Electronic Transactions on Numerical Analysis.
Volume 6, pp. 106-119, December 1997.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

MULTIGRID ALGORITHM WITH CONDITIONAL COARSENING FOR THE NON-ALIGNED SONIC FLOW*

BORIS DISKIN†

Abstract. A multigrid approach using conditional coarsening in constructing solvers for non-elliptic equations on a rectangular grid is presented. Such an approach permits the achievement of a full multigrid efficiency even in the case where the equation characteristics do not align with the grid. The 2D sonic-flow equation linearized over a constant velocity field has been chosen as the model problem. An efficient FMG solver for the problem is demonstrated.

Key words. multigrid methods, conditional coarsening, sonic flow, non-alignment.

AMS subject classifications. 65N55, 76H05, 76M20.

1. Introduction.

Full-multigrid algorithms have proved to be the most efficient solvers for discretized elliptic problems. They solve a general elliptic system of discretized partial differential equations in just several *minimal work units*, where a minimal work unit is defined as the number of computer operations required for the *simplest* discretization of the problem on the target grid. This efficiency was rigorously proved for uniformly elliptic systems [5], [4]. In recent years an extensive activity has been directed at applying the same techniques to non-elliptic problems. However, these attempts have met with more limited success. Although such multigrid solvers proved to be much more efficient than comparable single-grid ones, the *textbook multigrid efficiency* — obtaining a solution in just a few minimal work units — has not been attained.

Indeed, many reported solvers require hundreds of minimal work units. Others, like various solvers based on ILU decomposition, being efficient in 2D, cannot be extended to 3D without loosing most of their efficiency. The important issue of applicability for massive parallel processing is not frequently taken into account.

A substantial preliminary step in constructing an efficient solver for a complicated problem is to detect different factors contributing to the increased amount of work and then to attack them, each one in separation from others. One of the basic difficulties inherent in non-elliptic problems is the treatment of non-alignment. This problem arises when the differential equation characteristics do not precisely coincide with grid directions.

1.1. Poor coarse-grid approximation.

In the presence of non-alignment the standard multigrid cycle with full coarsening (the meshsizes are doubled in all the directions in passing from fine to coarse grid) suffers from poor coarse-grid approximation for some *smooth* fine-grid error components. As a consequence the FMG algorithm employing such a cycle loses its efficiency. This trouble has been realized and treated for convection-diffusion problem (see [1]) for high-Reynolds incompressible entering flows (see [7]) and for sonic flows (see [9]).

A simple explanation for the problem can be given in the case when all the characteristics of the differential equation emanate from the boundary. In that case the quality of the coarse-grid correction is defined by how well certain cross-characteristic oscillations are advected on the coarse grid from the inflow boundary into the domain.

Any discretization of a non-aligned non-elliptic (or weakly elliptic) problem unavoidably introduces some cross-characteristic numerical viscosity which is absent in the original

* Received May 14, 1997. Accepted June 26, 1997. Communicated by J. Dendy.
† Department of Applied Mathematics and Computer Science, The Weizmann Institute of Science, Rehovot, Israel.

ETNA
Kent State University
etna@mcs.kent.edu

differential problem. This viscosity determines the "penetration distance" of incoming oscillations. The magnitude of the viscosity is usually proportional to some power of the grid meshsize and therefore differs from grid to grid. In the case of full coarsening the coarse-grid cross-characteristic viscosity is much larger than that of the fine grid, which causes increased decay and phase shifting of the cross-characteristic oscillations on the coarse grid.

The idea suggested in [9] to overcome this trouble is to use *semi-coarsening* together with introducing a well-balanced *explicit numerical viscosity* on coarse grids to control the penetration of incoming cross-characteristic oscillations. The subject of the present research is to modify the algorithm of [9], making it somewhat cheaper and to exhibit another strategy for treating problems with variable couplings.

1.2. Sonic flow equation. The *sonic flow equation* we have chosen (similar to [9]) to be our model problem is the semi-elliptic equation

$$\tilde{\triangle}\Phi = f, \tag{1.1}$$

where $\tilde{\triangle}\Phi$ is the Laplacian on a lower dimensional manifold. We call such a manifold *a characteristic manifold*.

We study the phenomenon of non-alignment in the 2D constant coefficient case where the characteristic manifold is a straight line (*a characteristic line*). Although relatively simple, the problem nevertheless contains one of the main difficulties appearing in flows not consistently aligned with the discretization grid. The approach we borrow from [9] is to use a fixed Cartesian coordinate system independent of the characteristic plane. We choose the *x* axis to be *the reference axis*. Thus each characteristic line is uniquely defined by a point on it and its inclination to the reference axis. We restrict ourselves to characteristic lines whose inclinations are less than or equal to $\pi/4$ (45^o), in other words to lines that can be written as $\alpha x + \beta y = \delta$, where $|\beta| \le |\alpha|$. Otherwise the role of the axes should be interchanged.

1.3. Discretization. The principles of discretizing semi-elliptic equations were formulated in [9]. Briefly, they contain the following two stages:

1) Development of an *h-elliptic discretization* (see §2.1 in [3] or [2]) on a characteristic manifold. The auxiliary grid is defined by means of the *ghost points* placed at the intersections of the manifold with vertical grid lines. This lower-dimensional discrete operator is called the *low-dimensional prototype*.

