEC iteration, resulting in a slower convergence if the full correction q^c_{ijk} is used.

The discretised turbulent transport and energy equations are also solved in a segregated manner and the convergence of the iterative process is accelerated using dynamic tuning of the relaxation factors. Numerical experiments performed by the authors show that the ACM does not improve the convergence rate on these equations, due to their highly hyperbolic character. The same conclusions have been established for the turbulence transport equations elsewhere [16]. Two acceleration schemes based on the relative error and the local Reynolds/Peclet number were implemented. In these schemes, the relaxation coefficients used to update the solution at every iteration are modified between a maximum and minimum value according to the change in the solution.

For each individual discretised equation, the dependent variable at step $n+1$ is updated using the following relation

$$\psi^{n+1} = (1-\omega_4)\psi^n + \omega_4\psi^{n+1}, \tag{24}$$

where ω_4 is the relaxation factor ($0 < \omega_4 < 1$), and ψ^{n+1} on the RHS is the updated value of ψ obtained after iterating using the MSI algorithm. In the standard MSI algorithm, the relaxation factors are kept constant throughout the domain and the whole iteration process. However, as the solution field approaches the converged solution, higher values of ω_4 could be adopted without introducing instabilities. In the RE scheme, the relaxation coefficient ω_4 is dynamically tuned using the following algorithm

$$\omega_4 = \omega_{4_{\max}} \text{ if } \epsilon < 0.001, \tag{25}$$

$$\omega_4 = \omega_{4_{\min}} \text{ if } \epsilon > 0.01, \tag{26}$$

for $0.001 \leq \epsilon \leq 0.01$, ω_4 varies linearly between the maximum ($\omega_{4_{\max}}$) and the minimum ($\omega_{4_{\min}}$) values and is calculated using the following relation

$$\omega_4 = S_1(\epsilon - 0.001) + \omega_{4_{\max}}, \tag{27}$$

where

$$S_1 = \frac{(\omega_{4_{\min}} - \omega_{4_{\max}})}{0.009}, \tag{28}$$

and the relative error ϵ is obtained from

$$\epsilon = \frac{\|\Delta\psi^n\|}{\|\psi^n\|}, \tag{29}$$

where $\Delta\psi^n = \Psi^n - \Psi^{n-1}$ and $\|\cdot\|$ is the Eucledian norm. Therefore, ω_4 varies between each iteration n but is constant over the computational domain. The choice of the bounds between 0.01 and 0.001 for ϵ has been made to ensure numerical stability.

In the cell Reynolds/Peclet (ER/P) method, the relaxation coefficient is computed using

$$\omega_4 = \frac{(\omega_{4_{\min}} Re_{ijk} + \omega_{4_{\max}})}{1 + Re_{ijk}}, \tag{30}$$

where Re_{ijk} is the cell Reynolds or Peclet number. Therefore, the value of the relaxation coefficients is adjusted at every iteration and is varying over the computational domain. For large values of Re_{ijk} (large local advection), $\omega_4 \to \omega_{4_{\min}}$ decreasing the convergence speed and enhancing stability, and conversely for small values of Re_{ijk}.

Either the RE or the ER/P are applied to the turbulence closure, temperature and additional scalars equations. Also, these algorthims are applied to the momentum and velocity potential equations for initial stages of the iteration process (up to 30 iterations) when the ACM-S is less effective. The iterative solution procedure for all the discretised equations is as follows:

1. Relax the momentum and velocity potential equations on the fine grid using SIMPLEC, while the residual reduction is satisfactory.
2. Compute the velocity potential correction equation on the coarse grid.
3. Relax the velocity potential correction equation on the coarse grid using MSI.
4. Compute $f_{ijk}(\delta\varphi_{IJK})$ and $\bar{S}_{IJK}$ for the momentum coarse grid correction equations.
5. Relax the momentum coarse grid correction equations while the residual reduction is satisfactory.

6. Prolong the corrections to the fine grid.
7. Solve the turbulence transport and energy equations using MSI and dynamic relaxation.
8. Update the fluid properties and equations coefficients. Return to 1. This sequence is repeated until final convergence is achieved.

The procedure above is applied using different degrees of coarsening following the V-cycle schedule.

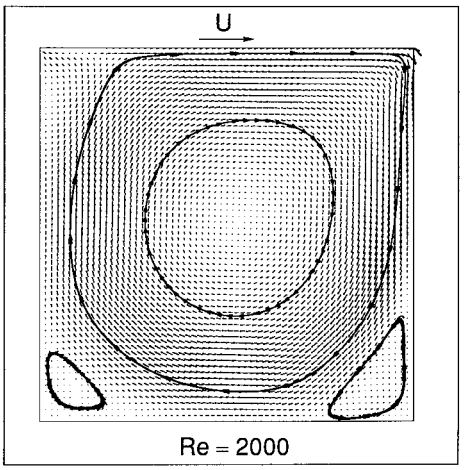

Figure 3. Velocity vectors and streamlines; square wall-driven cavity.

