JOURNAL OF PROPULSION AND POWER
Vol. 12, No. 1, January–February 1996

Upwind Relaxation Multigrid Method for Computing Three-Dimensional, Viscous Internal Flows

Jack R. Edwards*
North Carolina State University, Raleigh, North Carolina 27695

The development of an efficient, implicit multigrid strategy for solving the three-dimensional, compressible Navier–Stokes equations is presented in this work. Emphasis is placed on the rapid computation of high Reynolds number flows within configurations characteristic of aircraft propulsion systems. In this context, a planar Gauss–Seidel implicit operator based on a robust, approximate linearization of the AUSMV low-diffusion flux-vector splitting scheme is used as a smoother for a full multigrid/full approximation-storage convergence acceleration strategy. Four test cases are considered: 1) Mach 0.3 turbulent flow through a contoured diffuser, 2) laminar transonic flow through a prototype quiet wind-tunnel geometry, 3) Mach 2.89 turbulent flow over a blended cylinder–cone geometry, and 4) Mach 10 laminar flow through a sidewall-compression scramjet inlet. The grid sizes for the simulations range from approximately 0.5M nodes to over 2.9M nodes. For all test cases, the effects of grid refinement and choice of cycling strategy on convergence rates are examined, and comparisons between multigrid and single-grid approaches are presented.

Nomenclature

$A^{\pm}$	=	flux Jacobian matrices
a	=	speed of sound
D, L, U	=	components of Gauss–Seidel splitting
$\hat{E}, \hat{F}, \hat{G}$	=	inviscid fluxes
$\hat{E}^c$	=	convective flux
$\hat{E}^p$	=	pressure flux
E_t	=	total energy
$\hat{E}_v, F_v, \hat{G}_v$	=	viscous fluxes
e	=	specific internal energy
$\boldsymbol{F}$	=	residual or defect error
J	=	Jacobian of Cartesian-to-curvilinear transformation
M	=	Mach number
p	=	pressure
$\bar{p}$	=	prolongation operator
$\boldsymbol{R}$	=	residual error
$\bar{r}$	=	restriction operator
$\boldsymbol{U}$	=	vector of conserved variables
u, v, w	=	velocity components
$\boldsymbol{W}$	=	vector of primitive variables $(\rho, u, v, w, e)^T$
$\tilde{\boldsymbol{W}}$	=	vector of primitive variables $(\rho, u, v, w, p)^T$
X_{sw}	=	distance from leading edge to sidewall
X_t	=	distance from leading edge to throat
α, β	=	sonic point switching functions
ε, κ	=	slope limiter parameters
ξ, η, ζ	=	curvilinear coordinates
ρ	=	density
ω_b, ω_f	=	underrelaxation factors

Subscripts

k	=	grid level
L, R	=	left and right states about cell interface
x, y, z	=	differentiation variables
1/2	=	cell interface value
∞	=	freestream value

Superscript

n	=	time level

I. Introduction

PROPULSION system flowfields for aerospace configurations are often characterized by strong interactions between an inviscid core flow and multiple near-wall viscous regions. To account for such strong viscid/inviscid coupling, propulsion system design, analysis, and optimization methods based on full three-dimensional solutions of the compressible Navier–Stokes equations should be used. Unfortunately, detailed Navier–Stokes simulations of complex propulsion system flowfields often require extremely long run times, as complex flow patterns, strongly-mixed hyperbolic-elliptic regions, and high spatial resolution requirements combine to produce a large degree of numerical stiffness. In this article, an algorithmic approach that utilizes an upwind relaxation method as a smoother within a multigrid cycle is considered as a means of efficiently computing such complex flowfields.

Upwind relaxation techniques have evolved from a realization that the coefficient matrices resulting from a linearization of an upwind discretization of the three-dimensional Navier–Stokes set are fairly well conditioned, thus allowing the use of implicit operators based on planar Gauss–Seidel matrix splittings. In contrast to point Gauss–Seidel approaches, planar Gauss–Seidel splittings can be very good approximations to the actual Jacobian of the discretized Navier–Stokes system, particularly if the direction for relaxation is nominally aligned with the streamwise movement of the flow. As a result, fast convergence rates can often be achieved for flowfields that are predominately streamwise supersonic.[1–4] The presence of large regions of streamwise elliptic flow can degrade the convergence rate of planar Gauss–Seidel techniques, as the factorization error of a streamwise-aligned Gauss–Seidel matrix splitting is directly proportional to the amount of upstream influence allowed by the discretization.

A more nonlinear approach to upwind relaxation, described in Ref. 5, has shown success in alleviating this convergence degradation for shock-separated, three-dimensional internal flows. Further improvements in convergence rate have been

Presented as Paper 95-0208 at the AIAA 33rd Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan. 9–12, 1995; received Feb. 3, 1995; revision received May 26, 1995; accepted for publication May 31, 1995.

*Assistant Professor, Department of Mechanical and Aerospace Engineering, Campus Box 7910. Member AIAA.

obtained by incorporating a simple two-grid acceleration technique.[6] Unfortunately, the implementation of Ref. 6 is cumbersome, storage-intensive, and not extendable to more than two grid levels.

This article describes the next logical step in this line of algorithm development, the implementation of a more conventional planar Gauss–Seidel method within the context of a full multigrid/full approximation-storage (FMG–FAS) convergence acceleration strategy. The benefits expected to arise from this extension include good convergence behavior across the Mach number range, a decrease in complexity and storage requirements, and an increase in the mathematical rigor of the algorithm. To increase the generality of the approach even further, the hybrid upwind/central discretization utilized in Refs. 5 and 6 has been replaced by a full upwind discretization based on the AUSMV method of Wada and Liou.[7]

The remainder of this work describes the construction of the upwind relaxation-multigrid Navier–Stokes solver and its testing for an array of three-dimensional, viscous flows. The governing equations and details of the turbulence closure are first provided and the AUSMV flux-vector splitting is presented in a compact notation. A higher-order extension of AUSMV is then discussed and a procedure for deriving a robust approximate linearization of the scheme is outlined. The construction of the planar Gauss–Seidel implicit operator is briefly described and the extension of the relaxation approach to multiple grids and the cycling strategies employed in this investigation are outlined. Four test cases are discussed next: 1) a Mach 0.3 turbulent flow through a contoured diffuser similar to that utilized by Benay et al.,[8] 2) a laminar transonic flow through a prototype quiet wind-tunnel nozzle geometry,[9,10] 3) a Mach 2.89 turbulent flow over a blended cylinder–cone configuration,[11–13] and 4) a Mach 10 laminar flow through a sidewall compression inlet.[14] Contour and grid plots are presented to illustrate the flow complexity and the grid resolution, some comparisons with experimental data are shown, and convergence histories are presented to illustrate the numerical performance of the algorithm. Finally, some conclusions are drawn regarding the suitability of the upwind relaxation-multigrid methodology for large-scale Navier–Stokes computations.

