Journal of Computational Physics **165,** 566–591 (2000)
doi:10.1006/jcph.2000.6633, available online at http://www.idealibrary.com on IDEAL®

Acceleration of Multigrid Flow Computations through Dynamic Adaptation of the Smoothing Procedure

D. Drikakis,* O. P. Iliev,† and D. P. Vassileva‡

**Department of Engineering, Queen Mary, University of London, London E1 4NS, United Kingdom;*
†Institute for Industrial Mathematics (ITWM), Erwin-Schroedinger str., Building 49, D-67663 Kaiserslautern, Germany; and ‡Institute of Mathematics and Informatics, Bulgarian Academy of Science, Acad. G. Bonchev str., bl. 8, BG-1113 Sofia, Bulgaria

E-mail: *d.drikakis@qmw.ac.uk; †iliev@itwm.uni-kl.de, oleg@math.bas.bg; ‡vasileva@math.bas.bg

Received August 3, 1999; revised February 10, 2000

The paper presents the development and investigation of an adaptive-smoothing (AS) procedure in conjunction with the full multigrid–full approximation storage method. The latter has been previously implemented by the authors [1] for the incompressible Navier–Stokes equations in conjunction with the artificial-compressibility method and forms the basis for investigating the current AS approach. The principle of adaptive smoothing is to exploit the nonuniform convergence behavior of the numerical solution during the iterations to reduce the size of the computational domain and, subsequently, to reduce the total computing time. The implementation of the AS approach is investigated in conjunction with three different adaptivity criteria for two- and three-dimensional incompressible flows. Furthermore, a dynamic procedure (henceforth labeled *dynamic adaptivity*) for defining variably the AS parameters during the computation is also proposed and its performance is investigated in contrast to AS with constant parameters (henceforth labeled *static adaptivity*). Both static and dynamic adaptivity can provide similar acceleration, but the latter additionally provides more stable numerical solutions and also requires less experimentation for optimizing the performance of the method. Numerical experiments are presented for attached and separated flows around airfoils as well as for three-dimensional flow in a curved channel. For external flows the AS performs better when it is applied in all grid levels of the multigrid method, while for internal flows the best performance is achieved when AS is applied in the finest grid only. Overall, the results show that substantial acceleration of multigrid computations can be achieved by using adaptive smoothing.

Key Words: multigrid; Navier–Stokes equations; adaptivity.

0021-9991/00 $35.00

1. INTRODUCTION

Multigrid (MG) and adaptive-grid approaches are among the most powerful numerical methods for improving the efficiency of computational fluid dynamics (CFD) solvers. The multigrid method was originally developed for solving elliptic PDEs (for details see [2–6]) and, at present, is broadly used for solving different types of equations, including the Navier–Stokes equations (see, for example, [1, 7–13], and references therein, among others).

The use of the *adaptivity approach* has a long history in the numerical solution of PDEs. This general approach can be exploited in many different ways, including adaptive minimization of the computational domain, known as a "local-solution method" [14]; grid adaptation (see, e.g., [15, 16]); and the sparse-grid approach (see, e.g., [17, 18]). The aim of the aforementioned methods and techniques is to improve numerical efficiency in terms of savings on memory and CPU.

In the present work, the adaptivity is exploited at the stage of the iterative solution of the discretized problem, and this is done by performing adaptive smoothing (AS) within the global multigrid algorithm. The idea of adaptivity in conjunction with multilevel techniques was first exploited by Brandt [4, 19] and Bai and Brandt [15] for solving elliptic problems. Performing some additional smoothings near the known singularity of the solution and/or near the boundary, as well as excluding some subdomains from the relaxation sweeps, were proposed by Brandt and his co-workers as possible strategies, among other ideas. The adaptive minimization of the computational domain was also investigated by Drikakis and Tsangaris [14] in conjunction with the compressible Euler/Navier–Stokes equations for flows around airfoils. A pointwise adaptive-smoothing algorithm was also developed and theoretically investigated by Rüde [20, 21] in connection with multigrid solutions of linear elliptic equations. His algorithm was based on the Southwell method [22] for hand–solving systems of linear algebraic equations. The Southwell method is a variant of the Gauss–Seidel method, exploiting adaptive ordering of unknowns, based on the range of residuals.

The adaptive smoothing developed in the present work can be viewed as a block version of the Southwell method [22], applied to solving the nonlinear system of the Navier–Stokes equations. By adaptive smoothing we mean that the *smoother* (i.e., the single-grid flow solver in our case) acts only on an adaptively formed subset ω_s of the grid ω. In fact, this is the subset where the solution significantly changes and, therefore, the residuals have large values. The choice of the adaptivity criterion is an open question and may be problem dependent. Three adaptivity criteria are investigated in the present work in connection with the computation of external and internal flows. Moreover, we consider adaptivity criteria with fixed (constant) parameters (*static adaptivity*) as well as with dynamically defined parameters (*dynamic adaptivity*). In the former case, information about the residuals at the current iteration (or current time step) is used to reconstruct the subset ω_s, whereas according to dynamic adaptivity, information about the residuals at consecutive iterations is employed. Thus, not only the current residuals but also the convergence history is exploited in the dynamic-adaptive process. The authors are not aware of any previous work regarding the development of adaptive smoothing in conjunction with the full multigrid–full approximation storage (FMG–FAS) method and the artificial-compressibility formulation of the Navier–Stokes equations and, thus, this is the motivation for the present study.

The main objectives and contributions of the present research are: (i) to present the development and implementation of the adaptive-smoothing approach in conjunction with the nonlinear FMG–FAS method and the artificial-compressibility formulation of the

Navier–Stokes equations, (ii) to investigate various adaptivity criteria, (iii) to investigate the effects of *static* and *dynamic adaptivity* on multigrid acceleration, and (iv) to apply the above in the simulation of two- and three-dimensional external and internal flows featuring separation and secondary flow effects, respectively. The incompressible Navier–Stokes equations are discretized by a third-order characteristics-based scheme in space and by the explicit multistage Runge–Kutta method in time.

The remainder of the paper is organized as follows: Section 2 presents briefly the governing equations and characteristics-based method. The multigrid and adaptive-smoothing approaches are described in Sections 3 and 4, respectively. In Section 5 results from the investigation of the AS approach for attached and separated flows around an airfoil, as well as for three-dimensional flow in a curved channel, are presented. Finally, in Section 6 conclusions from the present study are drawn.

2. GOVERNING EQUATIONS AND SINGLE-GRID SOLUTION METHOD

The governing equations are the three-dimensional Navier–Stokes equations written in curvilinear coordinates (ξ, η, ζ) and in matrix form as

$$(JU)_t + (E_I)_\xi + (F_I)_\eta + (G_I)_\zeta = (E_V)_\xi + (F_V)_\eta + (G_V)_\zeta. \tag{1}$$

The unknown solution vector U is

$$U = (p/\beta, u, v, w)^T,$$

where p is the pressure, u, v, and w are the velocity components, and β is the artificial compressibility parameter. For steady flow problems t is a pseudo-time.

The inviscid E_I, F_I, G_I and the viscous E_V, F_V, G_V fluxes are written as

$$\begin{aligned}
E_I &= J(\tilde{E}_I\xi_x + \tilde{F}_I\xi_y + \tilde{G}_I\xi_z),\\
F_I &= J(\tilde{E}_I\eta_x + \tilde{F}_I\eta_y + \tilde{G}_I\eta_z),\\
G_I &= J(\tilde{E}_I\zeta_x + \tilde{F}_I\zeta_y + \tilde{G}_I\zeta_z),\\
E_V &= J(\tilde{E}_V\xi_x + \tilde{F}_V\xi_y + \tilde{G}_V\xi_z),\\
F_V &= J(\tilde{E}_V\eta_x + \tilde{F}_V\eta_y + \tilde{G}_V\eta_z),\\
G_V &= J(\tilde{E}_V\zeta_x + \tilde{F}_V\zeta_y + \tilde{G}_V\zeta_z),
\end{aligned}$$

where the fluxes with "tildes" denote the corresponding Cartesian fluxes:

$$\tilde{E}_I = \begin{pmatrix} u \\ u^2 + p \\ uv \\ uw \end{pmatrix}, \quad \tilde{F}_I = \begin{pmatrix} v \\ uv \\ v^2 + p \\ vw \end{pmatrix}, \quad \tilde{G}_I = \begin{pmatrix} w \\ uw \\ vw \\ w^2 + p \end{pmatrix},$$

$$\tilde{E}_V = \begin{pmatrix} 0 \\ \sigma_{xx} \\ \sigma_{xy} \\ \sigma_{xz} \end{pmatrix}, \quad \tilde{F}_V = \begin{pmatrix} 0 \\ \sigma_{yx} \\ \sigma_{yy} \\ \sigma_{yz} \end{pmatrix}, \quad \tilde{G}_V = \begin{pmatrix} 0 \\ \sigma_{zx} \\ \sigma_{zy} \\ \sigma_{zz} \end{pmatrix}.$$

The terms σ_{ij} $(i, j = x, y, z)$ are the viscous stresses and J is the Jacobian of the

transformation from Cartesian to generalized curvilinear coordinates:

$$J = x_\xi(y_\eta z_\zeta - y_\zeta z_\eta) + x_\eta(y_\zeta z_\xi - y_\xi z_\zeta) + x_\zeta(y_\xi z_\eta - y_\eta z_\xi).$$

A characteristics-based method [23] is used for discretizing the inviscid terms. According to this method, a Riemann problem in each flow direction is solved to define the primitive variables (p, u, v, w) at the cell faces of a computational volume, as functions of their values along the characteristics. The characteristic variables are calculated by a third-order upwind scheme [23]. The viscous terms are discretized by central differences and the time integration is obtained by an explicit fourth-order Runge–Kutta method with local time stepping. The time step varies between the four Runge–Kutta stages, as a function of the grid coordinates and flow velocities [1]. The above algorithm is henceforth labeled a *single-grid* method.