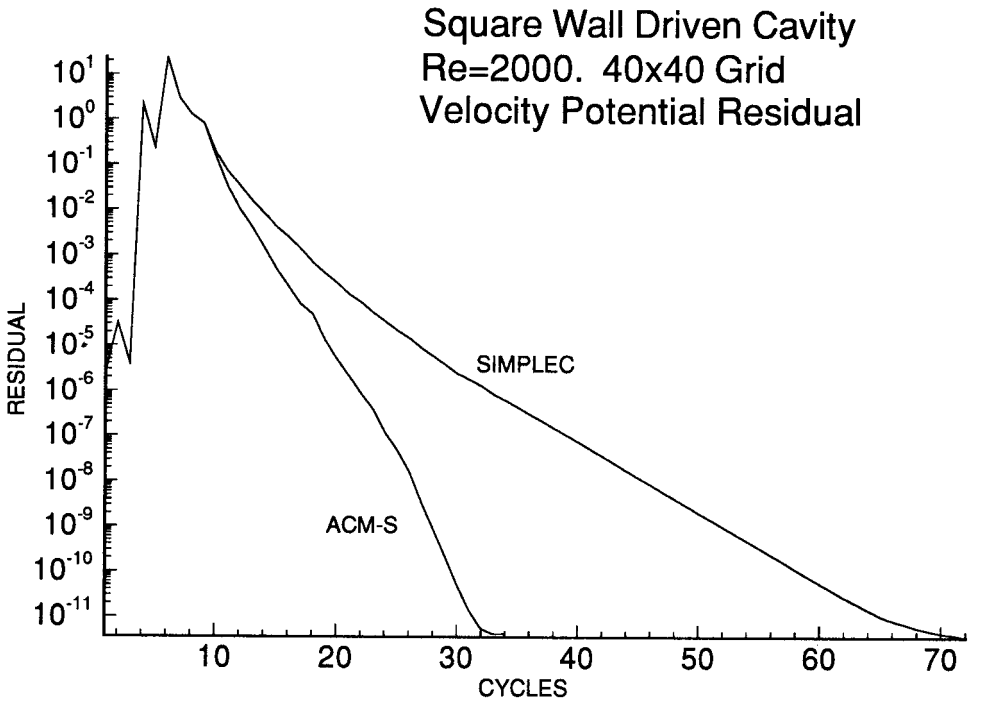

Figure 4. Convergence history.

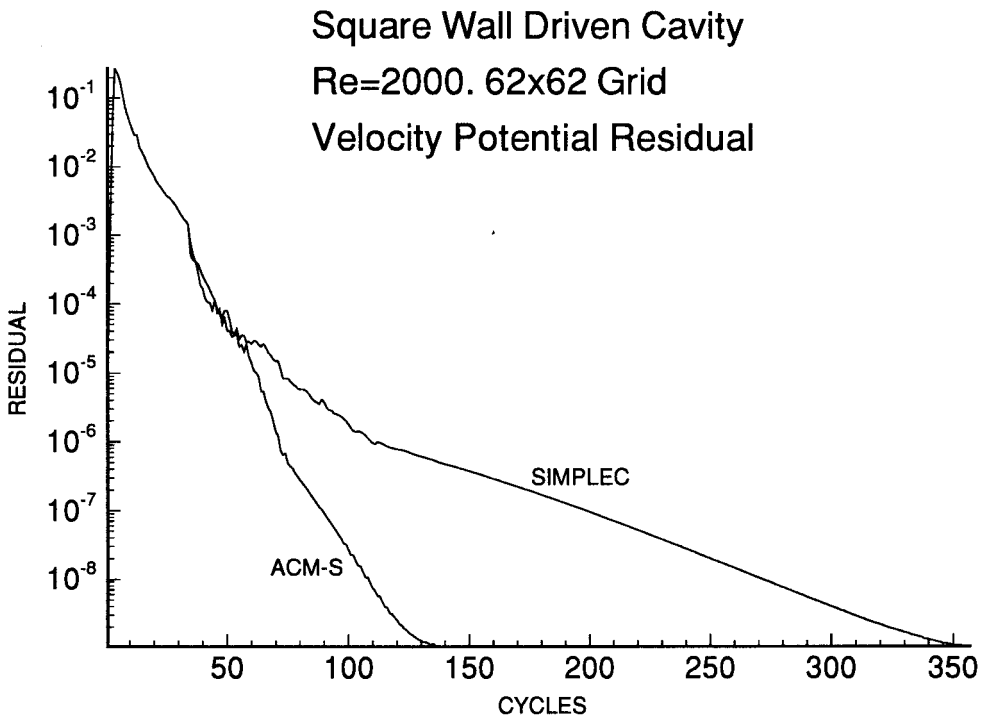

Figure 5. Convergence history.

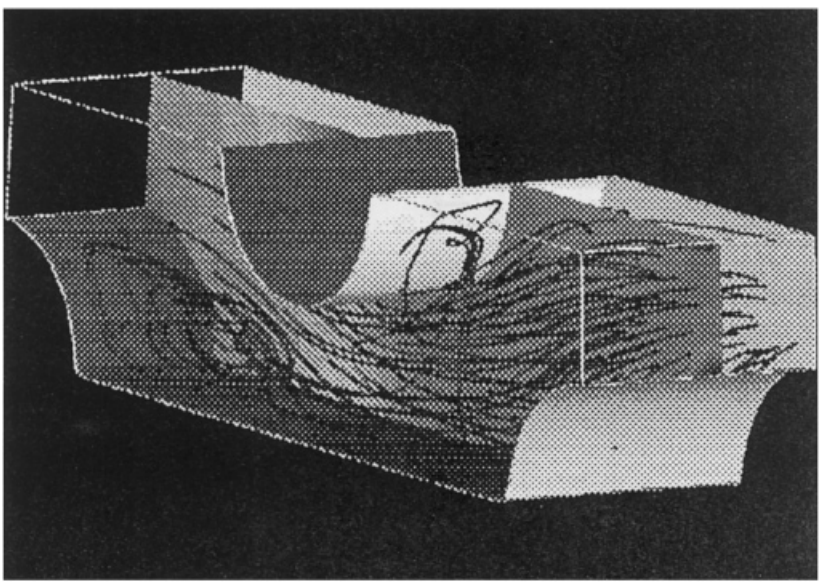

Figure 6. Stream traces; 3D tube bank configuration.