II. Governing Equations and Discretization

The flowfields under consideration are governed by the three-dimensional, compressible Navier–Stokes equations. In terms of a curvilinear coordinate system defined by the steady transformation $\xi = \xi(x, y, z)$, $\eta = \eta(x, y, z)$, $\zeta = \zeta(x, y, z)$, this system can be expressed as

$$\frac{\partial}{\partial t}\left(\frac{\boldsymbol{U}}{J}\right) + \frac{\partial(\hat{\boldsymbol{E}} - \hat{\boldsymbol{E}}_v)}{\partial \xi} + \frac{\partial(\hat{\boldsymbol{F}} - \hat{\boldsymbol{F}}_v)}{\partial \eta} + \frac{\partial(\hat{\boldsymbol{G}} - \hat{\boldsymbol{G}}_v)}{\partial \zeta} = 0 \quad (1)$$

For the turbulent simulations, Reynolds stress effects are incorporated by invoking the Boussinesq hypothesis, thus reducing the problem to the specification of an eddy viscosity and a turbulent Prandtl number (0.9 in this work). A one-equation model[5,12] is used to provide the eddy viscosity field directly. This equation is updated in a weakly coupled fashion following the completion of each multigrid cycle.[12]

In this work, the AUSMV scheme of Wada and Liou[7] is used for the control-volume discretization of the inviscid fluxes. The AUSMV is a modification of the advection upstream splitting method (AUSM) of Liou and Steffen[15] that allows for a monotone capturing of strong shock waves while preserving the accuracy of the original AUSM in shear-layer resolution. To begin the description of the AUSMV approach and its linearization, $\hat{\boldsymbol{E}}$ in the ξ direction is expressed as a sum of convective and pressure components:

$$\hat{\boldsymbol{E}} \equiv \hat{\boldsymbol{E}}^c + \hat{\boldsymbol{E}}^p = (|\nabla\xi|/J)\rho a M \tilde{\boldsymbol{E}}^c + (|\nabla\xi|/J)p\tilde{\boldsymbol{E}}^p \quad (2)$$

where

$$\tilde{\boldsymbol{E}}^c = \begin{bmatrix} 1 \\ u \\ v \\ w \\ H \end{bmatrix}, \qquad \tilde{\boldsymbol{E}}^p = \begin{bmatrix} 0 \\ \tilde{\xi}_x \\ \tilde{\xi}_y \\ \tilde{\xi}_z \\ 0 \end{bmatrix} \quad (3)$$

$$\tilde{\xi}_{x,y,z} = \xi_{x,y,z}/|\nabla\xi| \quad (4)$$

$$H \equiv (1/\rho)(E_t + p) \quad (5)$$

The contravariant M is given as

$$M = (1/a)(\tilde{\xi}_x u + \tilde{\xi}_y v + \tilde{\xi}_z w) \quad (6)$$

Utilizing the notation described in Ref. 16, a compact form for the AUSMV interface flux is defined by considering the behavior of the convective and pressure components separately. The convective portion of the interface flux $\hat{\boldsymbol{E}}^c_{1/2}$ is given by

$$\hat{\boldsymbol{E}}^c_{1/2} = (|\nabla\xi|/J)(C^+\tilde{\boldsymbol{E}}^c_L + C^-\tilde{\boldsymbol{E}}^c_R) \quad (7)$$

where

$$C^\pm = \tfrac{1}{2}(C_{1/2} \pm |C_{1/2}|) \quad (8)$$

$$C_{1/2} = \rho_L a_L \tilde{M}^+_L + \rho_R a_R \tilde{M}^-_R \quad (9)$$

The split Mach numbers $\tilde{M}^\pm_{L,R}$ are defined as follows

$$\tilde{M}^+_L = \alpha^+_L(1 + \beta_L)M_L - \frac{2\beta_L}{a_L + a_R}\left[+\frac{a_L}{4}(M_L + 1)^2 + \frac{(a_R - a_L)}{2}\alpha^+_L M_L\right] \quad (10)$$

$$\tilde{M}^-_R = \alpha^-_L(1 + \beta_R)M_R - \frac{2\beta_R}{a_L + a_R}\left[-\frac{a_R}{4}(M_R - 1)^2 + \frac{(a_L - a_R)}{2}\alpha^-_R M_R\right] \quad (11)$$

The functions $\alpha^\pm$ and β provide the correct sonic-point transition behavior[16]:

$$\alpha^\pm_{L,R} = \tfrac{1}{2}[1 \pm \mathrm{sgn}(M_{L,R})] \quad (12)$$

$$\beta_{L,R} = -\max[0, 1 - \mathrm{int}(|M_{L,R}|)] \quad (13)$$

The particular form of the split Mach number allows the numerical diffusion of the scheme to vanish if the cell interface behaves exactly as a stationary contact discontinuity, with M_L and M_R approaching zero while the pressures p_L and p_R equilibrate. This mechanism for diffusion removal is very similar to that of Roe's scheme and similar results for shear-layer resolution and shock capturing are generally obtained.[7]

The pressure contribution to the interface flux is defined as follows:

$$\hat{\boldsymbol{E}}^p_{1/2} = (|\nabla\xi|/J)\tilde{\boldsymbol{E}}^p(D^+_L p_L + D^-_R p_R) \quad (14)$$

where

$$D^\pm_{L,R} = \alpha^\pm_{L,R}(1 + \beta_{L,R}) - \beta_{L,R}P^\pm_{L,R} \quad (15)$$

Different forms for the subsonic pressure splitting $P^\pm$ can be

utilized. For the cases presented in this work, the simple splitting proposed in Ref. 15 is adopted:

$$P^{\pm}_{L,R} = \tfrac{1}{2}(1 \pm M_{L,R}) \tag{16}$$

In all of the previous expressions, the flow properties at left L and right R states are determined by Fromm interpolations of the primitive variable vector $\tilde{W} = (\rho, u, v, w, p)^T$ to the $i + \frac{1}{2}$ interface:

$$\tilde{W}_R = \tilde{W}_{i+1,j} - \frac{\Psi_{i+1}}{4}(\tilde{W}_{i+2,j} - \tilde{W}_{i,j}) \tag{17}$$

$$\tilde{W}_L = \tilde{W}_i + \frac{\Psi_i}{4}(\tilde{W}_{i+1,j} - \tilde{W}_{i-1,j}) \tag{18}$$

The function Ψ is a pressure-based slope-limiting operator that is designed to reduce the interpolation to first-order accuracy in regions of large gradient in the solution:

$$\Psi_i = \kappa\left(1 - \frac{|p_{i+1,j} - p_{i-1,j}|}{|p_{i+1,j} - p_{i-1,j}| + \varepsilon p_\infty}\right) \tag{19}$$

where κ is zero for a strict first-order interpolation and unity for a second-order limited interpolation. The quantity ε controls the sensitivity of the pressure limiter. This particular form of slope-limiter is not guaranteed to produce completely nonoscillatory solutions, but will, in general, not inhibit convergence to the steady state.

Approaches similar to that outlined previously are used for the inviscid interface fluxes in the η and ζ directions. The viscous components of Eq. (1) are central differenced to second-order accuracy. Standard boundary conditions are employed for solid surfaces and for supersonic inflow/outflow. Stagnation pressure, stagnation temperature, and flow direction are held fixed for a subsonic inflow; the velocity magnitude is extrapolated from the interior of the domain. For a subsonic outflow, pressure is held fixed, while entropy, the transverse velocity components, and the "outward" Riemann invariant are extrapolated from the interior.

III. Linearization

Implicit techniques for integrating the unsteady Navier–Stokes equations generally require a local time linearization of the nonlinear discrete system. When advanced upwinding approaches are utilized, the construction of an appropriate linearization for the inviscid fluxes can be a time-consuming task, due both to complexities inherent in the definition of the interface flux and to difficulties associated with the use of slope-limited higher-order interpolations. The latter difficulty is usually overcome by only performing the linearization to first-order accuracy. Such a convention maintains a block-tridiagonal structure for the linearization of each flux derivative, thus facilitating the use of alternating-direction inplicit (ADI) factorization procedures. In the development that follows, we abide by this convention and consider only the version of AUSMV formed by setting κ in Eq. (19) to zero.