3. MULTIGRID ALGORITHM

To accelerate the convergence of the single-grid Navier–Stokes method, a *full multigrid–full approximation storage* method has been developed [1] and forms here the basis of the present investigation. The three-stage FMG–FAS algorithm is schematically shown in Figs. 1 and 2, while the basic steps of the third stage are as follows:

Stage III—Multigrid Sweeps on Three Grids (V-Cycles)

repeat

1. perform ν_1 presmoothings using the single-grid (SG) solver on the finest grid;
2. compute the finest grid defect, restrict it to the intermediate grid, and compute the right-hand side on the intermediate grid;
3. perform ν_1 presmoothings on the intermediate grid;
4. compute the intermediate grid defect, restrict it to the coarsest grid, and compute the right-hand side on the coarsest grid;
5. perform ν_{cg} iterations on the coarsest grid;
6. compute the correction on the coarsest grid, prolongate it to the intermediate grid, and correct the solution on the intermediate grid;
7. perform ν_2 postsmoothing iterations on the intermediate grid;
8. compute the correction on the intermediate grid, prolongate it to the finest grid, and correct the solution on the finest grid;
9. perform ν_2 postsmoothing iterations on the finest grid;

until the steady state solution on the finest grid is achieved.

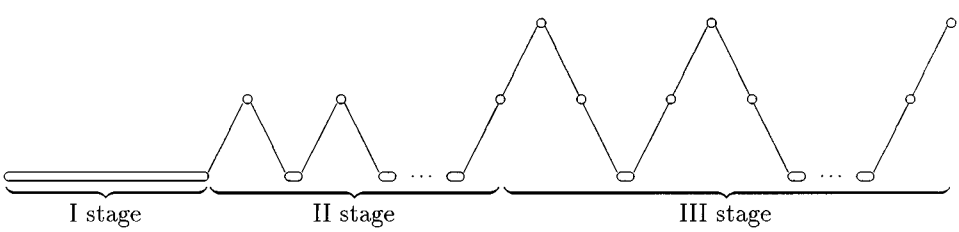

FIG. 1. Schematic of the full multigrid (FMG) for three grids: I. single grid computation on the coarsest grid; II. two-level multigrid computation on the intermediate grid; III. three-level multigrid computation on the finest grid.

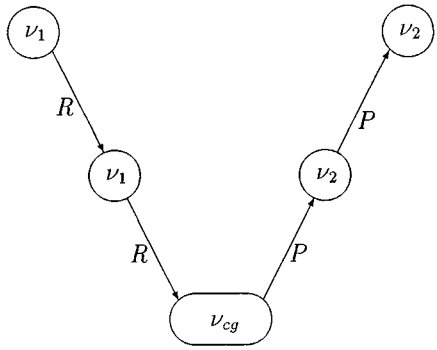

FIG. 2. Schematic of the V-cycle (R and P indicate restriction and prolongation, respectively).

According to the FMG approach, computations are initially performed on the coarsest grid to provide a good initial guess on the intermediate grid. The same procedure is repeated on the intermediate grid to provide a good initial guess on the finest grid. Thus, the FMG for three grids can be divided into three stages: two auxiliary stages, where the steady-state coarsest and intermediate grid solutions are computed, and the main stage where multigrid sweeps on three grids are performed. In the present multigrid implementation, the FMG is combined with the FAS algorithm. The latter is used to account for the nonlinearity of the problem. The FAS method was first proposed by Brandt (see [4] and discussion in [5, 6]), and it is now widely used in multigrid solutions of nonlinear problems. The single-grid algorithm, i.e., Runge–Kutta and characteristic-based methods, is used as a relaxation procedure. The restriction operator for the residuals is obtained by the volumes' weighted summation of the residuals over the fine-grid control volumes (CVs). Bilinear and trilinear prolongation operators are employed for two- and three-dimensional cases, respectively. A detailed description and performance investigations of the FMG–FAS algorithm can be found in [1].

4. ADAPTIVE-SMOOTHING ALGORITHM

The objective of the adaptive smoothing algorithm is to increase the performance of the multigrid solution and, consequently, to accelerate further the fluid flow computations. The adaptive smoothing acts only on an adaptively formed subset ω_s (*active set*) of the full grid ω. The idea is to avoid having to make too many computations in those cells of the computational domain where the solution converges very quickly, i.e., where the residuals have small values. In this way, the number of operations per iteration is reduced, thus resulting in acceleration of the computations.

The relaxation procedure $S(U, \nu, \gamma, n_s)$ for performing adaptive smoothing is described below. The notation ν stands for any of ν_1, ν_{cg}, or ν_2. The parameters γ and n_s govern the adaptivity; γ (defined later on) is used in the reconstruction of the active set $\omega_s = \omega_s(n)$ (where n is the current iteration on the corresponding grid in the current MG sweep) and determines the number of CVs belonging to the active set at the current (i.e., nth) relaxation. The parameter n_s stands for the length of the adaptivity cycle, i.e., the number of iterations for which AS is applied.

The development of an adaptive-smoothing algorithm requires the following two issues to be considered: (i) choice of an appropriate algorithm (i.e., the adaptivity criterion) for

reconstructing ω_s, based on the local convergence behavior of the iterative solver, and (ii) optimization of the *global exchange* of information between different subdomains to avoid *stagnation* (constant residuals values) or divergence of the iterative solver. These two issues are discussed below.

4.1. *Adaptivity Criterion*

Obviously, the subset ω_s must contain those CVs $P \in \omega$ for which the residuals are relatively large. One possibility is to put in ω_s only the CV in which the current residual reaches its maximum value and repeat this procedure for every iteration. Such a pointwise adaptive-smoothing algorithm has been proposed and realized for elliptic linear problems by Rüde [20, 21]. In this case the solution update in every single point where the residual reaches its maximum value would be followed by the reconstruction of ω_s and this would require extra operations, thus resulting in a significant increase of computing time. It is, therefore, more efficient to work with larger subsets ω_s, especially when complex flow problems on fine grids are to be solved.

The identification of *large residuals* can be done either with respect to the convergence criterion or with respect to the current norm of the residuals. These issues are discussed below in conjunction with a number of adaptivity criteria. According to our computational approach, a steady-state problem is solved by performing pseudo-time steps. Therefore, the norm of the current residuals of the steady-state problem is equal to the norm of the time derivative term and in our case this is the term $\|JU_t\|$. Let us denote by $\mathrm{res}(P)$ the value of the maximum component of the discrete analogue of JU_t on a CV P, and let ε be the required accuracy of the iterative solution of the steady-state problem (Eq. (1)). In other words, the convergence criterion on the finest grid should be

$$\|\mathrm{res}\|_{C(\omega)} < \varepsilon.$$

Aiming at constructing the subset ω_s in a computationally inexpensive way, we "freeze" the residuals for several time steps in those CVs where they have relatively small values. Let us denote by $\widetilde{\mathrm{res}}(P)$ the last computed residual on CV P to distinguish it from the "true" residual $\mathrm{res}(P)$. To understand the difference between the above two residuals, consider two neigboring cells, Q and P, where Q belongs to the *active set*, but not P. At the next adaptive-smoothing step, the solution is updated only for the cell Q, and, subsequently, the residual is also computed there. Because the residual corresponding to the cell P would depend on the solution in cell Q, if the entire domain was solved, the "true" residual in P has to be recomputed after the solution in Q has been updated. However, this would be computationally expensive. Therefore, in the definition of the various adaptivity criteria given below, we use the last computed residual in P, instead of the "true" one. Bearing in mind the above notation, we can write the adaptivity criteria considered here for reconstructing ω_s as

- *absolute criterion:* $\omega_s = \{P: |\widetilde{\mathrm{res}}(P)| \geq \gamma\varepsilon,\ P \in \omega\}$,
- *relative C criterion:* $\omega_s = \{P: |\widetilde{\mathrm{res}}(P)| \geq \gamma\|\widetilde{\mathrm{res}}\|_{C(\omega)},\ P \in \omega\}$,
- *relative L_2 criterion:* $\omega_s = \{P: |\widetilde{\mathrm{res}}(P)| \geq \gamma\|\widetilde{\mathrm{res}}\|_{L_2(\omega)},\ P \in \omega\}$,

where $\gamma \geq 0$. It is obvious that the subset ω_s is identical to the full grid ω for $\gamma = 0$. When varying γ, larger or smaller subsets ω_s of ω can be constructed.

During the computations γ may be either constant (static adaptivity) or variable (dynamic adaptivity), i.e., $\gamma = \gamma(n)$. The use of variable γ ensures numerical robustness and this is explained below. The value of γ determines the number of CVs involved in the active set ω_s. If the number of CVs reduces very quickly, the iterative solver may diverge. However, if the number of CVs reduces very slowly, then the acceleration of the convergence may not be significant. Therefore, the adaptive smoothing procedure needs to have some current information about the convergence behavior of the solver to decide how fast the number of CVs in ω_s should be reduced. In the case of variable γ we use the rule

$$\gamma = \begin{cases} \gamma_{\max}, & q \le 1, \\ \gamma_{\max} + \frac{q-1}{q_{\max}-1}(\gamma_{\min} - \gamma_{\max}), & 1 < q < q_{\max}, \\ \gamma_{\min}, & q \ge q_{\max}, \end{cases}$$

where $\gamma_{\min}$, $\gamma_{\max}$, and $q_{\max} \ge 1$ are given parameters. The notations used here are

$$q = \frac{\widetilde{\mathrm{res}}^n_{\max}}{\widetilde{\mathrm{res}}^*_{\max}}, \qquad \widetilde{\mathrm{res}}^n_{\max} = \max_\Omega\{\widetilde{\mathrm{res}}(P)\}, \tag{2}$$

with n, $1 \le n \le \nu$, being the current iteration on the corresponding grid in the current MG sweep, and

$$\widetilde{\mathrm{res}}^*_{\max} = \begin{cases} \widetilde{\mathrm{res}}^{n-1}_{\max}, & \widetilde{\mathrm{res}}^n_{\max} < \widetilde{\mathrm{res}}^{n-1}_{\max}, \\ \min\limits_{i<n}\left\{\widetilde{\mathrm{res}}^i_{\max}\right\}, & \widetilde{\mathrm{res}}^n_{\max} \ge \widetilde{\mathrm{res}}^{n-1}_{\max}. \end{cases}$$

As can be seen from the above formulae, the dynamic adaptivity procedure is based on monitoring the rate of convergence q. The overall procedure as described above encompasses information both regarding the absolute value of the residuals as well as regarding the rate of convergence of the iterative procedure. Through the use of the parameter γ we pursue the implementation of adaptive smoothing in the regions with large residuals, while through q we account for the rate of convergence in order to dynamically adapt the values of γ to the convergence behavior of the iterative process. Certainly, the proposed adaptivity procedure is one of the possible ways to establish numerically dependence of adaptivity on the decrease of residuals. At present, since the idea of adaptive smoothing in conjunction with a nonlinear multigrid and the Navier–Stokes equations is rather new, we are mainly interested in examining whether the approach works in principle and leave its further optimization to be an issue of future research.