3. RESULTS AND DISCUSSION

3.1. Laminar flow results

The algorithm proposed above has been tested for two laminar flow cases: the flow simulation over a 2D unit square wall-driven cavity and for a 3D tube bank configuration. The former case results in a strongly elliptic flow and because of the simplicity of the grid is usually a benchmark problem used to test multigrid methods. The latter is a good case to test the performance of the algorithm for a more hyperbolic-type flow on a body-fitted non-uniform mesh. For both test cases, only the ACM-S algorithm was used.

The velocity vectors and streamlines for the square wall-driven cavity problem for $Re = 2000$ with a 62×62 grid are presented in Figure 3. These results agree qualitatively well with other simulations reported by Cassulli [17] for a similar grid. Convergence history results for a 40×40 grid and a 62×62 grid using SIMPLEC and the ACM-S for the velocity potential equation residual are presented in Figures 4 and 5 respectively. Here, the number of cycles represents the number of visits to the fine grid, irrespective of whether the correction from the coarse grid is used or not, so there is the same basis of comparison for both results. For these cases, three grid levels were used with a coarsening of 2×2. For the 40×40 grid case, 33 cycles are required to achieve round-off error convergence when the correction for the coarse grid is used, while 71 cycles are required if only MSI is used. Similarly, for the 62×62 grid case, 350 or 135 cycles are needed to achieve convergence whether the correction from the coarse grid is or is not used respectively. This results show that convergence acceleration obtained with the ACM-S is grid-dependent, with larger savings being obtained for larger grids.

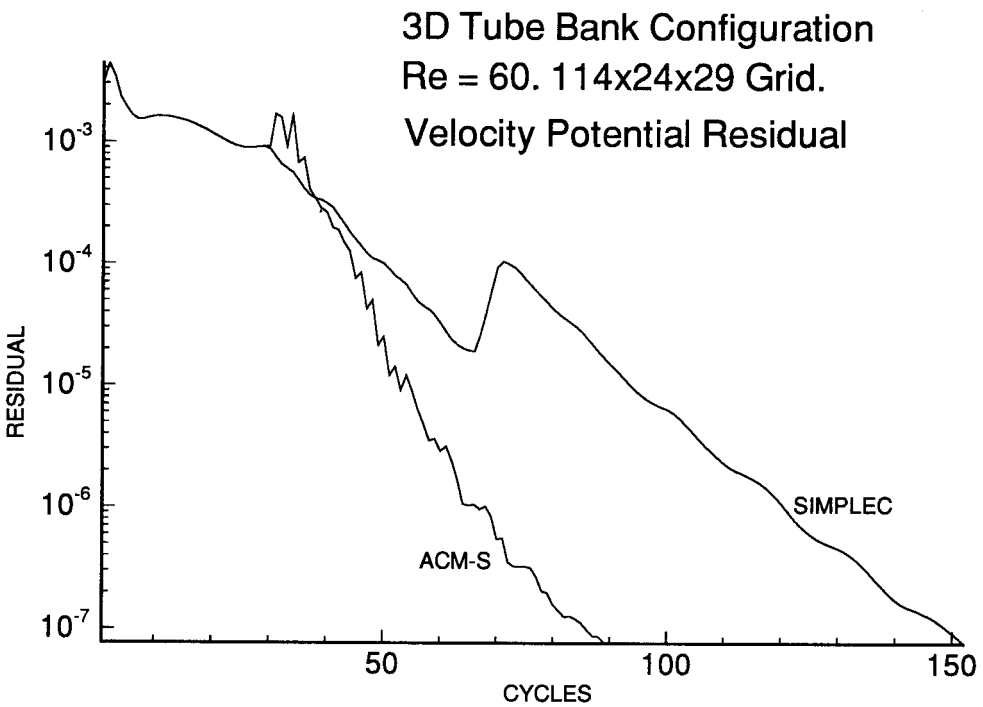

Figure 7. Convergence history.

Table I. Performance of the ACM-S algorithm

Test case	Grid size	SIMPLEC (CPU; s)	ACM-S (CPU; s)
2D square cavity	40×40	103.2	88.2
2D square cavity	62×62	1327.3	907.6
3D tube bank	$114 \times 24 \times 29$	5836.7	4753.2

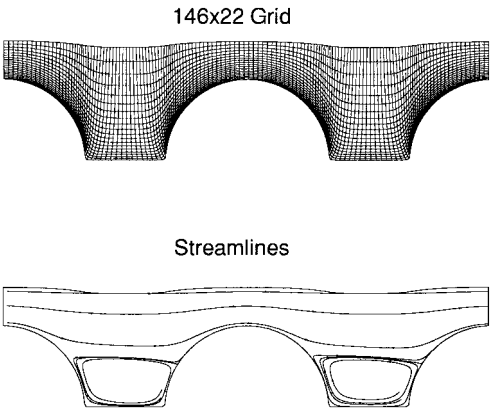

Figure 8. Streamlines; 2D in-line tube bank configuration.