We can construct a linearization of the AUSMV interface flux $\hat{E}_{i+1/2}$ in terms of the variable set $W = (w1, w2, w3, w4, w5)^T$ by defining

$$\frac{\partial \hat{E}_{i+1/2}}{\partial W_{i+1/2}}\Delta W_{i+1/2} \equiv A^+_{1/2}\Delta W_i + A^-_{1/2}\Delta W_{i+1} \tag{20}$$

where

$$A^+_{1/2} \equiv \frac{\partial(\hat{E}^c_{1/2} + \hat{E}^p_{1/2})}{\partial W_i}$$

$$A^-_{1/2} \equiv \frac{\partial(\hat{E}^c_{1/2} + \hat{E}^p_{1/2})}{\partial W_{i+1}}$$

As shown in Ref. 16, hybrid flux-vector/flux-difference splittings such as AUSMV are amenable to an exact term-by-term linearization, provided that the switching functions $\alpha^{\pm}$, β, etc., are held locally constant. In the case of AUSMV, the resulting flux Jacobians A^+ and A^- are quite complex, with each displaying a dependence on fluid properties at states i and $i + 1$ about the interface. A simpler, approximate linearization can be derived by contracting the exact expressions for the positive and negative flux Jacobians $A^+_{1/2}$ and $A^-_{1/2}$ to the single grid nodes i and $i + 1$, respectively. The result is the following implicit representation:

$$\frac{\partial \hat{E}_{i+1/2}}{\partial W_{i+1/2}}\Delta W_{i+1/2} = A^+_i\Delta W_i + A^-_{i+1}\Delta W_{i+1} \tag{21}$$

where

$$A^{\pm}_i = \frac{|\nabla\xi|}{J}\left\{\rho a\alpha^{\pm}M\frac{\partial \tilde{E}_c}{\partial W} + \tilde{E}^c\left[\rho a D^{\pm}w^T + \tilde{M}^{\pm}x^T \mp \frac{\beta\rho}{4}(|M| - 1)^2 y^T\right] + D^{\pm}\tilde{E}^p z^T\right\}_i \tag{22}$$

$$w^T = \left(\frac{\partial M}{\partial w1}, \frac{\partial M}{\partial w2}, \ldots, \frac{\partial M}{\partial w5}\right) \tag{23}$$

$$x^T = \left(\frac{\partial \rho a}{\partial w1}, \frac{\partial \rho a}{\partial w2}, \ldots, \frac{\partial \rho a}{\partial w5}\right) \tag{24}$$

$$y^T = \left(\frac{\partial a}{\partial w1}, \frac{\partial a}{\partial w2}, \ldots, \frac{\partial a}{\partial w5}\right) \tag{25}$$

$$z^T = \left(\frac{\partial p}{\partial w1}, \frac{\partial p}{\partial w2}, \ldots, \frac{\partial p}{\partial w5}\right) \tag{26}$$

In the previous expressions, $D^{\pm}$ is defined as in Eq. (15), and $\tilde{M}^{\pm}$ is given as

$$\tilde{M}^{\pm} = \alpha^{\pm}(1 + \beta)M \mp (\beta/4)(M \pm 1)^2 \tag{27}$$

Metric derivatives utilized in the definition of the flux Jacobians can either be evaluated consistently at the cell interface or averaged to the individual cell centers i and $i + 1$. Further details regarding the implicit formulation of low-diffusion flux-splitting techniques can be found in Ref. 16.

IV. Relaxation Approach

The time-integration scheme used in this investigation is based on a planar Gauss–Seidel partitioning of the system Jacobian matrix. The approach for constructing such an implicit operator has been described in Ref. 5; the only changes needed for this investigation involve the addition of the flux Jacobians described earlier. At time level n, the basic Gauss–Seidel iterative procedure can be represented in matrix notation as follows.

Step 1: backward sweep in ξ direction (i coordinate):

$$\text{solve:}\quad D^n_i\delta W^{n+1/2}_i = -\omega_b[F(W^n_i) + U^n_{i+1}\delta W^{n+1/2}_{i+1}]$$

$$\text{update:}\quad W^{n+1/2} = W^n + \delta W^{n+1/2}$$

Step 2: forward sweep in ξ direction (i coordinate):

$$\text{solve:}\quad D^{n+1/2}_i\delta W^{n+1}_i = -\omega_f[F(W^{n+1/2}_i) + L^{n+1/2}_{i-1}\delta W^{n+1}_{i-1}]$$

$$\text{update:}\quad W^{n+1} = W^{n+1/2} + \delta W^{n+1} \tag{28}$$

The matrix D_i consists of all matrix elements corresponding to a constant ξ plane, and L_{i-1} and U_{i+1} are composed of the off-plane elements corresponding to the linearization of the ξ flux derivative. At each ξ station, the inversion of the sparse, block–pentadiagonal matrix D_i is accomplished via an ADI technique, stabilized through the use of Broyden subiterations.[5,17] The vector $\boldsymbol{F}$ represents the discretized residual (or collected defect error for the multigrid procedures) and ω_b and ω_f are 0.9 and 0.7, respectively. Planar matrix storage is utilized to render the algorithm reasonably efficient in terms of memory usage. Cray Y-MP run-time memory for the entire algorithm (including the multigrid extension) is approximately 39 words per grid point.

Algorithm (28) is used as the smoother for a multigrid acceleration method discussed later. A fully implicit treatment of the boundary conditions is used on all grid levels. While such an implementation involves some extra logic (e.g., to prevent nonzero interpolated corrections from updating no-slip or constant-temperature surfaces), it was found necessary to maintain a high rate of convergence.

V. Multigrid Approach

Several excellent references that describe both the theory and the implementation of multigrid algorithms are available in the literature.[18–20] While most of the theoretical developments have concentrated on simple linear elliptic problems, there has been substantial use of multigrid ideas for nonlinear and nonelliptic problems, with the full approximation-storage (FAS) concept of Brandt[18] being the basis for most extensions. Used in combination with a nested iteration [or full multigrid (FMG)] initialization strategy, the FAS multigrid scheme has proven exceedingly popular for fluid flow computations.[21–23]

Many current multigrid-based algorithms for solving the Navier–Stokes equations are based on explicit or point-implicit local smoothers, as these approaches are simple and easily vectorizable. However, for computations involving severe mesh clustering in one or more directions, the simpler strategies often fail, due mostly to a degradation in the damping properties of the schemes with increasing cell AR. Another problem, termed strong alignment,[24] results when the direction of the flow aligns with one particular coordinate direction. High-frequency errors perpendicular to that direction cannot be removed by a local smoother because of the lack of coupling. Such errors cannot be adequately represented on the coarser grids, leading to the failure of the multigrid strategy. For viscous flows at high Reynolds numbers, both of these problems can be present, and multilevel approaches that work well for inviscid flow problems can experience severe convergence degradation. One way of alleviating the AR and alignment problems simultaneously for three-dimensional flows is to utilize a more global smoother, such as the planar relaxation algorithm (28). The directions for the sweeps can be aligned such that the scheme is fully implicit in the direction(s) with heavy mesh clustering, thus removing or reducing the associated time-step restriction. Algorithms of this type can be difficult to program and are generally harder to extend to arbitrary domains, but nevertheless, they would appear to be quite suitable for computing viscous internal flows. This research effort attempts to verify this conjecture.