The case $q_{\max} = 1$ corresponds to the change of γ value from $\gamma_{\max}$ to $\gamma_{\min}$, as long as the norm of the residual increases. In this case only two values of γ are used. However, for $q_{\max} > 1$, we allow γ to also have some intermediate values. As a result, γ decreases gradually if a moderate increase of the residuals occurs. Using a variable γ, divergence of the solution or slow convergence rates can be prevented. In the above procedure the residuals are computed during the Runge–Kutta iterations and, thus, no additional operations for implementing the adaptive smoothing are required.

4.2. *Global exchange*

Global exchange is realized in two ways: (i) through the solution of the equations on the entire coarsest grid and (ii) by performing a complete smoothing after every successive

($n_s - 1$) adaptive smoothings. Complete smoothing means that the equations are solved in all CVs of ω. The complete smoothing is required for computing the "true" residuals res(P) in all CVs, as well as for propagating more accurate information between different subregions. Because the "true" residuals res(P) are not computed in all CVs at every iteration, the convergence is checked only when the corresponding smoothing is performed in all CVs of the finest grid.

Using all the aforementioned definitions, we can describe the procedure schematically as

$\omega_s = \omega$
compute res^0_{max}
for $n = 1$ *to* ν *do*
  for all $P \in \omega_s$ *perform a smoothing with Runge–Kutta time stepping procedure*
  compute $\widetilde{res}^n_{max}$
  if $(\omega_s \not\equiv \omega)$ *then*
    $i_s = i_s + 1$
  else
    if $(res^n_{max} < \varepsilon)$ *exit*
    $i_s = 0$
  end if
  if $(i_s = n_s - 1$ *or* $n = \nu - 1)$ *then*
    $\omega_s = \omega$
  else
    reconstruct ω_s
    if $(\omega_s = \emptyset)$ $\omega_s = \omega$
  end if
end do

where i_s is the current adaptive smoothing iteration.

5. RESULTS

The performance of the adaptive-smoothing multigrid (AS–MG) algorithm was investigated (i) for attached and separated flows around the NACA 0012 airfoil corresponding to angles of incidence $a = 0^\circ$ and $a = 10^\circ$, respectively, and Reynolds number Re = 1000 and (ii) for the flow in a three-dimensional curved channel at Re = 790.

A three-grid AS–MG algorithm was employed in all computations. The efficiency of all algorithms employed here is measured in *work units*. One *work unit* is the computational work required for one iteration on the finest grid with all grid points involved in the computation, i.e., the work performed by the single-grid solver to complete one Runge–Kutta time step (four Runge–Kutta iterations) on the finest grid ω. In the results presented below, the reported work units also account for the operations performed on the coarser grids.

Different variants of the adaptive-smoothing algorithm were implemented. We denote by AS the variant in which the adaptive smoothing is applied in all three grids; AS_1 stands for the variant where adaptive smoothing is applied in the intermediate and finest grids, and AS_2 stands for the variant where adaptive smoothing is applied in the finest grid only. For comparison purposes, computations were also carried out using the original FMG–FAS method [1] (henceforth labeled MG). The following notations are also used in the results tables: ν_2 stands for the number of postsmoothings performed on the intermediate

or finest grids and ν_{cg} stands for the number of relaxations performed on the coarse grid. Following the conclusions of a previous investigation [1], we performed no presmoothings. Furthermore, in some cases comparisons of the MG acceleration with the corresponding mesh-sequencing (MS) solution are also presented. In the MS case, no multigrid or adaptive smoothing is used, but simply the equations are first solved on the coarsest and intermediate grids to provide a better initial guess, via interpolation, onto the finest grid.

5.1. *Attached and Separated Flows around the NACA 0012 Airfoil*

Both for the attached and separated flows corresponding to $a = 0^\circ$ and 10°, respectively, the employed finest grid has 288×72 grid points and the convergence accuracy was $\varepsilon = 10^{-5}$. This grid size was found to provide grid-independent solutions regarding the flowfield results. An enlargement of the grid around the airfoil is shown in Fig. 3. The u-velocity contours for the $a = 0^\circ$ and 10° cases are shown in Figs. 4 and 5, respectively. In these figures comparison of the results between MG and various AS–MG cases, using different adaptivity criteria, shows that the solution is the same irrespective of the criterion employed.

5.1.1. MG performance without AS. Initially, computations were performed using the MG algorithm without AS. The performance results for the attached ($a = 0^\circ$) and separated ($a = 10^\circ$) flows are given in Tables I and II, respectively. As can be seen from these tables, the efficiency of the multigrid (MG) against the mesh-sequencing (MS) method depends mainly

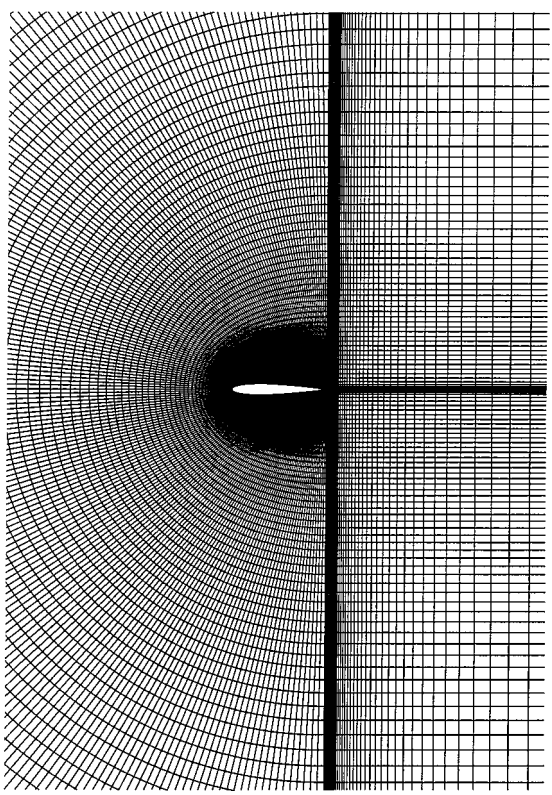

FIG. 3. View of the 288×72 computational grid around the NACA 0012 airfoil.

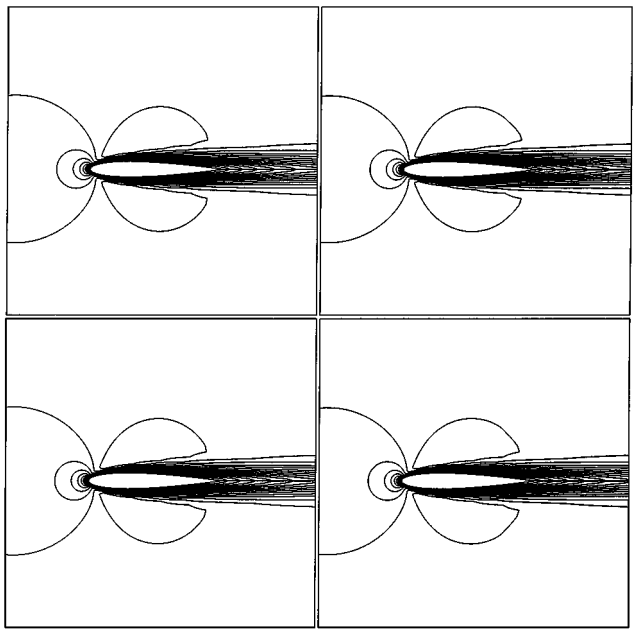

FIG. 4. Comparison of the results (u-velocity contours) between MG (top left) and AS–MG cases for the attached flow ($a = 0^\circ$) around the NACA 0012 airfoil, using different adaptivity criteria: absolute (top right), relative C (bottom left), and relative L_2 criterion (bottom right).

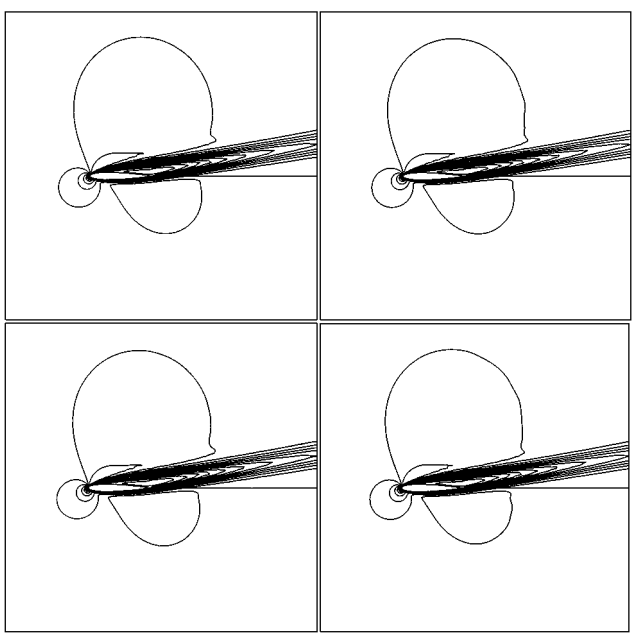

FIG. 5. Comparison of the results (u-velocity contours) between MG (top left) and AS–MG cases for the separated flow ($a = 10^\circ$) around the NACA 0012 airfoil, using different adaptivity criteria: absolute (top right), relative C (bottom left), and relative L_2 criterion (bottom right).