2) In the real dimension one should consider the *low-dimensional prototype* locally at each grid point. The discretization is obtained by replacing values at the ghost points with values linearly interpolated from the neighboring genuine grid points placed on the same vertical grid line, together with the addition of several compensating points on the vertical grid line going through the point where the discrete operator is being defined. This discretization is h-elliptic in the full dimension.

Such a discretization possesses numerical viscosity because of the non-alignment and the resulting interpolation. We will call it the *inherent numerical viscosity*, to distinguish it from the *explicit numerical viscosity* introduced below. Quantitatively, the inherent numerical viscosity is defined as the coefficient of the pure cross-characteristic derivative arising in the *first differential approximation* (FDA) to the discrete operator (see [13]), the cross-characteristic direction being defined as the direction perpendicular to the characteristic manifold.

Following [9] we introduce some useful notation. A function defined on the space under consideration will be referred to as a *characteristic component* if it is a very smooth function on the characteristic manifold. The term *high-frequency characteristic component* will refer to a characteristic component that is highly-oscillating in the *cross-characteristic* direction.

Similarly the term *smooth characteristic component* will denote a characteristic component that is smooth in the *cross-characteristic* direction (but not as smooth as in any of the characteristic directions).

Previous studies on several types of non-elliptic equations (see [1], [7] and [9]) have shown that the basic trouble in constructing an efficient multigrid solver is the poor approximation of smooth characteristic components on coarse grids. The reason is the increased coarse-grid inherent numerical viscosity appearing in cycles with full coarsening. A general way to overcome this trouble would be to use *semi-coarsening*, with meshsizes being doubled only in the reference directions. When applied in its pure form, semi-coarsening also results in some difficulties, since the inherent numerical viscosity of the semi-coarsened grid will be much less than that of the fine grid. But we can supply the operator on the semi-coarsened grid with an additional term (explicit numerical viscosity term) so that the total viscosity on the semi-coarsened grid is the same as on the fine grid.

The three- or even four-level version of such a cycle, with two pointwise relaxations on each level and appropriate inter-grid transfers, can already be used to solve efficiently the model problems discretized on a uniform target grid. However, the implementation of a cycle with more levels raises a new difficulty.

1.4. Strong cross-characteristic coupling. The inherent numerical viscosity in our algorithm arises from the vertical interpolation to the ghost points. To obtain the same total viscosity, we introduce an explicit numerical viscosity on the coarse grids by adding a term which is a discrete approximation to a vertical derivative of suitable order.

The multigrid theory of h-elliptic discrete operators (see [3], [1]) shows that a pointwise relaxation can reduce only the error components that oscillate in the strong-coupling directions. A simple coupling analysis of the discretization used below shows that the *target-grid* discrete-operator direction of stronger coupling approximately coincides with the characteristic manifold. Thus a target-grid pointwise relaxation can reduce efficiently the non-characteristic error components and also some of the high-frequency characteristic components of the error. That is all we need from the relaxation since the smooth characteristic components (and most of the high-frequency characteristic components) are well reduced on the next *semi-coarsened* grids. However, successive semi-coarsening implies a fast decrease in the *inherent numerical viscosity* on the coarse grids and hence a fast increase in the weight of the compensating explicit numerical viscosity in the coarse-grid discrete operator. Thus the direction of the strongest coupling after several semi-coarsening steps tends to be vertical; hence any pointwise relaxation on such coarse grids will not reduce efficiently some *non-characteristic* components of the error.

One can perform a simple coupling analysis to distinguish two directions: the first is the characteristic direction and the second is the viscosity action direction (vertical direction in our case). When the "viscous" coupling dominates over the "characteristic" one the approach above loses its efficiency. The way suggested in [9] to overcome this difficulty was to use a "conditional relaxation" technique, meaning that one should switch from pointwise to vertical line relaxation starting from a grid where the viscous coupling is strong enough. In the present paper we test a "conditional coarsening" technique — to switch to full coarsening whenever it is possible. It is clear that we would like to use full coarsening as much as possible, since it is cheaper. The main hindrance is the increased inherent numerical viscosity on coarse grids. (An explicit numerical viscosity term with a sign opposite to that of the inherent numerical viscosity can cause instabilities in the discretization and therefore its appearance is undesirable.) But if the full coarsening step follows a semi-coarsening one (which, in turn, decreases the inherent numerical viscosity) then the inherent numerical viscosity on the grid obtained by the full coarsening is already about the same as on the target grid. Thus one could use full coarsening for every other coarsening step. However, in practice, we need a more delicate criterion for the switching since the inherent numerical viscosity depends on the characteristic inclination; therefore the rule above cannot be applied automatically.

Another more accurate criterion, derived from the coupling analysis and confirmed in the two-level experiments, will be exhibited in the next section.

2. Non-aligned Sonic Flow.

2.1. Problem statement. We consider the simple equation

$$\frac{\partial^2 \Phi}{\partial \xi^2} = F, \tag{2.1}$$

where the characteristic direction is $\xi = (x + ty)(1 + t^2)^{-1/2}$ and $t = \tan\psi$ is the tangent of the *angle of non-alignment*, i.e., the angle between the characteristic direction and the reference $x-$axis. We require $|t| \leq 1$. The unknown scalar function $\Phi(x, y)$ is defined on the square $(x, y) \in [0, 1] \times [0, 1]$

Eq. (2.1) is supplied with Dirichlet boundary conditions in the x direction and periodic conditions in the y direction:

$$\Phi(0, y) = g_0(y), \;\; \Phi(1, y) = g_1(y), \;\; \Phi(x, y) = \Phi(x, y + 1), \tag{2.2}$$

where $g_0(y)$ and $g_1(y)$ are given functions.