In addition to the choice of an appropriate smoother, the development of a FMG–FAS multigrid approach requires two other ingredients, one of which is a restriction operator for transferring the defect error and the fluid properties to the coarser grid levels. The underlying coarse grids (up to four in this investigation) are determined by successively removing each fine grid line in all coordinate directions (full coarsening). As the basic discretization is node based, the restriction of the fluid properties to the coarser grids is accomplished by injection. A conservative restriction of the residual error is achieved by a weighed summation of the 27 fine-grid values that compose a coarse-grid cell on the interior of the mesh.[6] The restriction operator for the boundary residuals is also injection. The final ingredient in the construction of the multigrid algorithm is a prolongation operator for transferring the coarse-grid corrections (and the solution itself within the FMG phase) from coarser to finer grid levels. For this effort, the prolongation operator is simple trilinear interpolation.

Given that $\bar{r}$ represents a restriction operator and that $R(\boldsymbol{W}_k)$ is the Navier–Stokes residual at level k, the following sequence of operations is performed within the coarser levels of the FAS/planar relaxation V cycle: ($\bar{p}$ represents the prolongation operator).

Step 1: restrict solution values from levels $k-1$ to k:

$$\boldsymbol{W}_k = \bar{r}(\boldsymbol{W}_{k-1})$$

Step 2: smooth using backward Gauss–Seidel ($\boldsymbol{F}_{k-1}$ is the collected error at the previous grid level):

$$\text{solve:} \quad (D + \omega_b U)_k \delta \boldsymbol{W}_k^b = \omega_b \bar{r}(\boldsymbol{F}_{k-1})$$

Step 3: calculate residual error at level k:

$$\boldsymbol{F}_k = \bar{r}(\boldsymbol{F}_{k-1}) - [\boldsymbol{R}(\boldsymbol{W}_k + \delta \boldsymbol{W}_k^b) - \boldsymbol{R}(\boldsymbol{W}_k)] \qquad (29)$$

Step 4: smooth using forward Gauss–Seidel:

$$\text{solve:} \quad (D + \omega_f L)_k \delta \boldsymbol{W}_k^f = \omega_f \boldsymbol{F}_k$$

Step 5: recalculate residual error at level k:

$$\boldsymbol{F}_k = \bar{r}(\boldsymbol{F}_{k-1}) - [\boldsymbol{R}(\boldsymbol{W}_k + \delta \boldsymbol{W}_k^b + \delta \boldsymbol{W}_k^f) - \boldsymbol{R}(\boldsymbol{W}_k)]$$

Step 6: restrict residual error $\boldsymbol{F}_k$ to next coarsest level and continue cycling process:

Step 7: sum corrections at grid level k:

$$\delta \boldsymbol{W}_k = \delta \boldsymbol{W}_k^b + \delta \boldsymbol{W}_k^f + \bar{p}(\delta \boldsymbol{W}_{k+1})$$

The notation in steps 2 and 4 represents the application of the planar relaxation iteration (28) on the particular grid level k. The matrices that define the planar Gauss–Seidel operator are computed using the restricted fluid properties and the metric-derivative information corresponding to grid level k. Step 6 and the prolongation part of step 7 are not performed on the coarsest level. On the finest level, algorithm (28) is utilized directly, with F replaced by the discretized fine-grid residual. Coarse-grid residual values needed within steps 3 and 5 of Eq. (29) are computed using a first-order discretization for the inviscid components. A W cycle strategy can also be defined by performing steps 3–7 two times per grid level.

A good initial guess for the fine-grid cycling strategy is provided by a full multigrid (FMG) approach. The FMG procedure starts on the coarsest mesh, where the basic two-sweep relaxation scheme is used to solve the problem to a specified tolerance. The solution is then interpolated to the next finest mesh and is improved by the application of a number of two-level multigrid cycles. The process is repeated, each time increasing the number of grid levels used in the multigrid cycling. After the interpolation to the finest level, the procedure reverts to the standard cycling strategy.

VI. Results

The primary objective of this work is to evaluate the numerical performance of the planar relaxation/multigrid approach for large-scale Navier–Stokes simulations. As such, the complexity of the computed flowfields is de-emphasized in the discussions that follow. Some contour and grid plots

Table 1 Flow conditions

	Diffuser	Nozzle	Cylinder–cone	Inlet
Inflow Mach no.	0.3	0.55	2.89	10.0
Re/m	1.31×10^7	3.28×10^6	1.70×10^7	7.05×10^6
T_∞, K	242	140	105	49
T_{wall}/T_∞	Adiabatic	Adiabatic	Adiabatic	6.48
Boundary-layer state	Turbulent	Laminar	Turbulent	Laminar

are presented, however, to acquaint the reader with the general arrangement of the test geometries and the relevant features of the computed flows. Some comparisons with experimental data are also provided. Where appropriate, references are made to other works in the literature within which a more detailed description of the flowfields can be found. Unless otherwise noted, all computations utilize the second-order limited interpolations (17) and (18) to define the interface flux.

The numerical performance of the developed algorithm is measured in two ways. The most important measure, single-processor Cray Y-MP CPU time until a specified convergence criterion is reached, is emphasized as the abscissa of the residual history plots that follow. The ordinate of the residual history plots is the L_2 norm of the fine-grid residual vector, normalized by its initial value at the start of the FMG sequence. In this way, the level of fine-grid error reduction corresponding to the FMG initialization can be directly ascertained. Iterations corresponding to the coarse-grid stages of the FMG sequence are not shown; the CPU time required for these iterations is indicated, however, by a shift in the fine-grid residual history. To provide a fair evaluation, the single-grid computations used for comparison are also initialized using the FMG procedure. The number of cycles until convergence is also used as an indicator of the effectiveness of the algorithm. These numbers do not represent an equal amount of work, one V cycle is roughly 1.38 times as expensive as one fine-grid application of Eq. (28), while one W cycle is about 1.26 times as expensive as one V cycle.

Pertinent freestream conditions for each of the test cases are summarized in Table 1. Grid resolution constraints and other details are discussed in the subsections that follow.

A. Subsonic Diffuser Geometry

The first test case involves subsonic, turbulent flow through a rectangular cross-sectional diffuser similar to that considered by Benay et al.[8] The most noteworthy feature of the diffuser geometry is a humped lower wall, designed such that the maximum height of the hump is along a line angled obliquely to the sidewalls of the diffuser. The maximum height of the hump is 10 mm, and the length of the diffuser is 560 mm. Other details of the basic geometry can be found in Ref. 8. In the simulations, turbulent boundary layers develop on all four walls. The minimum normal spacing for the grid (shown in Fig. 1) is 0.003 mm, fine enough to ensure a minimum wall-coordinate $\hat{n}^+$ value of around 3.0 for all surfaces. The flow development, illustrated in Fig. 2, is rather benign, with axially separated flow only observed in the corner regions immediately downstream of the maximum hump displacement.

The convergence behavior of the upwind relaxation-multigrid approach for the diffuser flow is illustrated in Fig. 3. For the coarser $113 \times 73 \times 65$ grid, the W cycle strategy performs best, converging five orders-of-magnitude in 166 iterations, compared with 258 iterations for the V cycle. The higher per-iteration cost of the W cycle somewhat offsets the gain in cycle count, so that the actual gain in CPU time is more modest. The W cycle converges about three times faster than the single-grid iteration, and the scaling of the W cycle convergence rate for a resolution increase to $129 \times 113 \times$

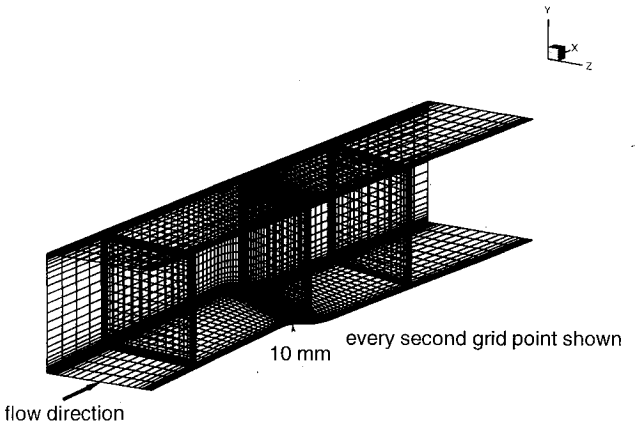

Fig. 1 Grid for subsonic diffuser.