TABLE I
Effects of the Coarse-Grid (ν_{cg}) and Post-Relaxation (ν_2) Iterations on the MG Sweeps and Work Units for the Airfoil Case at $a = 0°$

Method	ν_{cg}	ν_2	MG sweeps	Work units	Acceleration
MS				8284	1.00
MG	5	5	428	2807	2.95
MG	11	11	195	2805	2.95
MG	21	21	101	2766	2.99
MG	81	21	101	3143	2.64

on the relation between ν_{cg} and ν_2. It is also very interesting that the MG method provides more significant acceleration as the flow becomes more complex (Table II; $a = 10°$). The distribution of residuals at the end of certain MG sweeps are shown in Figs. 6 and 7 for $a = 0°$ and $a = 10°$, respectively. As can be seen, in both cases there is a strong nonuniform behavior of the numerical convergence, resulting in large residuals in some regions and small residuals in other regions. Moreover, both for $a = 0°$ and $a = 10°$ the residuals are relatively small with respect to the maximum ones, in a significant part of the computational domain, and therefore the adaptive smoothing is particularly suitable in this case.

5.1.2. AS–MG for the case $a = 0°$ (attached flow). Results for this case using the AS–MG algorithm are shown in Tables III, IV, and V for the absolute, relative C, and relative L_2 criterion, respectively. We should also mention that in all numerical experiments discussed below the MG parameters have been fixed as $\nu_{cg} = 81$ and $\nu_2 = 21$, instead of the optimum values for these parameters being selected according to the best MG acceleration as reported in Tables I and II. The rationale behind this is twofold: First, we are interested in examining whether and how the AS–MG performance varies from one flow case to another. The latter can only be done if the MG parameters are the same for all cases. Second, the effects of AS on the MG method are independent of the values of ν_{cg} and ν_2. Therefore, bearing in mind the best acceleration the MG method can provide in contrast to the MS solution (Tables I and II), we are trying to examine what acceleration the AS can provide in addition to the MG acceleration.

As can be seen from Tables III and IV, the adaptive smoothing approach accelerates the multigrid computations by a factor of three for a broad range of γ values. It should be pointed out that wherever in the tables the γ_{min} and γ_{max} are different, the case corresponds to a variable γ (dynamic adaptivity). The relative C criterion provides the best acceleration

TABLE II
Effects of the Coarse-Grid (ν_{cg}) and Post-Relaxation (ν_2) Iterations on the MG Sweeps and Work Units for the Airfoil Case at $a = 10°$

Method	ν_{cg}	ν_2	MG sweeps	Work units	Acceleration
MS				32767	1.00
MG	21	21	474	13045	2.51
MG	81	21	228	7119	4.60
MG	41	11	442	7210	4.54
MG	321	21	115	5306	6.18

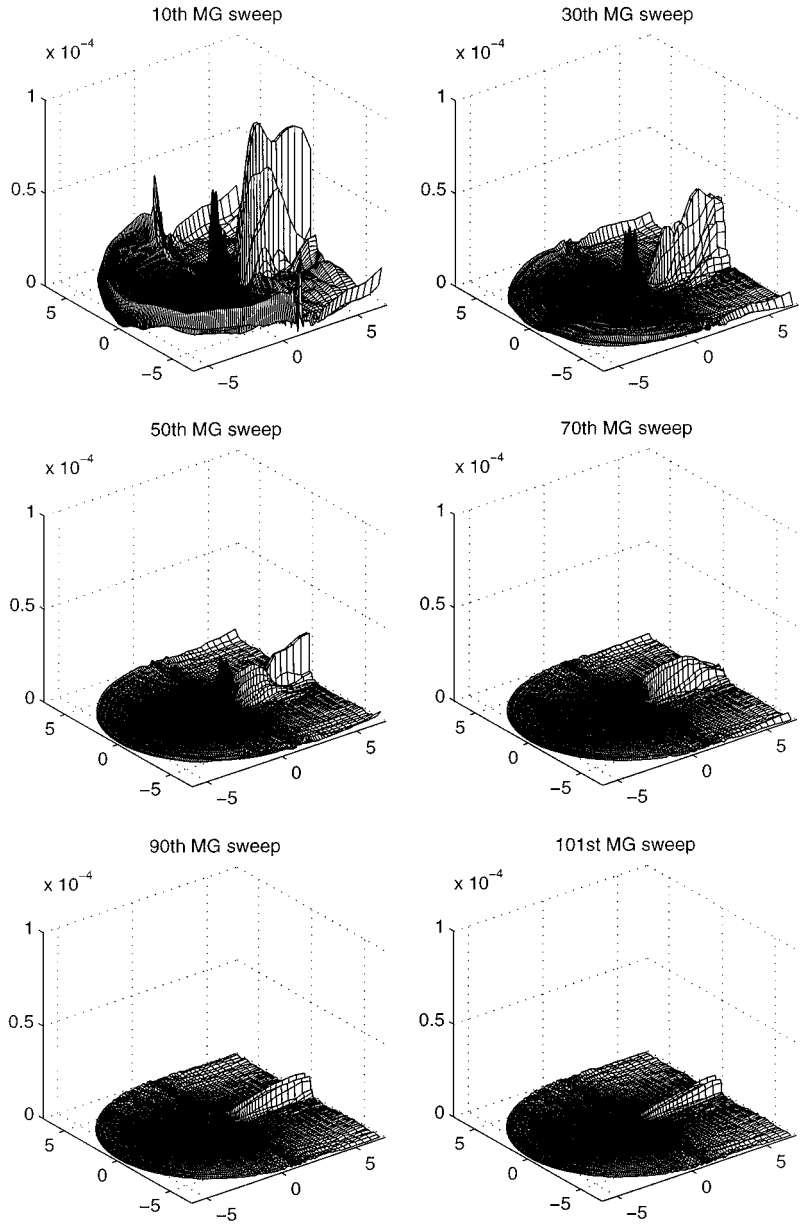

FIG. 6. Distribution of MG residuals at the end of certain MG sweeps (attached flow case where $a = 0^\circ$).

in this case. In general, it is not known in advance what the optimal value of γ is. The value of $\gamma = 1$ works well in conjunction with both the absolute and relative L_2 criteria but not for large values for n_s (see the second row in Table III as well as the second and third rows in Table V), while it is not clear what the optimum γ value for the relative C criterion is. It is also seen from these tables that large values of n_s (i.e., large length of the adaptivity cycle) can lead to divergence of the numerical solution (see the sixth row in Table III as well as the seventh and ninth rows in Table V). When the active set is small (i.e., large values for γ are used) less computational effort per smoothing is spent. However, if the active set is

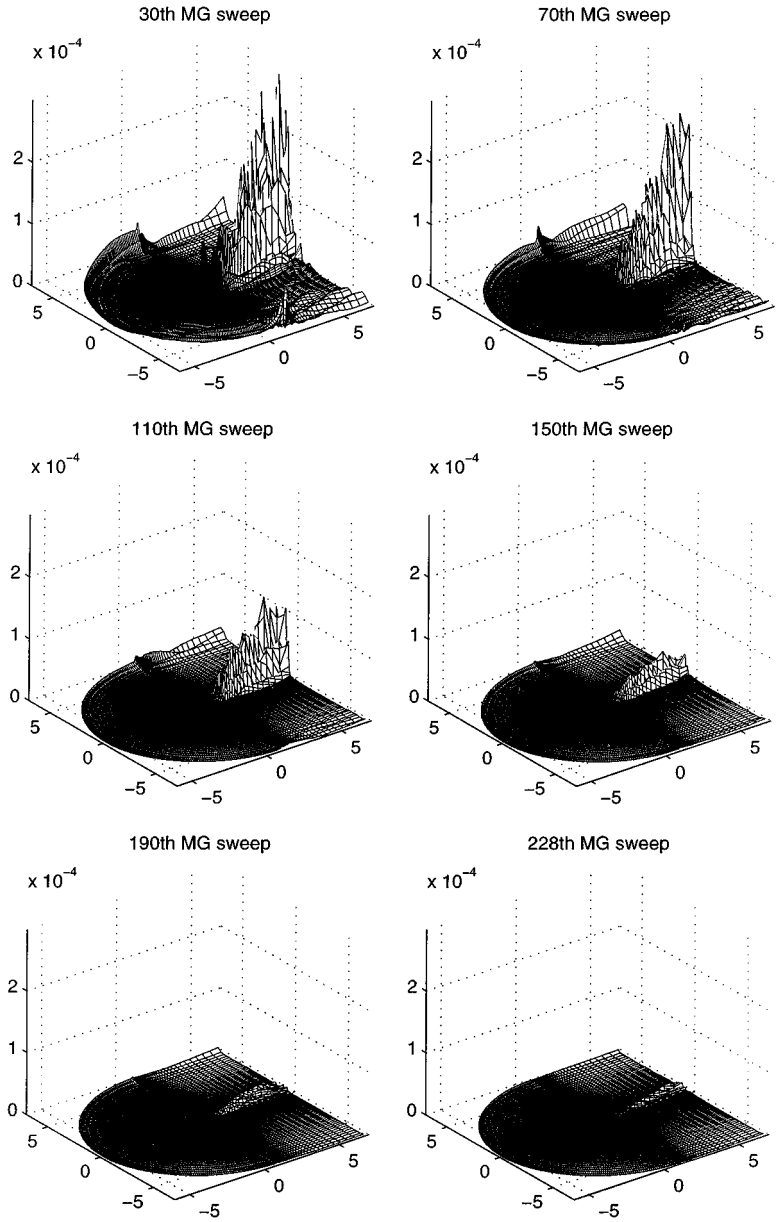

FIG. 7. Distribution of MG residuals at the end of certain MG sweeps (separated flow case with $a = 10^{\circ}$).

too small, then the number of MG sweeps usually increases and, additionally, divergence of the solution may also occur.