The streak lines for the 3D tube bank configuration for $Re = 60$ and for a $114 \times 24 \times 29$ grid are presented in Figure 6. The convergence history is presented in Figure 7. For this case, 150 cyles are required when using SIMPLEC alone, while 89 cycles are required when using the ACM-S to reduce the velocity potential equation residual to round-off error levels. Here, three grid levels with a coarsening of 2×2 were also used.

The cases outlined above were run on a HP-735 workstation. The CPU times required are summarised in Table I. These results indicate that the use of the ACM-S strategy is effective in reducing the overall CPU time required to achieve round-off error convergence within the momentum–velocity potential equations SIMPLEC cycle, and that the algorithm is also effective for problems with body-fitted stretched grids.

3.2. Turbulent flow with heat transfer results

The methodology described above has been tested on three cases that include turbulent flows with heat transfer. The test cases include tube banks and power station boiler configurations resulting in highly stretched curvilinear grids.

The first test case consists of a two-dimensional in-line tube bank configuration for $Re = 30000$. The boundary conditions assume an inlet air temperature of 400 K and a constant wall temperature of 288 K. The grid and a streamlines plot of the converged solution are presented in Figure 8. The convergence history using the standard velocity potential (SIMPLEC) algorithm and the ACM-S with dynamic relaxation for all the variables are presented in Figure 9. Three grid levels with coarsening of 2×2 were used for the ACM-S algorithm. The values of the relaxation factors when using the SIMPLEC formulation are the highest possible for numerical stability leading to fastest convergence, which correspond to 0.75 for u and v, 0.9 for the velocity potential, 0.7 for k and ϵ and 0.5 for temperature. For the ACM-S with dynamic relaxation algorithm, these values are used as the minimum to establish a

common base of comparison, while the maximum values of the relaxation coefficients are set to 0.99, 0.99, 0.99, 0.99, 0.95 and 0.99 for u, v, velocity potential, k, ϵ and temperature respectively. The choice of the highest possible relaxation factors for the SIMPLEC algorithm is aimed at comparing the 'best performance' of this algorithm with the methodology proposed in this work. In practice, relaxation factors are chosen below optimum, making the ACM-S-DT method comparatively more efficient. For the SIMPLEC algorithm, the variable with the slowest convergence is temperature, requiring approximately 600 iterations to achieve round-off error convergence.