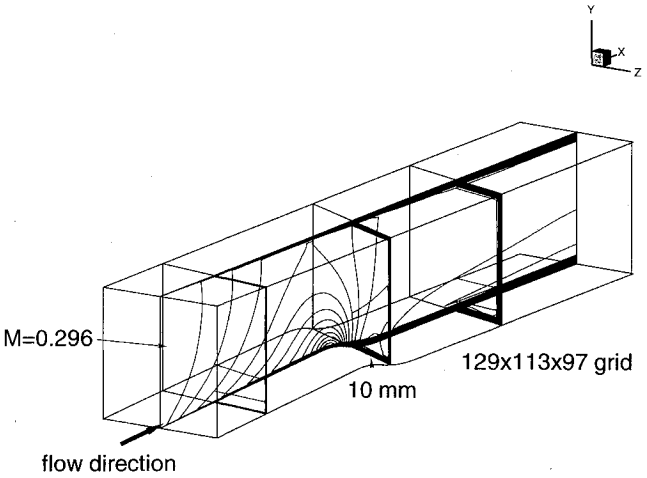

Fig. 2 Mach contours: subsonic diffuser.

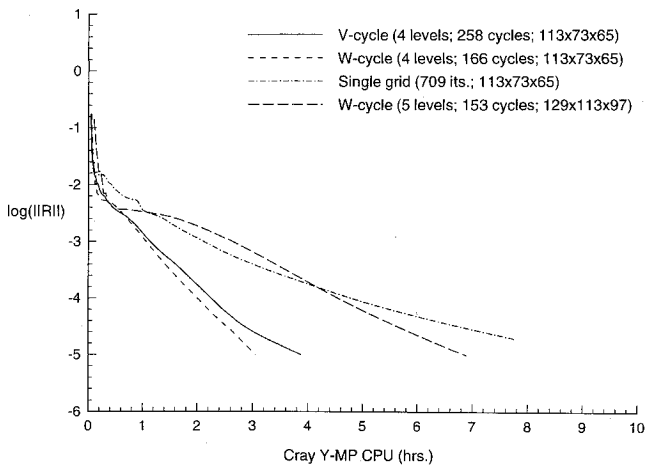

Fig. 3 Convergence histories: subsonic diffuser.

97 nodes is slightly better than linear. This result is probably due to the addition of another coarse-grid level to the cycle for the larger simulation.

B. Nozzle Geometry

The second test case considered in this investigation corresponds to laminar flow through a square cross-sectional quiet wind-tunnel nozzle, designed for an exit Mach number of 2.4. The grid arrangement for a side view of the nozzle is shown in Fig. 4a. As the nozzle is bilaterally symmetric, only one-quarter of the overall extent is actually considered in the calculations. To save storage, the computational domain is divided into two zones, one encompassing the throat region ($65 \times 129 \times 129$ nodes) and one encompassing the supersonic expansion section ($177 \times 129 \times 129$ nodes). The grid spacing near the wall is chosen such that 35–50 points are present in the thin centerline boundary layer (see Fig. 4b). This level of refinement is needed for the success of the linear stability analyses described in Ref. 10. The fundamental difference between the flow development within the square cross-sectional nozzle and that within the analogous axisymmetric nozzle is that a significant amount of transverse flow movement occurs in the viscous layers as a result of a nonuniform cross-sectional pressure distribution. The collision of the developing viscous layers at the corner results in the formation of a vortex system that grows rapidly in size, but weakens in strength as the flow moves downstream (see Fig. 5). More details regarding the structure and stability of the nozzle flow can be found in Ref. 10.

Convergence histories for the nozzle simulations are shown in Fig. 6. For the transonic flow in the throat section, the W cycle strategy is again slightly superior to the V cycle, converging the nozzle flow seven orders of magnitude in only 128 cycles. As shown, the W cycle iteration again converges almost three times faster than the corresponding single-grid calculation. Also illustrated in the figure is the effect of a first-order discretization on the relative performance of the W cycle and single-grid iterations. For the first-order simulation, the W cycle converges slightly faster and more regularly than before (114 vs 128 cycles), and the convergence improvement relative to the single-grid iteration is slightly better than threefold. These results also indicate that, at least for the nozzle flow, the errors associated with the mismatch in accuracy between the implicit and explicit operators do not appear to degrade significantly the performance of either the multigrid or the single-grid algorithm. In lieu of a refined-grid calculation, the convergence history for the remainder of the nozzle flow is also shown in Fig. 6. For this section, the inviscid core flow is supersonic, and the W cycle iteration again provides rapid convergence.

C. Cylinder–Cone Geometry

The third simulation corresponds to supersonic, turbulent flow over the cylinder–cone configuration considered by Wideman et al.[11] in their experimental investigation. While not an internal flow, the gridding requirements for the cylinder–cone geometry are similar, as points are clustered both to the cylinder–cone surface and to the $X-Y$ symmetry plane to resolve elements of the flow topology. The nonuniform pressure gradient caused by the asymmetric shock surface results in the formation of a vortical structure, aligned roughly with the innermost cylinder–cone juncture (see Fig. 7). The lateral symmetry constraint retards the azimuthal movement of the flow, resulting in a pocket of recirculating fluid on the bottom of the configuration. Computed and measured streamwise pressure distributions are compared in Fig. 8. In the figure, $\theta = 0$ deg corresponds to the top of the apparatus, and $\theta = 180$ deg corresponds to the bottom. Further discussion of the flow structure, including more comparisons with experimental data, can be found in Refs. 12 and 13.

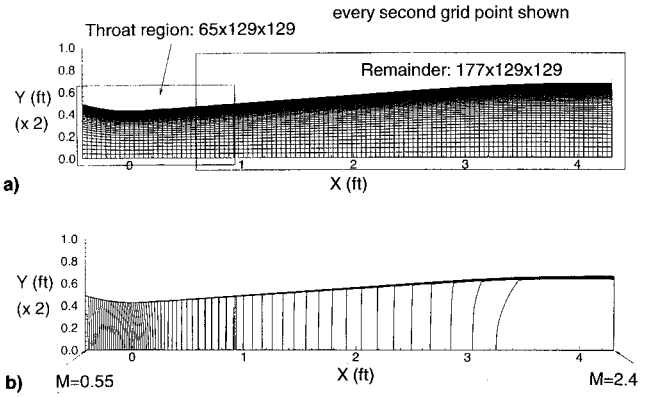

Fig. 4 Centerline grid and Mach contours: nozzle geometry, a) centerline grid and b) centerline Mach contours.

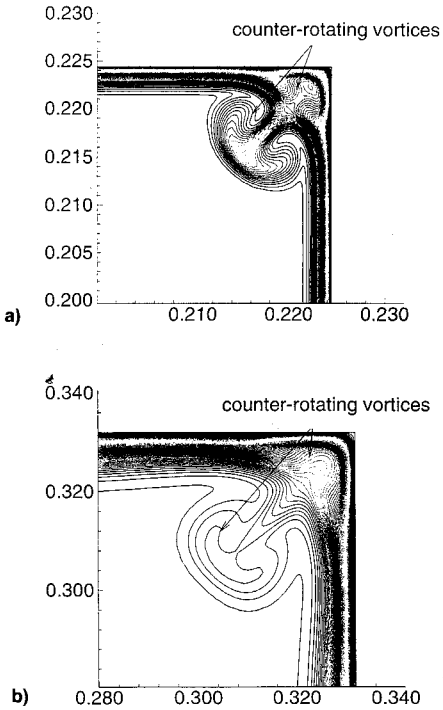

Fig. 5 Mach contours in corner: nozzle geometry, x = a) 0.5 and b) 4.0.