Comparing static and dynamic adaptivity, we find that the latter is more robust than the former in computations with slow convergence rates. However, for $q_{\max} = 1$ (see Eq. (2)) more operations are performed and the acceleration of computations due to AS essentially decreases. That is why the case with a tolerance $q_{\max} = 1.1$ has also been tested. As can be seen from the tables, in this case the acceleration is almost the same, as for the case with constant $\gamma = \gamma_{\max}$, but the case with variable γ is still more robust, especially for the

TABLE III

Results for the NACA 0012 Flow Case at $a = 0°$ Using the Absolute Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					101	3143	1.00
AS–MG	1	1		10	119	994	3.16
AS–MG	0.8	0.8		10	120	1072	2.93
AS–MG	0	1	1	10	101	1550	2.03
AS–MG	0	1	1.1	10	108	1002	3.14
AS–MG	1	1		20		no conv.	
AS–MG	0	1	1	20	103	1509	2.08
AS–MG	0	1	1.1	20	124	991	3.17

Note. γ_{min}, γ_{max}, and $q_{max} \geq 1$ are used according to Section 4.1 to define the dynamic-adaptive criterion. If $\gamma_{min} = \gamma_{max}$ then static-adaptivity is used. n_s denotes the number of adaptive smoothing iterations. The last column shows the acceleration factor of the MG method resulting from to the implementation of the adaptive-smoothing approach.

TABLE IV

Results for the NACA 0012 Flow Case at $a = 0°$ Using the Relative C Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					101	3143	1.00
AS–MG	0.1	0.1		10	93	800	3.93
AS–MG	0.2	0.2		10	98	634	4.96
AS–MG	0	0.2	1	10	105	1315	2.39
AS–MG	0	0.2	1.1	10	99	697	4.51
AS–MG	0.4	0.4		10	no conv.		
AS–MG	0	0.4	1.	10	116	1516	2.07
AS–MG	0	0.4	1.1	10	116	754	4.17
AS–MG	0	0.8	1.	10	141	2200	1.43
AS–MG	0	0.8	1.1	10	174	1173	2.68

TABLE V

Results for the NACA 0012 Flow Case at $a = 0°$ Using the Relative L_2 Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					101	3143	1.00
AS–MG	1	1		5	99	1015	3.10
AS–MG	1	1		10	102	823	3.82
AS–MG	0.8	0.8		10	108	944	3.23
AS–MG	0	1	1	10	101	1355	2.32
AS–MG	0	1	1.1	10	110	984	3.19
AS–MG	1	1		20	no conv.		
AS–MG	0	1	1	20	100	1282	2.45
AS–MG	0	1	1.1	20	no conv.		

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

relative C criterion, when it is not known in advance what value of γ should be prescribed. For the present combination of MG parameters ($\nu_{cg} = 81$, $\nu_2 = 21$) the overall acceleration achieved by the AS–MG against the MS solution, using the relative C criterion with constant γ, was about a factor of 13.09. The corresponding acceleration using variable γ is about a factor of 11.91.

An essential characteristic of the AS–MG algorithm is the dynamic variation of the number of CVs involved in the active set. Figure 8 shows the variation of the number of

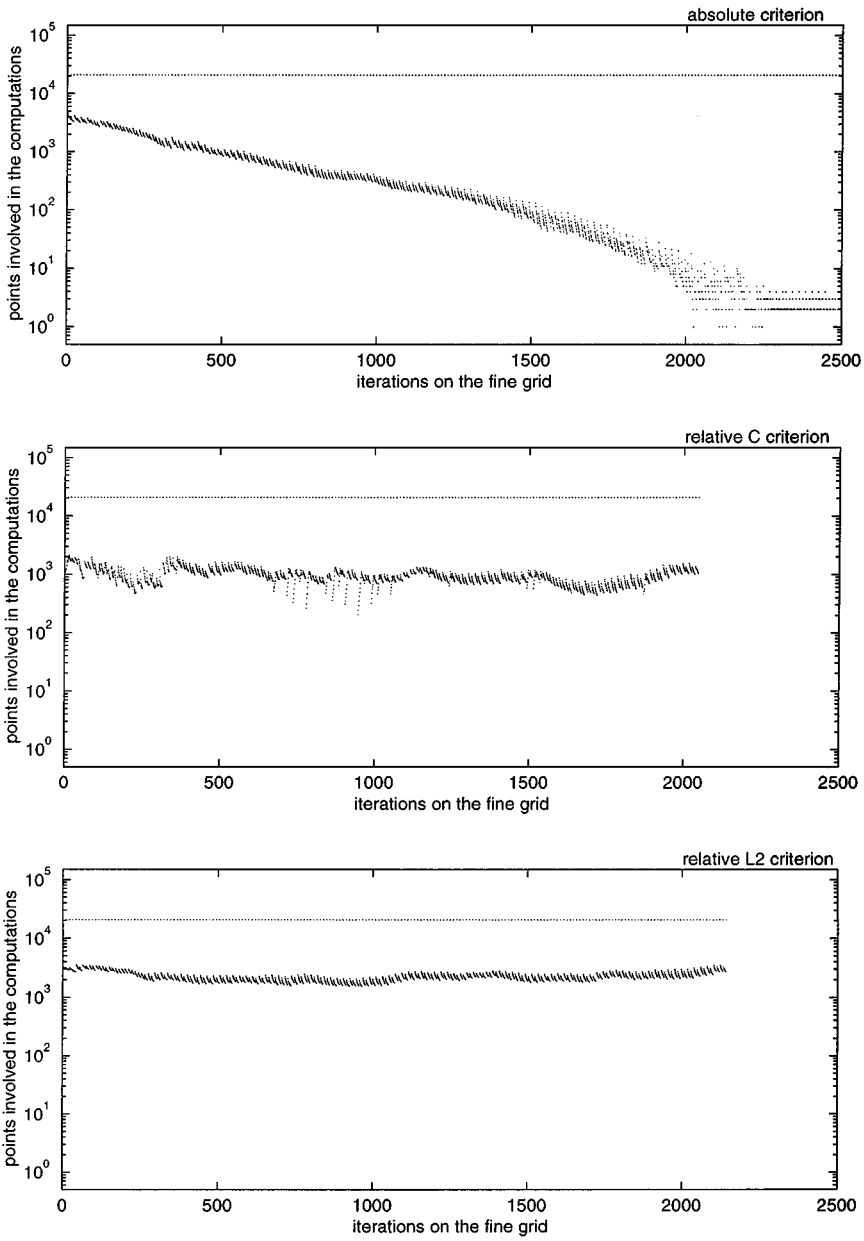

FIG. 8. Variation of the number of CVs involved in the AS–MG computation on the finest grid using different adaptivity criteria (attached flow case where $a = 0^\circ$).

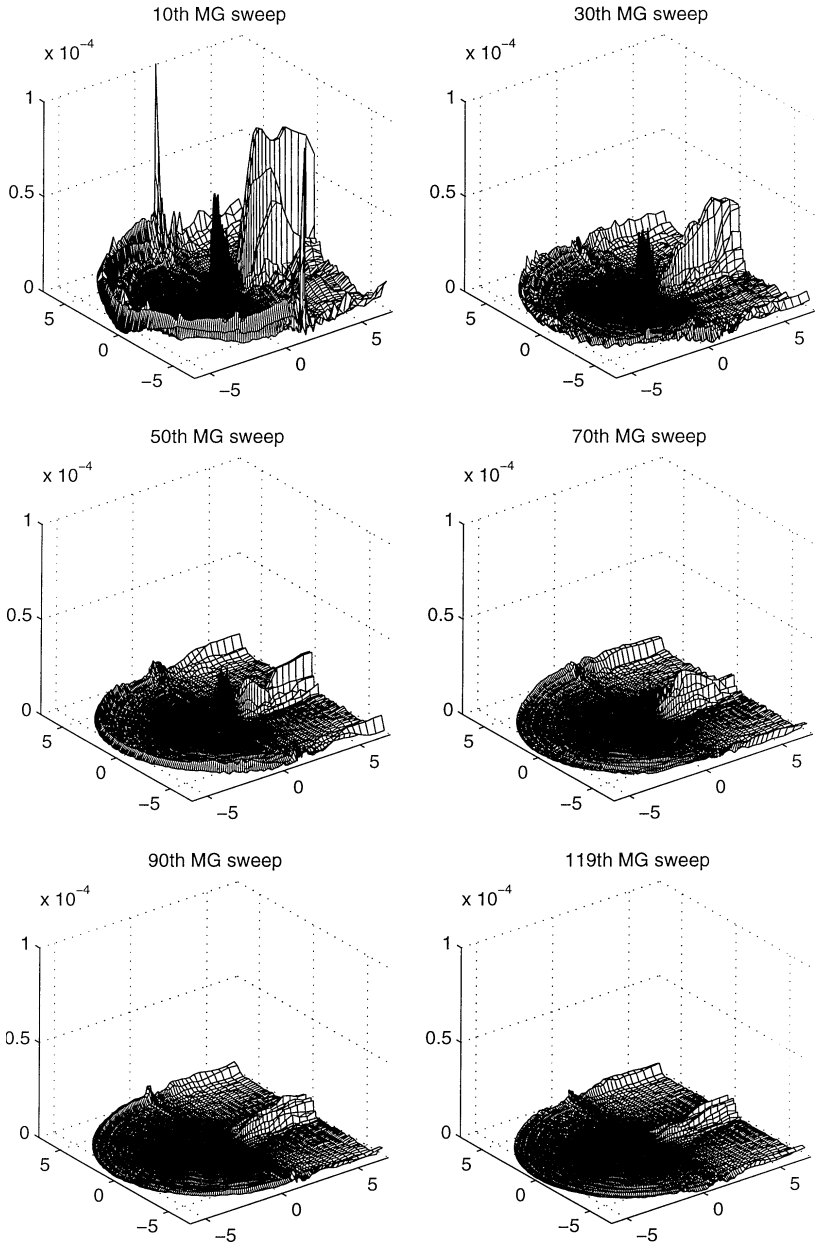

FIG. 9. Distribution of AS–MG residuals at the end of certain MG sweeps (attached flow case where $a = 0^\circ$; absolute criterion, $\gamma = 1$).