The nine-point discretization of (2.1) on a grid with aspect ratio $m = h_x/h_y$, where h_x and h_y are the meshsizes in the x and y directions, respectively, is defined by

$$\begin{array}{l} L^{(h_x,h_y)}\phi_{i_1,i_2} \equiv \\ \frac{1}{h_x^2+(k+s)^2h_y^2}\left[(1-s)\left(\phi_{i_1-1,i_2-k} + \phi_{i_1+1,i_2+k}\right)\right. \\ \quad +s\left(\phi_{i_1-1,i_2-(k+1)} + \phi_{i_1+1,i_2+(k+1)}\right) \\ \quad \left.-2\phi_{i_1,i_2} - s(1-s)\left(\phi_{i_1,i_2-1} - 2\phi_{i_1,i_2} + \phi_{i_1,i_2+1}\right)\right] \\ \quad -A\frac{1}{h_y^2}\left[\phi_{i_1,i_2+2} - 4\phi_{i_1,i_2+1} + 6\phi_{i_1,i_2} - 4\phi_{i_1,i_2-1} + \phi_{i_1,i_2-2}\right]. \end{array} \tag{2.3}$$

Here ϕ_{i_1,i_2} is a discrete approximation to $\Phi(i_1h_x, i_2h_y)$; $k + s = mt$, k is integer and $0 \leq s < 1$ (see Figure 2.1); A is the *explicit* numerical viscosity coefficient. Thus the differential problem (2.1)–(2.2) is discretized on the grid as

$$\begin{array}{rl} L^{(h_x,h_y)}\phi_{i_1,i_2} = f_{i_1,i_2}, & i_1 = 1, \ldots, n_1 - 1 \\ \phi_{0,i_2} = g_0(i_2h_y), & \\ \phi_{n_1,i_2} = g_1(i_2h_y), & \\ \phi_{i_1,i_2+n_2} = \phi_{i_1,i_2}, & i_1 = 0, 1, \ldots, n_1 \end{array} \tag{2.4}$$

where $n_1 = 1/h_x, \; i_2 \in \mathbf{Z}, \; n_2 = 1/h_y$ and $f_{i_1,i_2} = F(i_1h_x, i_2h_y)$.

The first differential approximation (cf. [13], [1]) to the operator (2.3) is

$$\phi^h_{\xi\xi} - h_y^2\left[A + \left(\frac{(1-s)s\cos(\psi)}{2m}\right)^2\right]\phi_{yyyy}, \tag{2.5}$$

where $\phi^h_{\xi\xi}$ is the first differential approximation to the one-dimensional prototype. For characteristic components $\phi^h_{\xi\xi} \approx \phi_{\xi\xi}$. On the target grid $A = 0$, and on coarser grids A is

ETNA
Kent State University
etna@mcs.kent.edu

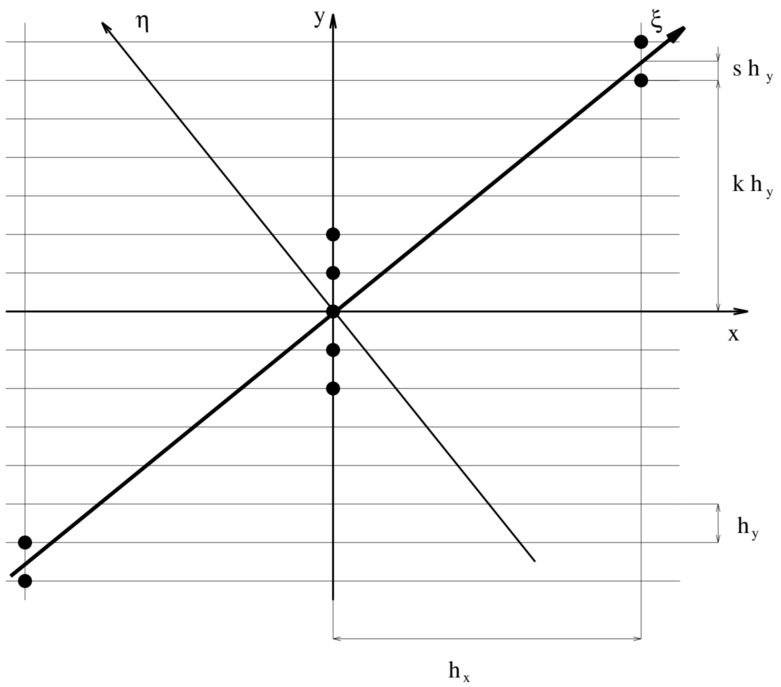

FIG. 2.1. *Anisotropic grid; nine point stencil.*

chosen so that the *total cross-characteristic viscosity*, i.e., the coefficient of ϕ_{yyyy} in (2.5), remains the same as on the target grid. Note that upon a semi-coarsening step the values of h_y and ψ remain unchanged while $s(1-s)/m$ decreases, i.e., the inherent numerical viscosity decreases. On the other hand upon a full coarsening step h_y increases, while all the other parameters remain the same, hence, the inherent numerical viscosity increases. Note also that the true cross-characteristic viscosity should be defined as the coefficient of the fourth derivative with respect to η, where $\eta = (-tx+y)(1+t^2)^{-1/2}$ is the variable along the cross-characteristic direction, but for the characteristic components in the constant coefficient case that is just proportional to the y-directional viscosity.