Convergence histories for the cylinder–cone simulations are illustrated in Fig. 9. Both the V and W cycle residual histories are characterized by a period of very rapid error reduction, followed by a period of residual increase and a period of slower error reduction. This pattern has been observed for other axially separated flows and may indicate the need for a stronger coupling between the turbulence equation and the mean-flow multigrid iteration. The turbulence equation is only advanced on the finest grid; eddy viscosity values are held constant throughout the coarse-grid acceleration process. For the coarser $97 \times 89 \times 65$ grid, the W cycle strategy is again superior to the V cycle in terms of iteration count (101 vs 112), but takes longer to converge. The V cycle iteration converges only two times faster than the single-grid iteration, which in itself performs well, converging in only 325 iterations. This degradation in relative performance for the predominately supersonic cylinder–cone interaction deserves further explanation.

As noted by Mulder,[25] among others, line (two-dimensional) or planar (three-dimensional) Gauss–Seidel methods can function more as a solver than as a smoother for situations in which the sweeping direction is aligned with the direction

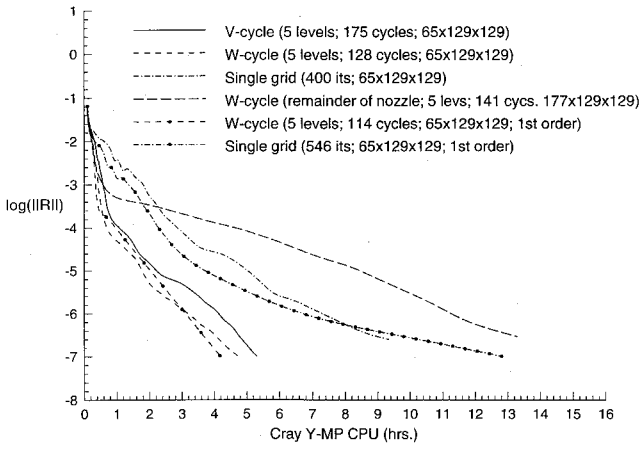

Fig. 6 Convergence histories: nozzle geometry.

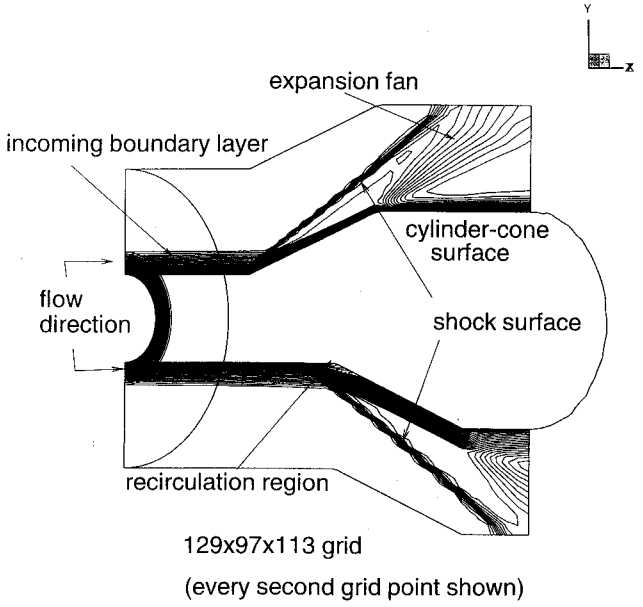

Fig. 7 Mach contours: cylinder–cone geometry.

of predominantly supersonic flow. Both high- and low-frequency error components can be attenuated effectively using only the fine-grid iteration, and the need for the coarse-grid correction disappears. In practice, of course, linearization and factorization errors inherent within the implicit formulation often prevent the algorithm from attaining such Newton-like performance, and the utility of multigrid cycling resurfaces. However, the relative improvement provided by multigrid acceleration for such situations may not be as significant as for purely elliptic flows, simply because the baseline relaxation scheme itself performs better.

The use of upwind-biased prolongation/restriction operators has also been cited as a means of improving multigrid performance for hyperbolic flows.[26–28] While the rationale behind the development of such approaches may be correct, many of the proposed operators have been designed for use only with particular discretizations and have been tested only for inviscid flows using local smoothers. Our attempts at utilizing some of these ideas to improve the performance of the present approach have been unsuccessful.

A slight decrease in the V cycle convergence rate (131 cycles vs 112) is evidenced for the refined-grid calculation (129 × 97 × 113 nodes). Again, turbulence-model coupling effects may play an important role in the attaining of grid-independent rates of convergence for the cylinder–cone interaction.

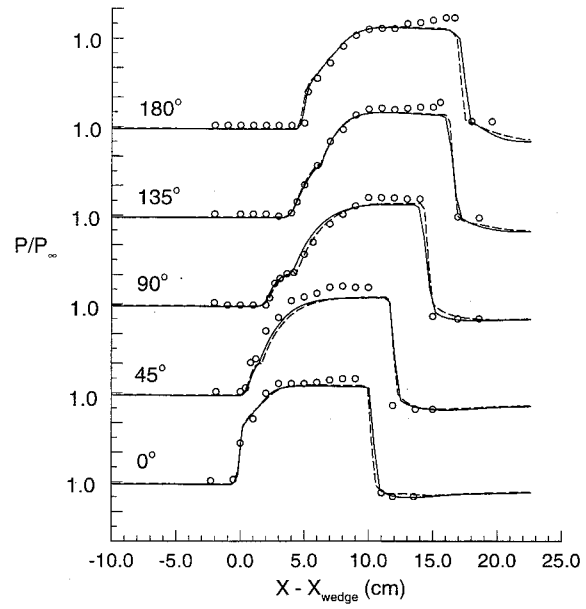

Fig. 8 Azimuthal pressure distributions: cylinder–cone geometry. ○, Wideman experiment; ——, 97 × 89 × 65 grid; and -----, 129 × 97 × 113 grid.

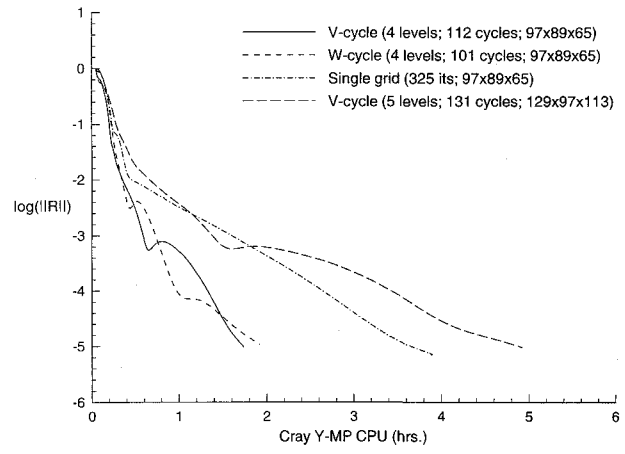

Fig. 9 Convergence histories: cylinder–cone geometry.