CVs, on the finest grid only, during iterations, for the absolute (constant $\gamma = 1$), relative C (constant $\gamma = 0.2$), and relative L_2 (constant $\gamma = 1$) criteria, respectively. The above values of γ correspond to the best acceleration obtained by using the three aforementioned convergence criteria. The dots in these figures correspond to the number of CVs during the postsmoothing iterations of each multigrid sweep. The x-axis in Fig. 8 includes the iterations performed on the complete grid (i.e., complete smoothings), as well as the iterations performed on the active set only (i.e., adaptive smoothings). In Fig. 8 the number

of CVs corresponding to a complete smoothing is represented by a horizontal line at the value of 20,736 CVs. Although the absolute criterion gradually leads to a continuous reduction of the CVs involved in the computation, the acceleration is less than the corresponding one obtained by the relative C criterion as well as by the relative L_2 criterion. This can be explained by the fact that for the cases plotted in Fig. 8 the absolute criterion requires more MG sweeps than the other two criteria. Several numerical experiments performed during the present study showed that, in general, the continuous reduction of the active set does not necessarily imply greater acceleration. The best performance is achieved through a balance of the size of the active set (i.e., the number of CVs) and the number of MG sweeps.

In Fig. 9, the distribution of the residuals for the AS–MG is shown. Comparing Figs. 6 and 9, one can see that the adaptive smoothing provides almost the same distribution of residuals, however, at a significantly reduced computational cost. One can notice that in certain regions of the domain AS–MG produces larger residuals than MG, but these are either very small regions in which the numerical errors die out very quickly or regions with very small residual values which have very little effect on the overall convergence.

5.1.3. AS–MG for the case $a = 10^\circ$ (separated flow). Results for the flow around the NACA 0012 at ten degrees angle of incidence are presented in Tables VI, VII, and VIII for the three convergence criteria, respectively. For $a = 10^\circ$ a large recirculation region over the airfoil occurs, which also extends downstream of the trailing edge in the wake (Fig. 10). The u-velocity contours for AS–MG and MG cases are shown in Fig. 5 and, similar to the $a = 0^\circ$ case, the AS does not affect the flow results.

The residuals distributions at different MG sweeps are shown in Fig. 11 for computations performed using the absolute criterion. Similar to the case $a = 0^\circ$, one can notice the nonuniform convergence behavior according to which the residuals decrease faster in certain regions of the flowfield than in the rest of the domain. The variation of the number of CVs belonging to the active set of the finest grid is shown in Fig. 12 for the same values of γ

TABLE VI
Results for the NACA 0012 Flow Case at $a = 10^\circ$ Using the Absolute Criterion

Method	$\gamma_{\min}$	$\gamma_{\max}$	$q_{\max}$	n_s	MG sweeps	Work units	Acceleration
MG					228	7119	1.00
AS–MG	1	1		5	203	2584	2.76
AS–MG	1	1		10	202	2161	3.29
AS–MG	0.8	0.8		10	195	2191	3.25
AS–MG	0.	1	1	10	201	3291	2.16
AS–MG	0.	1	1.1	10	201	2190	3.25
AS_1–MG	1	1		10	209	2400	2.97
AS_2–MG	1	1		10	202	2898	2.46
AS_2–MG	1	1		20	no conv.		
AS–MG	0.	1	1	20	205	3161	2.25
AS–MG	0.	1	1.1	20	209	2041	3.49

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

TABLE VII
Results for the NACA 0012 Flow Case at a = 10° Using the Relative C Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					228	7119	1.00
AS–MG	0.2	0.2		5	352	2921	2.44
AS–MG	0.4	0.4		5	515	3946	1.80
AS–MG	0	0.4	1.1	5	476	3734	1.91
AS–MG	0.1	0.1		10	358	2867	2.48
AS–MG	0.2	0.2		10	316	1911	3.73
AS–MG	0	0.2	1	10	352	4308	1.65
AS–MG	0	0.2	1.1	10	369	2205	3.23
AS–MG	0.4	0.4		10	no conv.		
AS–MG	0	0.4	1	10	568	6902	1.03
AS–MG	0	0.4	1.1	10	562	2981	2.39

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

and n_s, as for the flow case $a = 0°$. The plots in Fig. 12 correspond to the results shown in the third, sixth, and third rows of Tables VI, VII, and VIII, respectively. The number of iterations on the finest grid using the absolute, relative C, and relative L_2 criteria can be found if one multiplies the corresponding number of MG sweeps by the number of postsmoothing iterations on the finest grid (for the present case $\nu_2 = 21$). According to the above, 4242, 6636, and 6153 iterations on the finest grid are required using the absolute, relative C, and relative L_2 criteria, respectively. Although by using the absolute criterion less iterations on the finest grid are performed and, additionally, the number of CVs on

TABLE VIII
Results for the NACA 0012 Flow Case at a = 10° Using the Relative L_2 Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					228	7119	1.00
AS–MG	1	1		5	333	3307	2.15
AS–MG	1	1		10	293	2219	3.21
AS–MG	0.8	0.8		10	263	2135	3.33
AS–MG	0.6	0.6		10	230	2224	3.20
AS–MG	0	1	1	10	338	4456	1.60
AS–MG	0	1	1.1	10	287	2282	3.12
AS_1–MG	1	1		10	267	2945	2.42
AS_2–MG	1	1		10	242	3518	2.02
AS_2–MG	1	1		20	no conv.		
AS_2–MG	0	1	1	20	236	4505	1.58
AS_2–MG	0	1	1.1	20	261	3660	1.95

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

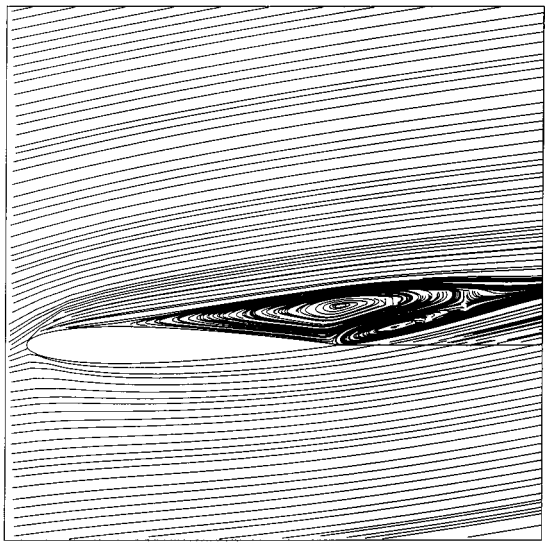

FIG. 10. Streamlines around the NACA 0012 airfoil for the flow case Re = 1000, $a = 10^\circ$.

the finest grid rapidly decreases during the iterations, the overall work units required are more than those needed by the relative C criterion (see the corresponding lines in Tables VI and VII). This is partly due to the fact that for about half of the iterations the number of CVs using the absolute criterion is larger than the corresponding number when the relative C criterion is employed. Moreover, when using the absolute criterion a larger number of CVs on the intermediate and coarsest grids are usually involved (these CVs are not shown in Fig. 12) compared to the cases of relative C or relative L_2 criterion. As a result of the above, the overall work units required are finally less in the case of the relative C criterion.

As can be seen from Tables VI, VII, and VIII, the AS approach provides additional acceleration on top of MG acceleration, by a factor of about three for a broad range of γ values. Although greater acceleration is achieved for constant γ (see the sixth row in Table VII), these results are very close to the best ones obtained using variable γ (see the last row in Table VI and the eighth row in Table VII) and, additionaly, the variable γ has been found, in general, to provide more stable solutions. The value of $\gamma = 1$ works well in conjunction with both the absolute and the relative L_2 criteria, but it is not clear what the optimal γ value in the case of the relative C criterion is.

Finally, by comparing the performance of AS, AS_1, and AS_2 (Tables VI and VIII) one can see that, for this case at least, it is more efficient to perform adaptive smoothing in all three grids. Additional numerical experiments (not presented here) verified this conclusion for the attached flow case ($a = 0^\circ$), as well.