2.2. Multigrid cycles and coarsening policy. In this section we first present a two-level cycle employing either semi- or full coarsening. The basic parts of the cycle, such as relaxation, residual transfer and correction interpolation are described. The two-level numerical tests together with the coupling analysis discussed here allow us to construct an efficient multigrid cycle, which is examined at the end of this section.

2.2.1. Relaxation. The relaxation to be used in the algorithm is the eight-color Gauss-Seidel relaxation. The elementary step of this relaxation is to change the solution approximation at the point (i_1, i_2) to satisfy Eq. (2.4). The order of performance of the elementary step obeys the following rules.

1) The odd vertical lines (the vertical lines with odd i_1 coordinate) are relaxed before the even ones.

2) The relaxation in each vertical line consists of four sweeps. Each sweep performs the elementary step for every fourth point on the line. The first sweep starts from the point with vertical coordinate $i_2 = 0$, the second — from the point with $i_2 = 2$, the third — from the point with $i_2 = 1$ and the last — from the point with $i_2 = 3$.

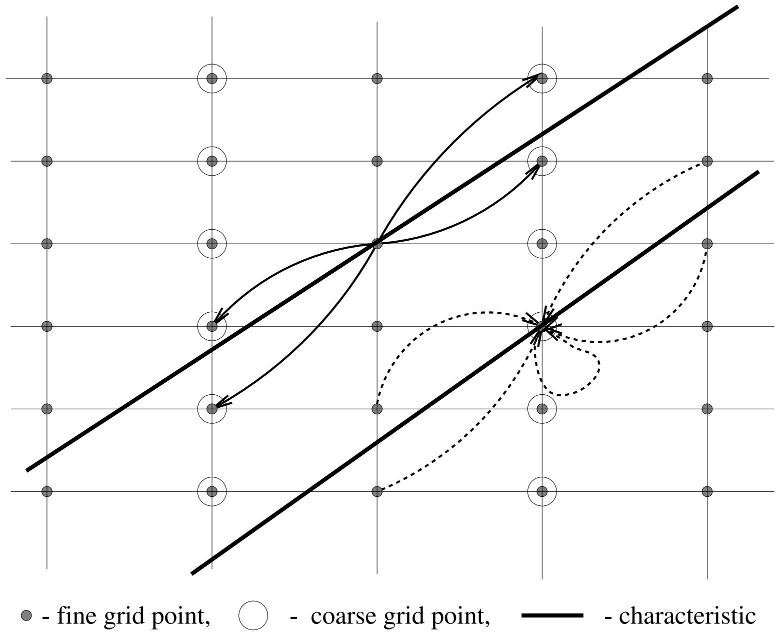

FIG. 2.2. *Residual transfer to semi-coarsened grid.*

It is a quite efficient and fully parallelizable relaxation. This order of relaxation is not necessary for efficient smoothing. It is chosen to enable full parallelization and preclude the appearance of relaxation "boundary layers". To be sure, the red-black relaxation scheme would be efficient as well, but then the results would slightly depend on *where* the sweeps start and end, which we wanted to avoid.

2.2.2. Inter-grid transfers. The inter-grid communication within any cycle consists of two types of transfers. The fine-to-coarse transfer forms the coarse-grid approximation to the fine-grid residual function

$$r_{i_1}^{i_2} = f_{i_1}^{i_2} - L^{(h_x,h_y)}\phi_{i_1}^{i_2}.$$

The coarse-to-fine transfer is the interpolation of the coarse-grid correction. In our cycle both transfers are anisotropic. The distinguished direction tends to coincide with the characteristic one.

Residual transfer to a semi-coarsened grid is given by

$$\begin{aligned} &R_{i_1,i_2} = \\ &\left(I_h^H r\right)_{i_1,i_2} = .5\; r_{2i_1,i_2} + .25\Big[(1-s)\Big(r_{2i_1-1,i_2-k} + r_{2i_1+1,i_2+k}\Big) \\ &+s\Big(r_{2i_1-1,i_2-k-1} + r_{2i_1+1,i_2+k+1}\Big)\Big], \end{aligned} \tag{2.6}$$

where R and r denote the coarse and fine grid residual functions, respectively.

The scheme is described in Figure 2.2. The solid arrows show where a fine-grid point residual is sent to. The dashed arrows exhibit all the fine-grid points sending their residuals to a given coarse-grid point.