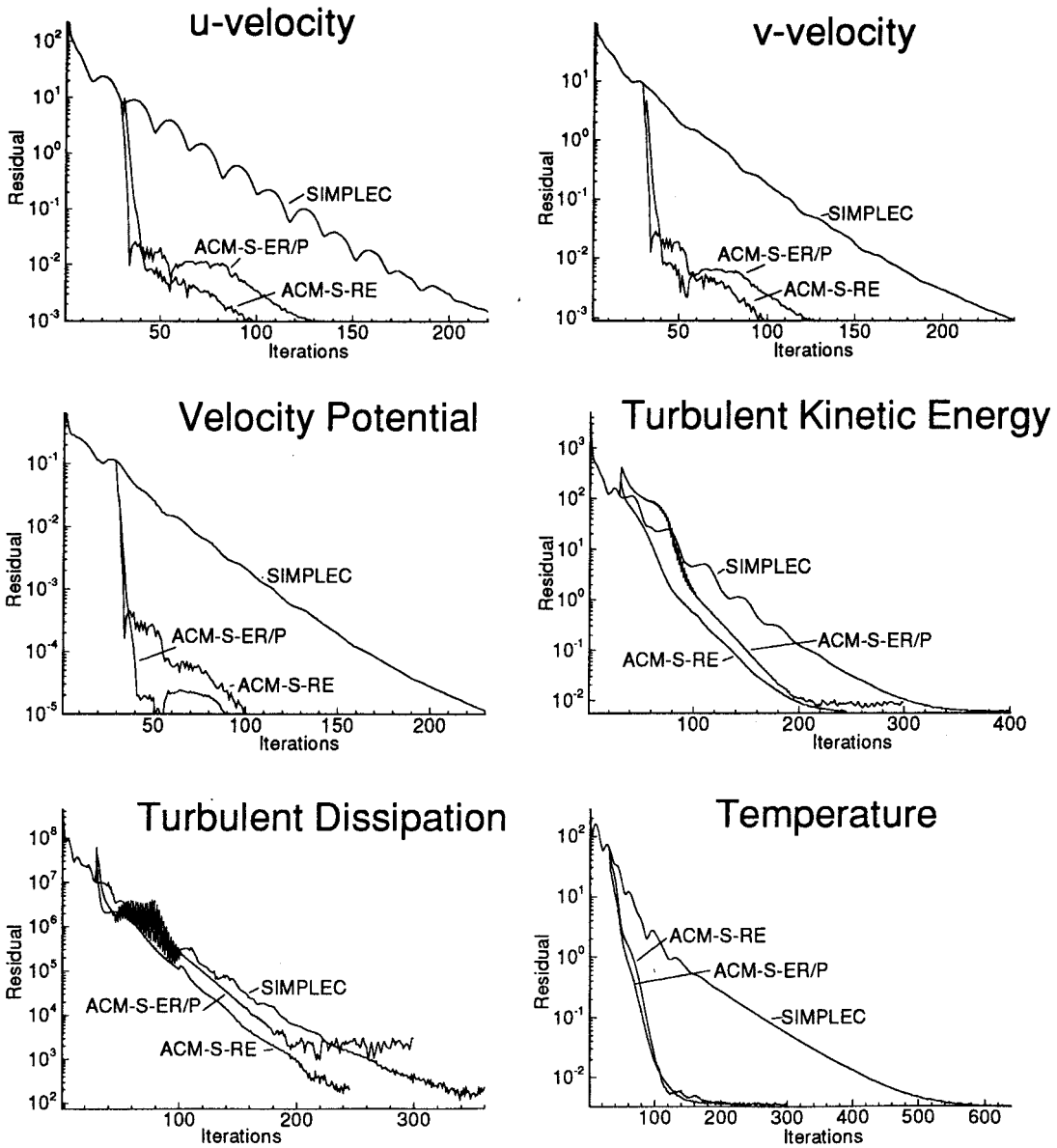

Figure 9. Convergence history.

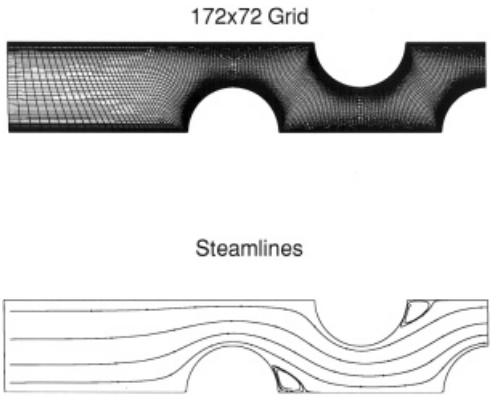

Figure 10. Streamlines; 2D staggered tube bank configuration.

The convergence of all equations is accelerated when using the ACM-S with dynamic relaxation (ACMS-DT). The ACM-S is applied after 30 iterations to take advantage of the efficiency of the SIMPLEC algorithm during the initial stages of the iterations process. During the first 30 iterations, only the dynamic tuning of the relaxation factors is applied. The u, v and velocity potential equations require between 100 and 130 cycles to converge, whether the ACM-S-RE or the ACM-S-ER/P methods are used. The residual history between iterations 30 and 40 shows typical multigrid behaviour, with a constant rate of residual reduction. However, the residual reduction rate drops for iterations larger than 40.

This behaviour may be explained because the residual reduction of the turbulence transport equations does not increase at the same rate, and the influence of k and ϵ through the eddy viscosity and turbulent diffusivity on the momentum and velocity potential equations is larger at lower residual levels. Therefore, it is numerically more efficient to switch back to the single grid algorithm in the momentum–velocity potential equations loop after the rate of convergence decreases. For the turbulence transport and temperature equations, there is also a significant increase in the convergence rate. The ACM-S-RE method performs better than the ACM-S-ER/P for k and ϵ, while for temperature both methods produce similar results. The residual for the ϵ equation presents a noisy behaviour between iterations 40 and 100, and the residual does not reach the same low level as for the other methods when using the ACM-S-ER/P method. The overall efficiency of the algorithm is constrained by the slowest equation(s) to converge (k and ϵ). Therefore, the ACM-S-RE algorithm is more efficient for this case.

The second test case consists of a two-dimensional staggered tube bank configuration for $Re = 18000$. The boundary conditions assume an inlet air temperature of 1000 K and constant wall temperature of 500 K. The grid and streamlines of the converged solution are presented in Figure 10. Three grid levels with coarsening of 2×2 were used for the ACM-S algorithm. The minimum–maximum values of the relaxation coefficients are 0.7–0.95, 0.7–0.95, 0.9–0.99, 0.6–0.9, 0.5–0.9 and 0.7–0.9 for the u, v, velocity potential, k, ϵ and temperature respectively. As for the previous case, the variable with slowest convergence using SIMPLEC is temperature, requiring approximately 2500 iterations to achieve round-off error convergence (Figure 11).

As in the previous test case, the convergence rate is accelerated when using the ACM-S-DT strategy. The u, v and velocity potential equations require approximately 400 iterations to achieve round-off error convergence when using the ACM-S-RE method, and approximately

750 iterations when using the ACM-S-ER/P method. As for the previous case, a decrease of the convergence rate is observed for these equations after 100 iterations. For the turbulence transport and temperature equations, the ACM-S-RE method leads to a higher convergence rate than the ACM-S-ER/P. For the temperature equation, the ACM-S-ER/P method does not improve the convergence rate when compared with the SIMPLEC algorithm. The poor performance of the ACM-S-ER/P method could be explained because the relative size of regions of recirculating flow is smaller than for the previous case, resulting in a narrower range of advection characteristics over the domain. Overall convergence is obtained after approximately 750 iterations when using the ACM-S-RE method.