D. Hypersonic Inlet Geometry

The final test case considered in this study corresponds to an experiment of Holland[14] and involves Mach 10 laminar flow through a sidewall-compression scramjet inlet, operating at a contraction ratio of 3.0. Some of the geometric features of the inlet include the backward sweep of the leading edges of the inlet sidewalls, a variable-position cowl, and the angling of the sidewalls inward to achieve compression. Centerline X-Z symmetry is assumed in the simulations, and grid clustering to the forebody, sidewall, cowl, and centerline planes is utilized (see Fig. 10). The flow structure within the inlet develops in response to the crossing-shock system generated by the angling of the sidewalls. The shock waves interact with the forebody, cowl, and sidewall viscous layers, producing multiple regions of separated, vortical flow. Figure 11 compares computed and measured pressure distributions along the centerline of the forebody and sidewall. Good agreement is observed in the prediction of the extent of axial separation along the forebody, but relatively poor agreement is evidenced in the throat region $[(X - X_{sw})/(X_t - X_{sw}) > 1.0]$. As discussed in Ref. 14, the flow in this region is most probably transitional or turbulent, rather than laminar. Associated changes in the viscous-layer displacement pattern may result

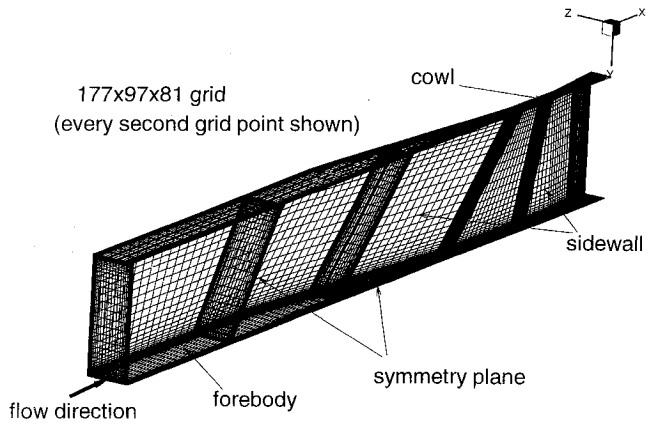

Fig. 10 Grid for hypersonic inlet.

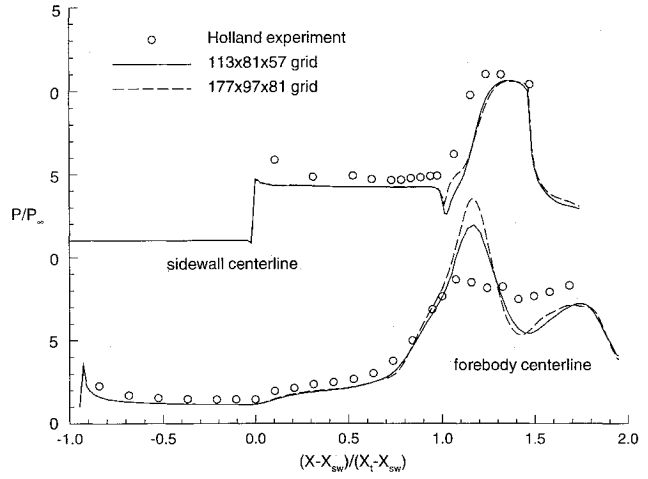

Fig. 11 Forebody and sidewall pressure distributions: hypersonic inlet.

in deviations of the type illustrated in Fig. 11. A more detailed description of the inlet flowfield, including more comparisons with experimental data and results from fully turbulent computations, may be found in Ref. 29.

The convergence of the upwind relaxation-multigrid approach for the inlet flowfield is illustrated in Fig. 12. As shown, none of the cycling strategies attain the required five order-of-magnitude residual reduction within the time allotted. Oscillatory behavior in the residual norm is eventually observed for both the V cycle iteration and the single-grid iteration (113 × 81 × 57 grid), and, as in the cylinder-cone simulation, only a twofold improvement in CPU time is achieved through multigrid acceleration. For the coarser grid, the W cycle iteration ceases to converge after about a two order-of-magnitude reduction. This behavior, not observed in any of the previous simulations, possibly indicates an inherent deficiency of the more expensive cycling strategy for flows with multiple shock interactions. For such situations, it is impossible to represent the details of the shock interactions adequately on the coarse grids, and the accuracy of the defect error calculation may suffer as a result. The W cycle will be influenced by such inaccuracies much more than the V cycle, as more work is performed on the coarser grid levels. It is quite possible that acceptable W cycle results could still be obtained through improving the high frequency damping properties of the Gauss–Seidel smoother, either by restricting the allowable time step or by optimizing the underrelaxation parameters ω_f and ω_b. Such refinements were not attempted in order to maintain a consistent comparison across the regimes of interest.

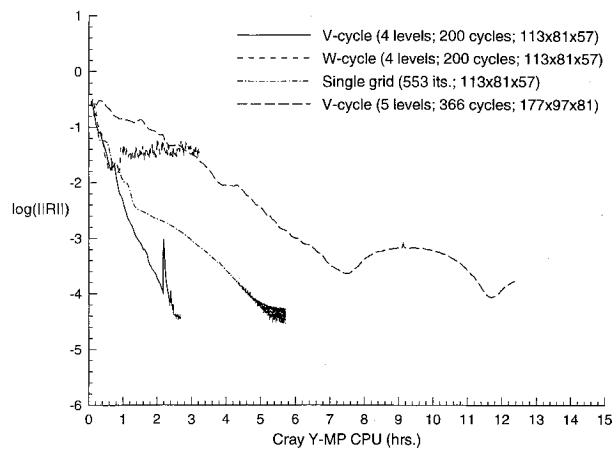

Fig. 12 Convergence histories: hypersonic inlet.

In contrast with the previous results, the V cycle convergence rate for the inlet simulation degrades significantly with grid refinement to 177 × 97 × 81 nodes, particularly near the four order-of-magnitude reduction point. For the refined-grid calculation, a pronounced hump in the residual history appears, a feature which possibly coincides with the formation and growth of a region of low-momentum, separated flow that is not captured on the coarser grid. This behavior provides an indication of the effects of flow complexity on the attaining of grid-independent rates of error reduction or, alternatively, on the attaining of any sort of steady-state convergence. The integration of powerful multiprocessor computers as routine tools for computational fluid dynamics applications will, in coming years, allow multigrid ideas to be utilized more regularly for million-grid-point simulations of truly complex, viscous flows. When this occurs, it is this author's opinion that the capturing of the inherent unsteadiness of the fluid motion by the numerics will force a reassessment of our notions of multigrid performance for Navier–Stokes calculations.

VII. Summary and Conclusions

This work has outlined the development of a multigrid Navier–Stokes solver based on upwind relaxation concepts. Four test cases, each representative of a different flight regime, have been simulated, and the effects of grid refinement and cycling strategy on the obtained rates of convergence have been examined. The results have shown that the convergence rates attained by the upwind relaxation-multigrid algorithm are rather insensitive to the adverse effects of grid AR and grid refinement. This behavior is primarily due to the use of a subiteration procedure to control factorization error growth during the planar Gauss–Seidel smoothing phases of the multigrid cycle.

For the lower-speed diffuser and nozzle flows, the W cycle strategy is superior to the V cycle in terms of both iteration count and CPU time until convergence. Compared with the single-grid procedure, the W cycle generally provides a threefold improvement in time to convergence. For the supersonic cylinder–cone and hypersonic inlet interactions, the V cycle performs best, exhibiting a twofold improvement in time to convergence relative to the single-grid iteration. The effectiveness of the W cycle iteration diminishes considerably for the hypersonic inlet flow, a possible consequence of a poor coarse-grid representation of the shock-separated flowfield.