ETNA
Kent State University
etna@mcs.kent.edu

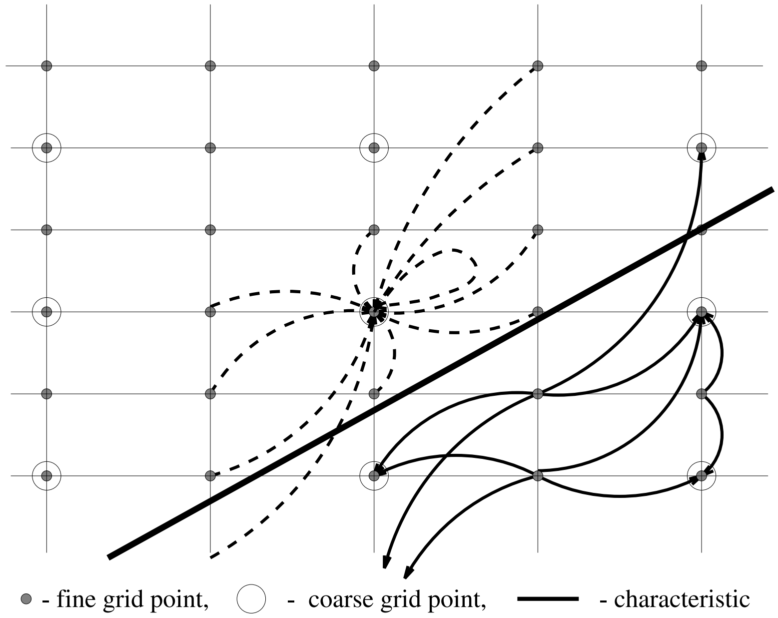

FIG. 2.3. *Residual transfer to fully coarsened grid.*

Correction interpolation from a semi-coarsened grid is given by the operator adjoint to (2.6),

$$\left\{ \begin{array}{lcl} v_{2i_1,i_2} & = & V_{i_1,i_2}, \\ v_{2i_1+1,i_2} & = & .5\Big[\Big(1-S_1\Big)\Big(V_{i_1,i_2-K_1}+V_{i_1+1,i_2+K_1}\Big) \\ & & +S_1\Big(V_{i_1,i_2-K_1-1}+V_{i_1+1,i_2+K_1+1}\Big)\Big], \end{array} \right.$$

where V is the solution of the coarse-grid problem, v denotes the correction to the fine-grid solution approximation and K_1 is an integer such that $(K+S)/2 = K_1+S_1, 0 \leq S_1 < 1, K$ and S being the parameters of the coarse-grid discretization (defined like k and s in Fig. 2.1).

Residual transfer to a fully coarsened grid is defined by

$$\begin{aligned} R_{i_1,i_2} = \; & \frac{1}{4} r_{2i_1,2i_2} + \frac{1}{8}\Big(r_{2i_1,2i_2-1} + r_{2i_1,2i_2+1}\Big) \\ & + \frac{1}{16}\Big[(1-s)\left(r_{2i_1-1,2i_2-(k-1)} + r_{2i_1+1,2i_2+(k-1)}\right) \\ & + \quad (1+s)\left(r_{2i_1-1,2i_2-(k+1)} + r_{2i_1+1,2i_2+(k+1)}\right)\Big] \\ & + \frac{1}{16}\Big[(2-s)\left(r_{2i_1-1,2i_2-k} + r_{2i_1+1,2i_2+k}\right) + s\left(r_{2i_1-1,2i_2-(k+2)} + r_{2i_1+1,2i_2+(k+2)}\right)\Big]. \end{aligned}$$

The map of this transfer is shown in Fig. 2.3. The notations remain the same: dashed arrows show the fine-grid points contributing to a given coarse-grid point residual, solid arrows show where a given fine-grid point residual is sent to.

Correction interpolation from a fully coarsened grid is described in the following formulas

$$\left\{\begin{array}{lcl} v_{2i_1}^{2i_2} & = & V_{i_1}^{i_2}, \\ v_{2i_1+1}^{2i_2} & = & .5\left[(1-S_1)\left(V_{i_1}^{i_2-K_1}+V_{i_1+1}^{i_2+K_1}\right)+S_1\left(V_{i_1}^{i_2-(K_1+1)}+V_{i_1+1}^{i_2+(K_1+1)}\right)\right], \\ v_{2i_1}^{2i_2+1} & = & .5\left[\left(V_{i_1}^{i_2}+V_{i_1}^{i_2+1}\right)\right], \\ v_{2i_1+1}^{2i_2+1} & = & .5\left[(1-S_2)\left(V_{i_1}^{i_2+1-K_2}+V_{i_1+1}^{i_2+K_2}\right)+S_2\left(V_{i_1}^{i_2+1-(K_2+1)}+V_{i_1+1}^{i_2+(K_2+1)}\right)\right], \end{array}\right.$$

where K_1 and K_2 are integers, $K_1 + S_1 = (K + S)/2,\ K_2 + S_2 = (K + S + 1)/2,\ 0 \le S_1, S_2 < 1$; K and S are the coarse-grid discretization parameters (in the case of full coarsening they are in fact the same as k and s of the fine-grid discretization).

2.2.3. Coarsening policy and numerical experiments. As mentioned in Section 1.4, the criterion whether to use full or semi- coarsening can be derived from a coupling analysis of the FDA approximation (2.5). The "characteristic" coupling is determined by $\phi^h_{\xi\xi}$ and a quantitative measure of this coupling is

$$h_\xi^{-2} = \left(\left(m^2 + (k+s)^2\right)h_y^2\right)^{-1}.$$

The "viscous" coupling is maintained by the second term in (2.5) in which ϕ_{yyyy} has the coupling strength h_y^{-4}. We can switch to full coarsening when this "viscous" coupling becomes comparable with the "characteristic" one, i.e., when the ratio between them, which we call the *Relative Coupling (RC)*, becomes close to one. In practice one should switch to full coarsening already when RC $> .5$, since otherwise the viscous coupling on the next semi-coarsened grid will be too strong, making pointwise relaxation there somewhat inefficient. The derived criterion completely agrees with the experimental comparison between the asymptotic convergence rate of the two-level cycles employing full vs. semi-coarsening.