5.2. *Three-Dimensional Flow in a Curved Channel*

The final flow case is the three-dimensional flow in a 90° bend (Fig. 13) at Re = 790. The finest grid has 95 nodes in the streamwise and 112×56 points in the transverse plane (i.e., a total of 595,840 grid points) Because of symmetry, computations could be carried out using a half section of the channel in the z-direction. As inflow conditions, the corresponding fully

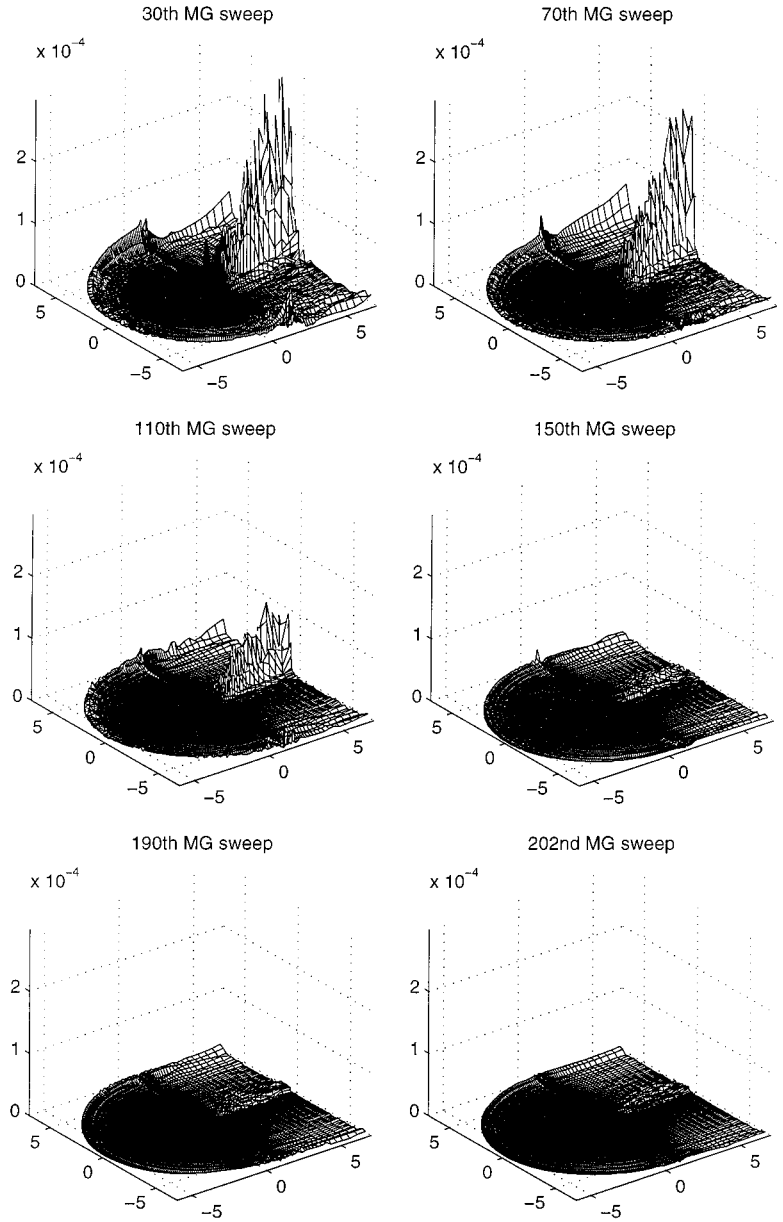

FIG. 11. Distribution of AS–MG residuals at the end of certain MG sweeps (separated flow case with $a = 10^{\circ}$; absolute criterion, $\gamma = 1$).

developed flow in a straight channel at $\mathrm{Re} = 790$ was imposed at the inlet. The bend has a large enough turning angle and a small enough mean radius to generate severe distortion and significant secondary flow. The formation of the secondary flow at $\theta = 90^{\circ}$ is shown in Fig. 14.

Detailed comparisons between computations and experiment, as well as investigation of the MG performance against the single-grid and mesh-sequencing solutions, can be found in [1]. In the present work, the investigation focuses on the results obtained by AS–MG. Like

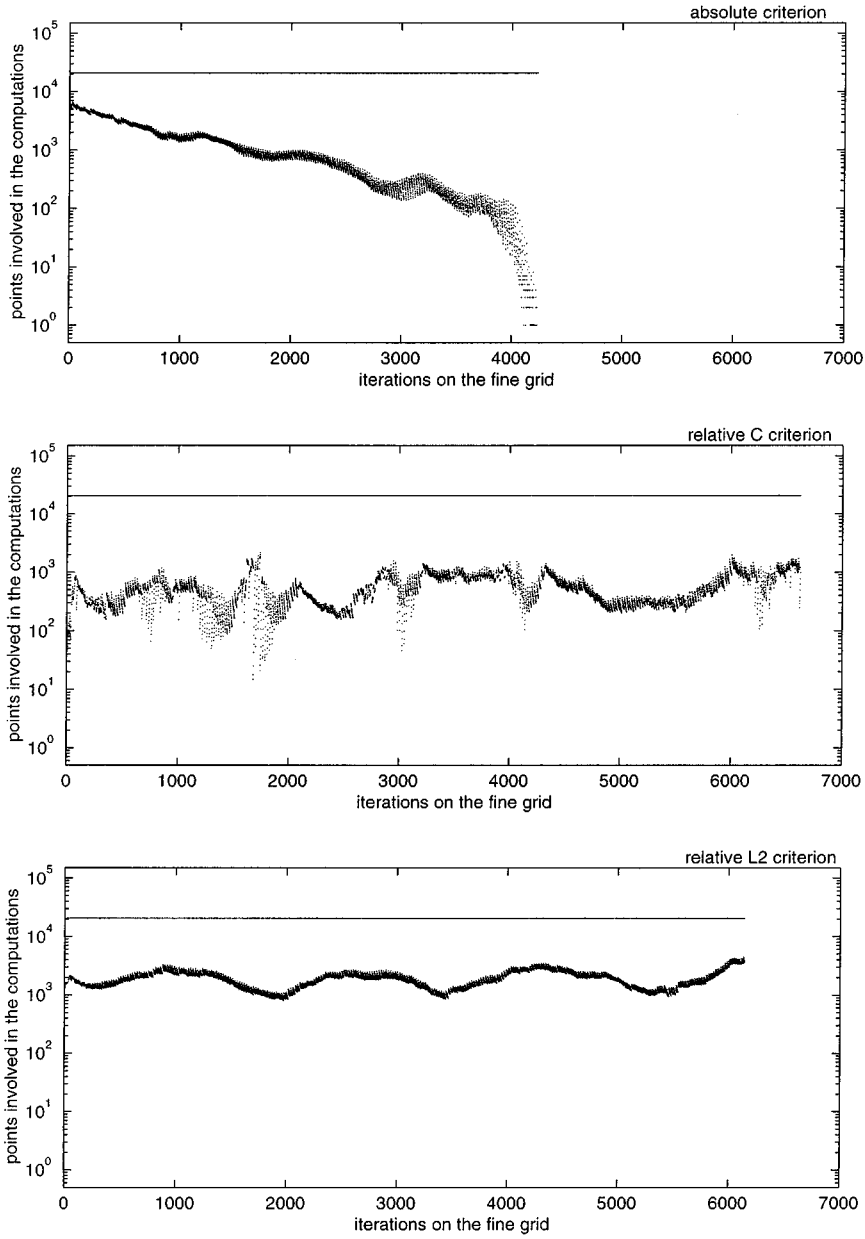

FIG. 12. Variation of the number of CVs involved in the AS–MG computation on the finest grid using different adaptivity criteria (separated flow case with $a = 10^{\circ}$).

the airfoil cases, the computations were performed using $\nu_{cg} = 81$ and $\nu_2 = 21$. Results from the numerical experiments using the absolute, relative C, and relative L_2 criteria are given in Tables IX, X, and XI, respectively. The results show that the acceleration achieved by the adaptive smoothing is less, compared to the airfoil case, but still improves the performance of the MG algorithm by a factor of 2 to 2.54, depending on the AS–MG variant used. For this flow case, the performance results using different adaptivity criteria show small differences. Concerning the variable γ, conclusions similar to the airfoil

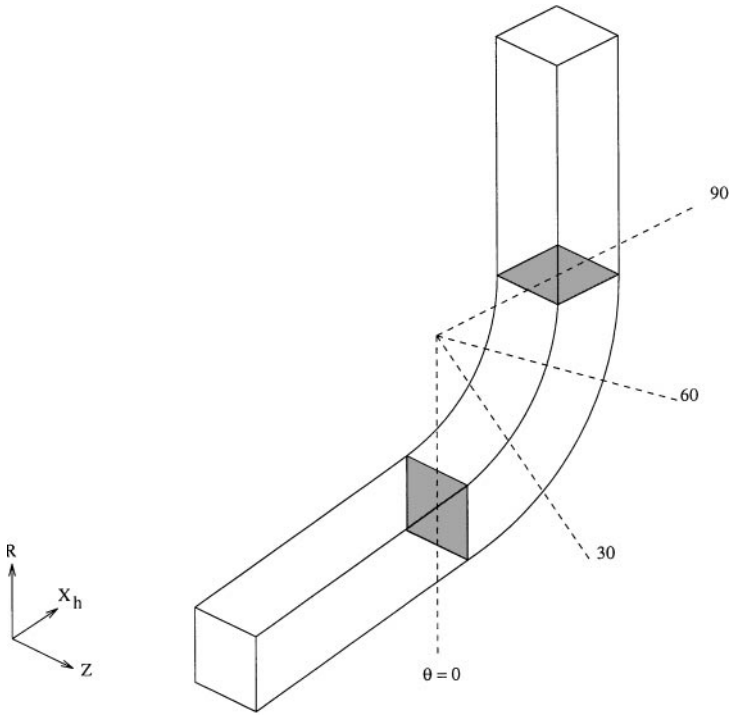

FIG. 13. Schematic of the 90° curved channel.

case can also be drawn here. The use of variable γ accelerates the solution by about the same factor with the constant γ and, like the airfoil cases, also leads to more stable solutions.