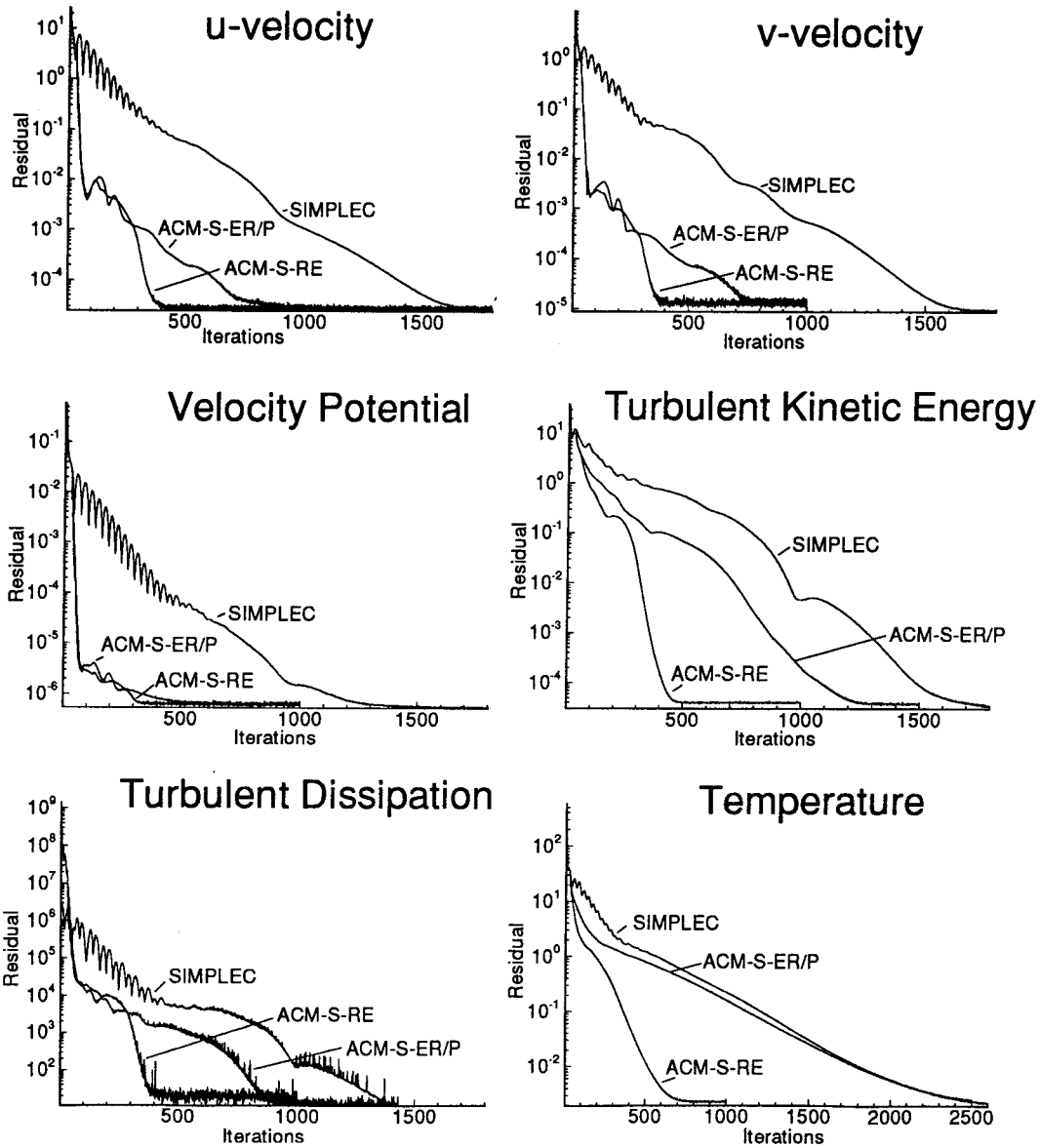

Figure 11. Convergence history.

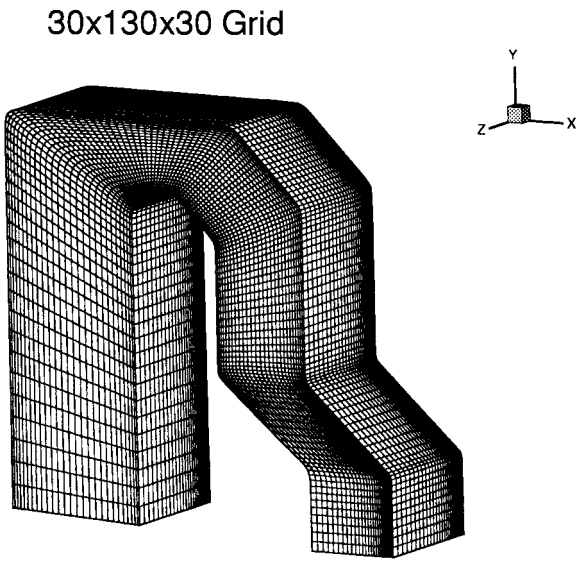

Figure 12. 3D boiler configuration: grid.

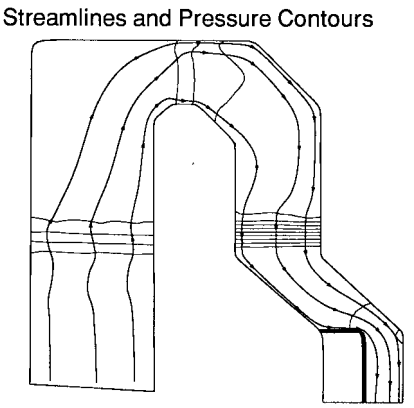

Figure 13. Streamlines; symmetry plane.