The two-level cycle $V_2(\nu_1, \nu_2)$ can formally be defined by the following six steps.

(i) *Pre-relaxation sweeps*. Improve the initial fine-grid approximation by ν_1 relaxation sweeps

(ii) *Residual transfer*. Form on the coarse grid the approximation R to the fine-grid residual function.

(iii) *Coarse-grid equation*. Form the coarse-grid equation

$$L^{(H_x,H_y)}V = R.$$

The discrete operator $L^{(H_x,H_y)}V$ depends on the coarse-grid discretization parameters K and S which should be recalculated by means of the coarse-grid aspect ratio. The value of the coarse-grid explicit numerical viscosity parameter A is chosen so that the coarse-grid total viscosity is exactly the same as on the fine grid.

(iv) *Exact solution*. Solve the coarse-grid equation by whatever method.

(v) *Coarse-grid correction*. Interpolate the obtained coarse-grid solution to the fine grid to correct the current fine-grid approximation.

(vi) *Post-relaxation sweeps*. Improve the corrected fine-grid approximation by ν_2 relaxation sweeps

We have run the two-level cycles with either full or semi-coarsening on grids with aspect ratios $m = 1, 2, 4, 8, 16$ for different characteristic inclinations. We chose zero right hand side and zero boundary conditions, so that $U(x, y) \equiv 0$ is the exact solution of the differential problem (2.1-2.2). This choice of data together with random initial error facilitates observing the cycle asymptotic behavior. The fine-grid explicit numerical viscosity parameter A value has been calculated under the assumption that the fine grid itself was obtained by ($\log_2 m$ steps of) semi-coarsening starting with a uniform target grid. In other words the total numerical viscosity in the cycle was chosen to be equal to the inherent numerical viscosity of the uniform grid with meshsize h_y.

Within each experiment we have performed two runs of the $V_2(1,1)$ cycle, starting each time from a random initial error. The first run was done for the cycle employing semi-coarsening and the second — for the cycle with fully coarsened coarse grids. Each run consisted of at least 12 cycles, stopping further cycling if the maximal difference in the convergence factors of the three last cycles did not exceed 0.01. The results of these experiments are exhibited in Table 1.

This table uses the following notation: h_y is the vertical meshsize; m is the fine-grid aspect ratio; $t = \tan\psi$ is the inclination parameter, where ψ is the angle between the characteristic direction and the reference x axis; ν_1 and ν_2 are cycle parameters. The column "cycles" shows the number of cycles performed until the convergence factor has been stabilized. In the column "final" the convergence factor of this last cycle (the L_2 error norm before the cycle divided by that norm after the cycle) is printed. The column "aver." exhibits the convergence factor averaged over *all* the cycles performed in the run.

The comparison of the results for different types of coarsening shows that for RC $\leq .25$ the only efficient cycle is that employing semi-coarsening; for .25 $<$ RC $<$ 2 the results are satisfactory for both types of coarsening and for 2 $<$ RC both cycles are inefficient. The results mean that the threshold value of the relative coupling to switch between semi- and full coarsening should be $.25 \leq$ RC $\leq .5$. We choose RC = .5, to separate the cases of different coarsening.

Thus, the *multilevel* cycle employing *conditional coarsening* obeys the following rule: if on the current grid RC $\leq .5$ then the grid is next semi-coarsened; otherwise full coarsening is used. We performed experiments with a 6-level $V_6(1,1)$ cycle using conditional coarsening. This cycle can formally be described similar to the two-level cycle description, except that step (iv) is replaced by a recursive call to the same cycle applied to the coarse-grid problem. These experiments show a stable asymptotic convergence with slightly slower rate than in the two-level experiments. Table 2 compares the results of conditional coarsening cycle with those of a cycle using the conditional relaxation technique (see [9]). The conditional coarsening cycle seems to be a bit less efficient, but it is cheaper and this can be significant in the case when the target grid discretization possesses a relatively strong viscous coupling. (But not as strong as RC > 2; in that case the adequate coarse grid is obtained by semi-coarsening in the *vertical* (viscous coupling) direction.)

2.3. FMG solver. The FMG solver based on the $V(1,1)$ cycle using conditional coarsening possesses *textbook multigrid efficiency*. Its *setup* work can be described by the following steps.

1. *Target-grid problem.* We formulate the discrete equation (2.3) on the chosen target grid. The parameter A for this grid is set to zero. The total viscosity value for the entire algorithm is defined as this target-grid inherent numerical viscosity. A proper discretization of the right-hand side f and the boundary condition functions g_0 and g_1 is also performed. In our implementation these discrete functions are simply injected from the corresponding continuous ones.

2. *Next coarse-grid construction.* The next coarse grid is constructed by either semi- or full coarsening, depending on the current-grid relative coupling value in the same way as in the cycle described above.