The most interesting conclusion for this particular flow is that the adaptive smoothing is more efficient when it is implemented only in the finest grid (AS_2 variant) and this applies for all adaptivity criteria considered here. This numerical behavior is exactly the opposite from the results obtained for the airfoil flow. A possible explanation for this is as follows: Although both flows are incompressible, the flow around the airfoil has

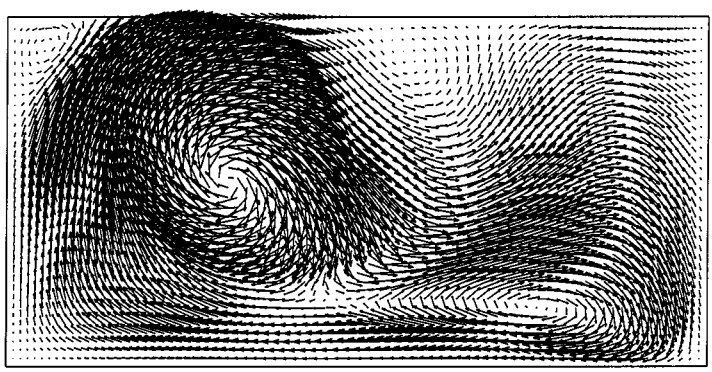

FIG. 14. Formation of the secondary flow at $\theta = 90°$.

TABLE IX
Results for the Curved Channel Case Using the Absolute Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					31	766	1.00
AS–MG	1	1		5	88	671	1.14
AS–MG	0	1	1.1	5	31	361	2.12
AS_1–MG	1	1		5	39	359	2.13
AS_2–MG	1	1		5	36	381	2.01
AS–MG	0	1	1.1	10	32	324	2.36
AS_1–MG	1	1		10	69	436	1.76
AS_1–MG	0	1	1.1	10	32	316	2.42
AS_2–MG	1	1		10	34	306	2.50
AS_2–MG	0	1	1.	10	30	349	2.19
AS_2–MG	0	1	1.1	10	30	323	2.37

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

large regions in which the solution does not significantly change during the computations. However, the elliptic character of the flow is more dominant in the curved channel case and, therefore, changes of the solution in subsets of the domain are much more likely to influence the solution in the neighboring regions. This behavior is more severe on coarser grids and, thus, the AS approach performs better when it is implemented only in the finest grid.

TABLE X
Results for the Curved Channel Case Using Relative *C* Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					31	766	1.00
AS–MG	0.1	0.1		5	48	464	1.65
AS–MG	0	0.1	1.1	5	39	433	1.77
AS_1–MG	0.1	0.1		5	41	428	1.79
AS_2–MG	0.1	0.1		5	32	355	2.16
AS_2–MG	0	0.1	1.1	5	30	374	2.05
AS_2–MG	0.2	0.2		5	84	684	1.12
AS_2–MG	0	0.2	1.1	5	30	359	2.13
AS_2–MG	0.1	0.1		10	41	361	2.12
AS_2–MG	0	0.1	1	10	31	389	1.97
AS_2–MG	0	0.1	1.1	10	32	358	2.14
AS_2–MG	0	0.2	1.1	10	32	359	2.13
AS_2–MG	0.05	0.05		20	38	397	1.93
AS_2–MG	0.1	0.1		20	40	301	2.54
AS_2–MG	0	0.1	1	20	30	359	2.13
AS_2–MG	0	0.1	1.1	20	31	346	2.21

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

TABLE XI
Results for the Curved Channel Case Using the Relative L_2 Criterion

Method	γ_{min}	γ_{max}	q_{max}	n_s	MG sweeps	Work units	Acceleration
MG					31	766	1.00
AS–MG	1	1		5	52	633	1.21
AS–MG	0	1	1.1	5	43	516	1.48
AS_1–MG	1	1		5	40	467	1.64
AS_2–MG	1	1		5	30	362	2.12
AS–MG	0	1	1.1	10	44	505	1.52
AS_1–MG	0	1	1.1	10	41	467	1.64
AS_2–MG	1	1		10	33	336	2.28
AS_2–MG	0	1	1.1	10	30	353	2.17
AS_2–MG	1	1		20	35	318	2.41
AS_2–MG	0	1	1.1	20	29	335	2.29

Note. The table includes results from the implementation of three variants of the approach: in all three grids (AS), in the intermediate and finest grids (AS_1), and in the finest grid only (AS_2). See footnote of Table III for additional information.

6. CONCLUSIONS

Implementation and investigation of an adaptive-smoothing approach in conjunction with the FMG–FAS method was presented.

Three different adaptivity criteria were investigated both for attached and separated flows around a NACA 0012 airfoil, as well as for three-dimensional flow in a curved channel. Several numerical experiments were performed showing that additional acceleration of the computations can be achieved by AS–MG. In addition to the use of constant adaptivity parameters, a dynamic procedure (dynamic- adaptivity) for defining variably these parameters was also proposed. The main difference between the AS–MG variants based on static and dynamic adaptivity is that the former makes use of information about the current residuals, whereas the latter exploits information about the convergence history of the iterative process and, therefore, leads to more stable numerical solutions.

Furthermore, the numerical experiments showed that for external airfoil flow the AS–MG performs better if it is applied in all grid levels of the MG algorithm. In contrast, for internal flow it was found that the AS–MG works more efficiently if it is applied only in the finest grid. It was also found that AS–MG provides greater acceleration for the airfoil flow cases than for channel flow. The absolute criterion was found to lead to a gradual reduction of the CVs as the computation progresses, resulting in fast reduction of the active set toward the end of the computation. In contrast, for the relative C criterion the number of CVs fluctuates during the iterations. However, irrespective of the particular numerical behavior resulting from different adaptivity criteria, it was found that the overall acceleration is achieved through a proper balance of the size of the active set (i.e., the number of CVs involved in the computation) and total number of MG sweeps required for convergence.

The present work investigated the idea of adaptive smoothing for the solution of the incompressible Navier–Stokes equations in conjunction with the FMG–FAS and the artificial-compressibility method. However, the same idea can also be exploited for compressible flows as well as for other physical systems (e.g., in electromagnetism). The implementation of adaptive-smoothing can also offer significant advantages in problems which require large

computing resources (e.g., direct numerical simulation of turbulence) and research toward this direction needs to be performed. Furthermore, the potential to accelerate even further fluidflow computations can possibly be achieved if other approaches for determining the dynamic-adaptivity parameters are developed. The latter as well as the extension of the AS–MG to unsteady incompressible and compressible flows is part of our future research agenda.

ACKNOWLEDGMENT

Part of this work has been supported by BFSI under grant MM-811.

REFERENCES

1. D. Drikakis, O. P. Iliev, and D. P. Vassileva, A nonlinear multigrid method for the three-dimensional incompressible Navier–Stokes equations, *J. Comput. Phys.* **146**, 301 (1998).
2. R. P. Fedorenko, A relaxation method for solving elliptic difference equations, *USSR Comput. Math. Phys.* **1**, 1092 (1961).
3. N. S. Bakhvalov, On the convergence of a relaxation method with natural constraints on the elliptic operator, *USSR Comput. Math. Phys.* **6**, 101 (1966).
4. A. Brandt, A multilevel adaptive solution of boundary value problems, *Math. Comput.* **31**, 333 (1977).
5. W. Hackbusch, *Multi-Grid Methods and Applications* (Springer-Verlag, Berlin/Heidelberg/New York/Tokyo, 1985).
6. P. Wesseling, *An Introduction to Multigrid Methods* (Wiley, New York, 1991).
7. R. Radespiel and R. C. Swanson, Progress with multigrid schemes for hypersonic flow problems, *J. Comput. Phys.* **116**, 103 (1995).
8. P. Lotstedt, Improved convergence to the steady state of the Euler equations by enhanced wave propagation, *J. Comput. Phys.* **114**, 34 (1994).
9. A. Brandt, *Barriers to Achieving Textbook Multigrid Efficiency (TME) in CFD*, Gauss Center Report WI/GC-10a (The Weizman Institute of Science, Rehovot, Israel, Dec. 1998).
10. A. Jameson, Solution of the Euler equations for two dimensional transonic flow by a multigrid method, *Appl. Math. Comput.* **13**, 327 (1983).
11. A. Jameson, Computational transonics, *Comm. Pure Appl. Math.* **41**, 507 (1988).
12. F. Liu and A. Jameson, Multigrid Navier–Stokes calculations for three-dimensional cascades, *AIAA J.* **31**, 1785 (1993).
13. J. Steelant, E. Dick, and S. Pattijn, Analysis of robust multigrid methods for steady viscous low Mach number flows, *J. Comput. Phys.* **136**, 603 (1997).
14. D. Drikakis and S. Tsangaris, Local solution acceleration method for the Euler and Navier–Stokes equations, *AIAA J.* **30**, 340 (1992).
15. D. Bai and A. Brandt, Local mesh refinement multilevel techniques, *SIAM J. Sci. Stat. Comput.* **8**, 109 (1987).
16. S. McCormick, *Multilevel Adaptive Methods for Partial Differential Equations* (SIAM, Philadelphia, 1989).
17. C. Zenger, Sparce grids, in *Parallel Algorithms for Partial Differential Equations, Proc. of the Sixth GAMM-Seminar, Kiel, 1990*, edited by W. Hackbusch, Notes on Numerical Fluid Mechanics (Vieweg Verlag, Braunschweig, 1991), Vol. 31.
18. M. Griebel, Sparse grid methods, their parallelization and their application to CFD, in *Parallel Computational Fluid Dynamics 1992, Rutgers, USA*, edited by J. Häuser (Elsevier, North-Holland, 1993).
19. A. Brandt, Rigorous local mode analysis of multigrid: I. Constant coefficient two level cycle with L_2 norm, *SIAM J. Numer. Anal.* **31**, 1695 (1994).

20. U. Rüde, On the multilevel adaptive iterative method, in *Preliminary Proc. of the 2nd Copper Mountain Conference on Iterative Methods*, edited by T. Manteuffel (University of Colorado at Denver, 1992).
21. U. Rüde, Fully adaptive multigrid methods, *SIAM J. Numer. Anal.* **30**, 230 (1993).
22. R. Southwell, *Relaxation Methods in Engineering Science—A Treatise in Approximate Computation* (Oxford University Press, London, 1940).
23. D. Drikakis, P. Govatsos, and D. Papantonis, A characteristic based method for incompressible flows, *Int. J. Numer. Methods Fluids* **19**, 667 (1994).