The third test case consists of a three-dimensional boiler configuration for $Re = 5.5 \times 10^6$. The boundary conditions assume an inlet air velocity of 10 m s^{-1} and an inlet temperature of 1000 K and a constant wall temperature of 373 K. In this case, a momentum and energy sources formulation is used to simulate the overall temperature and pressure drop across the tube banks [18]. The grid is presented in Figure 12. A combined streamlines–pressure contours

Table II. Performance of the ACM-S-RE algorithm

Test case	Grid size	SIMPLEC (CPU; s)	ACM-S-REBRS (CPU; s)
2D in-line tube bank	146×22	364.7	239.3
2D staggered tube bank	172×72	10 616.6	6763.3
3D boiler configuration	$30 \times 130 \times 30$	47 728.3	20 567.61

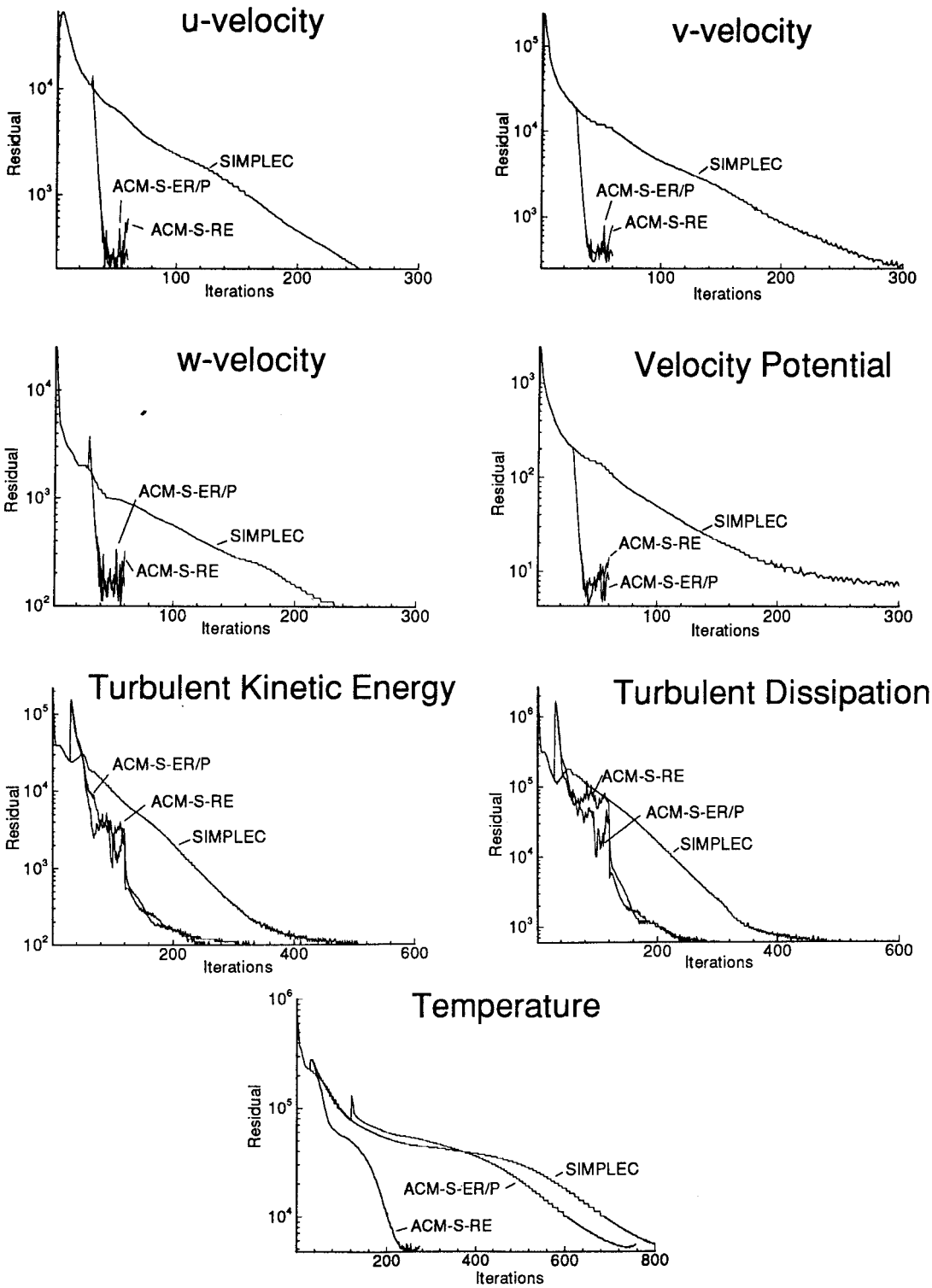

Figure 14. Convergence history.

plot for an intermediate plane ($k = 15$) is presented in Figure 13. Large pressure gradients are observed in the regions where the tube banks are located. Three grid levels with coarsening of 2×2 were used for the ACM-S algorithm. The minimum–maximum values of the relaxation coefficients are 0.6–0.95, 0.6–0.95, 0.6–0.95, 0.8–0.99, 0.6–0.9, 0.6–0.9 and 0.6–0.9 for the u, v, w, velocity potential, k, ϵ and temperature respectively. The convergence histories for all equations are presented in Figure 14. As for the previous cases, the variable with slowest convergence using SIMPLEC is temperature, requiring approximately 800 iterations to achieve round-off error convergence.