3. *Coarse-grid problem.* The discretization parameters such as the aspect ratio, the new

K, S parameters and the explicit numerical viscosity coefficient A are calculated for the new grid. The general form of the coarse-grid operator remains the same. The coarse-grid right-hand side function F is formed by the same averaging procedure that is used for the residual transfer inside the cycles, i.e.,

$$F_{i_1,i_2} = .5\ f_{2i_1,i_2} + .25\Big[(1-s)\Big(f_{2i_1-1,i_2-k} + f_{2i_1+1,i_2+k}\Big) + s\Big(f_{2i_1-1,i_2-k-1} + f_{2i_1+1,i_2+k+1}\Big)\Big]$$

for semi-coarsening and

$$\begin{aligned} F_{i_1}^{i_2} = \quad & \frac{1}{4} f_{2i_1}^{2i_2} + \frac{1}{8}\left(f_{2i_1}^{2i_2-1} + f_{2i_1}^{2i_2+1}\right) \\ & + \frac{1}{16}\left[(1-s)\left(f_{2i_1-1}^{2i_2-(k-1)} + f_{2i_1+1}^{2i_2+(k-1)}\right) + (1+s)\left(f_{2i_1-1}^{2i_2-(k+1)} + f_{2i_1+1}^{2i_2+(k+1)}\right)\right] \\ & + \frac{1}{16}\left[(2-s)\left(f_{2i_1-1}^{2i_2-k} + f_{2i_1+1}^{2i_2+k}\right) + s\left(f_{2i_1-1}^{2i_2-(k+2)} + f_{2i_1+1}^{2i_2+(k+2)}\right)\right] \end{aligned}$$

for full coarsening. The coarse-grid boundary conditions are *injected* from the previous fine grid (averaging could as well be used).

Steps 2 and 3 are repeated until the coarsest possible grid is reached and its problem is defined.

The *execution* of the FMG algorithm then involves the following steps:

$a)$ The coarsest-grid problem is solved by whatever method.

$b)$ The solution obtained on the current grid is interpolated to the next fine grid to serve as an initial approximation to the fine-grid solution. The "FMG interpolation" used in this step is the fourth order in the characteristic direction and the second order in the vertical direction. (In the vertical direction we may also need a higher order interpolation, especially for the full coarsening step, but the experiments show that even with this lower order vertical interpolation the algorithm successfully reduces the algebraic errors well below the level of the discretization errors.)

$c)$ The obtained initial approximation is improved by one V(1,1) cycle.

We repeat the steps $b)$ and $c)$ up to the target grid. There we perform one additional improving cycle (mainly for checking purposes).

We have performed numerical experiments with a six-level FMG algorithm. The target finest grid is always a uniform grid with meshsize $h_y = h_x = h = 2^{-7}$. The right-hand side function and boundary conditions of the differential problem (2.1)–(2.2) are chosen so that the continuous function $\sin(\theta x + \omega y)$ is the exact solution. For all types of components we check four different characteristic slopes $t = \tan\psi$. The results are collected in Table 3 where we compare the target-grid discretization error with the algebraic errors after the FMG interpolation to the target grid and at the end of the first and the second improving cycles. For nearly all the examined components the algebraic error after the first cycle is much less than the discretization error. In fact, in the case of characteristic components the algebraic error is at the level of the discretization error already after the FMG interpolation. For non-characteristic components, one $V(1,1)$ target-grid cycle is enough to obtain an approximation possessing the discretization accuracy.

We can thus conclude that the FMG algorithm requires only one $V(1,1)$ cycle per FMG level, or a total of about 13 "minimal work units", to reach the discretization accuracy for the target-grid approximation.

2.4. Notes about 3D sonic flow. Experiments performed with the 3D sonic flow solver confirm that this approach is well extendible to 3D, keeping the textbook multigrid efficiency.

(In fact, this 3D solver requires just *six* "minimal work units" to get the target-grid approximation, with an error which is less than the target-grid discretization error.) One can find the numerical test results in [10]. We should also emphasize that the work saving of the conditional coarsening algorithm in comparison with the conditional relaxation algorithm from [9] is less important in 3D since the 3D semi-coarsening has two coarsening directions and, therefore, conditional relaxation algorithm is extremely cheap as well.

Nevertheless, one can imagine cases when the conditional coarsening approach can give a real gain. For example, 3D semi-elliptic problems on a one-dimensional characteristic manifold (e.g. convection) fall into this category.

REFERENCES

[1] A. BRANDT, *Multigrid solvers for non-elliptic and singular-perturbation steady-state problems*, The Weizmann Institute of Science, Rehovot, Israel, December 1981 (unpublished).

[2] ———, *Guide to multigrid development*, in Multigrid Methods, Lecture Notes in Math, 960, W. Hackbusch and U. Trottenberg, eds., Springer-Verlag, 1982, pp. 220-312.

[3] ———, *Multigrid techniques: 1984 guide with applications to fluid dynamics*, Monograph, GMD-Studie 85, GMD-FIT, Postfach 1240, D-5205, St. Augustin 1, W. Germany, 1985 (unpublished). (Also available from Secretary, Department of Mathematics, University of Colorado at Denver, Colorado 80204-5300.)

[4] ———, *Rigorous quantitative analysis of multigrid, I: Constant coefficients two-level cycle with L2-norm*, SIAM J. Num. Anal., 31(1994), pp 1695-1730.