Compared with multigrid algorithms that use simpler smoothing operators, the present approach appears somewhat deficient in terms of convergence improvement relative to the single-grid algorithm. One reason for this discrepancy relates to the use of the FMG procedure to initialize both the multigrid and the single-grid iterations. As indicated in the re-

sidual plots, a significant amount of error reduction can occur during the FMG phase of the iteration sequence. If the single-grid iteration alone were required to expel these high frequency errors, it is possible that associated time-step restrictions could severely degrade its performance. Another reason hinges on the realization that the baseline Gauss–Seidel operator itself, when equipped with the Broyden subiteration procedure, is very effective in attenuating both high- and low-frequency error components for many predominately supersonic flows. In such cases, the multigrid cycle functions primarily as a means of eliminating errors associated with the presence of thin subsonic regions.

Nevertheless, the results of this investigation, obtained on meshes as fine as $2.9M$ nodes, do indicate that the upwind relaxation-multigrid methodology is a very promising alternative to more "standard" multigrid algorithms for large-scale Navier–Stokes simulations. Issues relating to the implementation of the algorithm within the Chimera domain-decomposition framework will be addressed in the near future, and the addition of finite rate chemistry capability is currently underway.

Acknowledgments

Cray Y-MP computing time was provided courtesy of grants from the North Carolina Supercomputing Center, NASA Lewis Research Center, and NASA Langley Research Center. The support and encouragement of Meng-Sing Liou and Louis Povinelli of NASA Lewis (ICOMP) and of Mujeeb R. Malik of High Technology Corporation, Hampton, Virginia, is gratefully acknowledged.

References

[1]Thomas, J. L., and Walters, R. W., "Upwind Relaxation Algorithm for the Navier–Stokes Equations," *AIAA Journal*, Vol. 25, No. 4, 1987, pp. 527–534.

[2]Newsome, R. K., Walters, R. W., and Thomas, J. L., "An Efficient Iteration Strategy for Upwind/Relaxation Solutions to the Thin-Layer Navier-Stokes Equations," AIAA Paper 87-1113, June 1987.

[3]MacCormack, R. W., "Solution of the Navier-Stokes Equations in Three-Dimensions," AIAA Paper 90-1520, June 1990.

[4]Bardina, J., and Lombard, C. K., "Three Dimensional CSCM Method for the Compressible Navier-Stokes Equations with Application to a Multi-Nozzle Exhaust Flowfield," AIAA Paper 85-1193, July 1985.

[5]Edwards, J. R., and McRae, D. S., "Nonlinear Relaxation Navier-Stokes Solver for Three-Dimensional, High-Speed Internal Flows," *AIAA Journal*, Vol. 31, No. 7, 1993, pp. 1222–1228.

[6]Edwards, J. R., and McRae, D. S., "A Coarse-Grid Correction/Nonlinear Relaxation Algorithm for the Three-Dimensional, Compressible Navier-Stokes Equations," *Proceedings of the AIAA 11th Computational Fluid Dynamics Conference*, AIAA, Washington, DC, 1993, pp. 223-233 (AIAA Paper 93-3317).

[7]Wada, Y., and Liou, M.-S., "A Flux Splitting Scheme with High Resolution and Robustness for Discontinuities," AIAA Paper 94-0083, Jan. 1994.

[8]Benay, R., Delery, J., and Pot, T., "Experimental Analysis of the Flow Through a Three-Dimensional Transonic Channel," *Proceedings of the Propulsion and Energetics 68th Specialist's Meeting*, CP 401, AGARD, Sept. 1986.

[9]Alcenius, T. J., Schneider, S. P., Beckwith, I. E., White, J. A., and Korte, J. J., "Development of Square Nozzles for High-Speed Low-Disturbance Wind Tunnels," AIAA Paper 94-2578, June 1994.

[10]Lin, R. S., Edwards, J. R., and Malik, M. R., "Transition Control in Corner Flow of a Supersonic Square Nozzle," Final Rept., NASA SBIR NAS1-20171, High Technology Corp., Hampton, VA, July 1994.

[11]Wideman, J. K., Brown, J. L., Miles, J. B., and Ozcan, O., "Skin Friction Measurements in a 3-D, Supersonic Shock-Wave/Boundary Layer Interaction," AIAA Paper 94-0314, Jan. 1994.

[12]Edwards, J. R., and Chandra, S., "Comparison of Eddy Viscosity-Transport Turbulence Models for Three-Dimensional, Shock-Separated Flowfields," AIAA Paper 94-2275, June 1994.

[13]Gaitonde, D., Edwards, J. R., and Shang, J. S., "The Computed Structure of a 3-D Turbulent Interaction Caused by a Cylinder/Offset-Flare Juncture," AIAA Paper 95-0230, Jan. 1995.

[14]Holland, S. D., "A Computational and Experimental Investigation of a Three-Dimensional Hypersonic Scramjet Inlet Flow Field," Ph.D. Dissertation, North Carolina State Univ., Raleigh, NC, 1991.

[15]Liou, M.-S., and Steffen, C. J., "A New Flux-Splitting Scheme," NASA TM-104404, May 1991.

[16]Edwards, J. R., "Numerical Implementation of a Modified Liou-Steffen Upwind Scheme," *AIAA Journal*, Vol. 32, No. 10, 1994, pp. 2120–2122.

[17]Broyden, C. G., "A Class of Methods for Solving Nonlinear Simultaneous Equations," *Mathematics of Computation*, Vol. 19, 1965, pp. 577–593.

[18]Brandt, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Mathematics of Computation*, Vol. 31, 1977, pp. 333–390.

[19]Hackbusch, W., "Multi-Grid Methods and Applications," *Springer Series in Computational Mathematics*, Springer–Verlag, Berlin, 1985.

[20]McCormick, S. W. (ed.), "Multigrid Methods," *Frontiers in Applied Mathematics Series*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1987.

[21]Martinelli, L., Jameson, A., and Grasso, F., "A Multigrid Method for the Navier-Stokes Equations," AIAA Paper 86-0208, Jan. 1986.

[22]Vatsa, V. N., and Wedan, B. W., "Development of an Efficient Multigrid Code for 3-D Navier-Stokes Equations," AIAA Paper 89-1891, June 1989.

[23]Turkel, E., Swanson, R. C., Vatsa, V. N., and White, J. A., "Multigrid for Hypersonic Viscous Two- and Three-Dimensional Flows," AIAA Paper 91-1572, June 1991.

[24]Brandt, A., "Guide to Multigrid Development," *Lecture Notes in Mathematics 960*, Springer–Verlag, New York, 1981, pp. 220–312.

[25]Mulder, W., "A New Multigrid Approach to Convection Problems," *Journal of Computational Physics*, Vol. 83, No. 2, 1989, pp. 303–323.

[26]Koren, B., and Hemker, P. W., "Damped, Direction-Dependent Multigrid for Hypersonic Flow Computations," *Applied Numerical Mathematics*, Vol. 7, No. 4, 1991, pp. 309–329.

[27]Grasso, F., and Marini, M., "Multigrid Techniques for Hypersonic Viscous Flows," AIAA Paper 93-0771, Jan. 1993.

[28]Leclercq, M. P., and Stoufflet, B., "Characteristic Multigrid Method Application to Solve the Euler Equations with Unstructured and Unnested Grids," *Journal of Computational Physics*, Vol. 104, No. 2, 1993, pp. 329–346.

[29]Edwards, J. R., "Nonlinear Relaxation Algorithms for the Compressible Navier-Stokes Equations in Two and Three Dimensions," Ph.D. Dissertation, North Carolina State Univ., Raleigh, NC, 1992.