When using the ACM-S-DT strategy, the u, v, w and velocity potential equations require approximately 40 iterations to achieve round-off error convergence when using the ACM-S-RE and the ACM-SER/P methods. Therefore, the solver was switched to a single grid algorithm in the momentum–velocity potential equations loop to increase numerical efficiency. For the turbulence transport and temperature equations, both the ACM-S-RE and ACM-S-ER/P methods give similar convergence rates. However, for the temperature equation, the ACM-S-ER/P results in a marginal improvement on the convergence rate when compared with the SIMPLEC algorithm. Overall convergence is obtained after approximately 240 iterations when using the ACM-S-RE method and complex computational domains.

The test cases described above were run on a HP-K200 workstation. The CPU times required are summarised in Table II. The CPU times indicated in Table II refer to the ACM-S-RE, which was the most efficient for all the cases studied. These results indicate that the use of the ACM-S-RE strategy is effective in reducing the overall CPU time required to achieve round-off error convergence for turbulent flows with heat transfer in problems with distorted–stretched grids, and when large gradients are present.

4. CONCLUSIONS

An ACM-S-DT procedure for the discretised incompressible Navier–Stokes equations, in which the iterative nature of segregated methods is preserved on the coarse grid allowing for the same solver to be used on all grid levels, is presented. The method has been tested for a two- and three-dimensional laminar and turbulent flows with heat transfer problems using stretched curvilinear grids leading to a significant reduction of the computational cost required to achieve round-off error convergence. This methodology becomes very attractive for practical flow problems with complex geometries and a large number of grid points.

ACKNOWLEDGMENTS

This research has been carried out as part of the Department of Industry, Science and Technology (DIST) Generic Technology Project, 'Design Software for Power Utility Boilers', using equipment supplied, in part, by Pacific Power.

REFERENCES

1. P. Wesseling, *An Introduction to Multigrid Methods*, Wiley, New York, 1994.
2. C.A.J. Fletcher, *Computational Techniques for Fluid Dynamics*, Springer, Berlin, 1991.
3. A. Settari and K. Aziz, 'A generalization of the additive correction method for the iterative solution of matrix equations', *SIAM J. Numer. Anal.*, **10**, 506–521 (1973).
4. B.R. Hutchinson and G.D. Raithby, 'A multigrid method based on the additive correction strategy', *Numer. Heat Transf.*, **9**, 511–537 (1986).

5. B.R. Hutchinson, P.F. Galpin and G.D. Raithby, 'Application of the additive correction multigrid to the coupled flow equations', *Numer. Heat Transf.*, **13**, 133–147 (1988).
6. K.M. Kelkar and S.V. Patankar, 'Development of generalized block correction procedures for the solution of the discretised Navier–Stokes equations', *Comput. Phys. Commun.*, **53**, 329–336 (1989).
7. F. Zdravistch, C.A.J. Fletcher and K. Srinivas, 'Multigrid acceleration of the full SIMPLEC algorithm', *Computational Techniques and Applications: CTAC-95*, Melbourne, Australia, 1995.
8. Fluid Dynamics International, *FIDAP 7.0 Theory Manual*, 1st edn., 1993.
9. B.E. Launder and D.B. Spalding, 'The numerical computation of turbulent flows', *Comput. Methods Appl. Mech. Eng.*, **3**, 263–289 (1974).
10. C.C. Chieng and B.E. Launder, 'On the calculation of turbulent heat transport downstream of an abrupt pipe expansion', *Numer. Heat Transf.*, **3**, 189–207 (1980).
11. F. Zdravistch, C.A.J. Fletcher and M. Behnia, 'Numerical laminar and turbulent fluid flow and heat transfer predictions in tube banks', *Int. J. Numer. Methods Heat Fluid Flow*, **5**, 717–733 (1995).
12. M. Zedan and E. Schneider, 'A three-dimensional modified strongly implicit procedure for heat conduction', *AIAA J.*, **21**, 295–303 (1983).
13. C.A.J. Fletcher and J.G. Bain, 'An approximate factorisation explicit method for CFD', *Comput. Fluids*, **19**, 61–74 (1991).
14. T. Gjesdal, 'A note on the additive correction multigrid method', *Int. Commun. Heat Mass Transf.*, **23**, 293–298 (1996).
15. S.F. McCormick, *Multigrid Methods*, *Frontiers in Applied Mathematics*, SIAM, Philadelphia, PA, 1987.
16. Fluent Incorporated, *Fluent User's Guide*, Lebanon, NH, 1995.
17. V. Casulli, 'Eulerian–Lagrangian methods for the Navier–Stokes equations at high Reynolds number', *Int. J. Numer. Methods Fluids*, **8**, 1349–1360 (1988).
18. F. Zdravistch, C.A.J. Fletcher and J.A. Reizes, 'Numerical simulation of fluid flow and heat transfer over tube bank heat exchangers in power station boilers using a momentum and energy sources formulation', *Int. Symp. on Transport Phenomena (ISTP-9)*, Singapore, 1996.