[5] ———, *Rigorous local mode analysis of multigrid*, in Preliminary Proc. 4th Cooper Mountain Conf. on Multigrid Methods, Cooper Mountain, Colorado, April 1989.

[6] A. BRANDT AND I. YAVNEH, *Inadequacy of first-order upwind difference schemes for some recirculating flows*, J. Comput. Phys, 93(1991), pp. 128–143.

[7] ———, *On multigrid solution of high–Reynolds incompressible entering flow*, J. Comput. Phys., 101(1992), pp. 151-164.

[8] ———, *Accelerated multigrid convergence and high-Reynolds recirculating flows*, SIAM J. Sci. Comput., 14(1993) pp. 607-626.

[9] A. BRANDT AND B. DISKIN, *Multigrid solvers for the non-aligned sonic flow: the constant coefficient case.*, to appear, Comput. & Fluids.

[10] B. DISKIN, *Multigrid solver for potential-flow equation*, Intermediate report on research leading to the Ph. D. degree, The Weizmann Institute of Science, Rehovot, Israel, November,1995

[11] J. C. SOUTH AND A. BRANDT, *Application of a multi-level grid method to transonic flow calculations*, Transonic Flow Problems in Turbomachinery, ICASE Report 76-8, T. C. Adamson and M. F. Platzer, eds. Hemisphere, Washington, 1977, pp. 180–207.

[12] A.J. VAN DER WEES, *FAS multigrid employing ILU/SIP smoothing: a robust fast solver for 3D transonic potential flow*, in Multigrid Methods II, Proceedings 2nd European Conference on Multigrid Methods, Cologne, 1985, W. Hackbusch and U. Trottenberg, eds., Springer-Verlag, pp. 316-331.

[13] N.N. YANENKO AND Y.I.SHOKIN, *Correctness of first differential approximations of difference schemes*, Dokl. Akad. Nauk SSSR, 182(1968), pp. 776-778.

[14] I. YAVNEH, *A method for devising efficient multigrid smoothers for complicated PDE system*, SIAM J. Sci. Comput., 14(1993), pp. 1437-1463.

[15] I. YAVNEH, C.H. VENNER, AND A. BRANDT, *Fast multigrid solution of the advection problem with closed characteristics*, SIAM J. Sci. Comput., 19 (1998).

3. Tables of Numerical Results.

Table 1. Multigrid solver of the eqn. $\frac{\partial^2 U}{\partial \xi^2} = f$ in 2D.

2 level algorithm;

Exact continuous solution: $U \equiv 0$

						asymptotic convergence rate					
					relative	semi coarsening			full coarsening		
h_y	m	t	ν_1	ν_2	coupling	cycles	final	aver.	cycles	final	aver.
0.03125	1	0.100	1	1	0.00203	20	4.21	13.8	14	1.1	2.73
0.03125	1	0.300	1	1	0.01103	20	3.44	5.54	20	1.33	2.43
0.03125	1	0.500	1	1	0.01562	21	4.99	6.76	17	1.4	2.73
0.03125	1	0.700	1	1	0.01103	18	3.48	5.76	23	1.32	2.24
0.03125	1	0.900	1	1	0.00202	13	4.18	19.6	16	1.09	2.6
0.01562	2	0.100	1	1	0.00810	19	3.39	6.99	17	1.22	2.48
0.01562	2	0.300	1	1	0.04410	23	6.05	8.94	18	1.95	3.3
0.01562	2	0.500	1	1	0.06250	33	13.3	14.8	32	1.6	2.23
0.01562	2	0.700	1	1	0.04410	28	6.09	8.44	16	1.94	3.58
0.01562	2	0.900	1	1	0.00810	20	3.39	6.73	15	1.24	2.65
0.00781	4	0.100	1	1	0.03240	26	5.35	7.36	17	1.74	3.15
0.00781	4	0.300	1	1	0.17640	30	17.8	20.9	18	3.8	4.88
0.00781	4	0.500	1	1	0.25000	28	20.9	22.2	24	3.05	4.1
0.00781	4	0.700	1	1	0.17640	35	17.8	20.4	24	3.79	4.66
0.00781	4	0.900	1	1	0.03240	24	5.38	7.52	17	1.7	3.1
0.00391	8	0.100	1	1	0.12960	28	15.9	19.3	20	2.99	4.15
0.00391	8	0.300	1	1	0.70560	31	6.19	6.65	25	6.33	6.79
0.00391	8	0.500	1	1	1.00000	15	3.7	4.3	16	3.7	4.24
0.00391	8	0.700	1	1	0.70560	30	6.18	6.65	27	6.31	6.8
0.00391	8	0.900	1	1	0.12960	30	15.9	19.3	23	3.01	4.08
0.00195	16	0.100	1	1	0.51840	36	8.86	9.59	37	8.84	9.35
0.00195	16	0.300	1	1	2.82240	15	1.64	2.4	14	1.64	2.47
0.00195	16	0.500	1	1	4.00000	15	1.43	2.35	15	1.43	2.34
0.00195	16	0.700	1	1	2.82240	15	1.63	2.39	14	1.64	2.46
0.00195	16	0.900	1	1	0.51840	42	8.87	9.55	35	8.84	9.35

ETNA
Kent State University
etna@mcs.kent.edu

Table 2. Multigrid solver of the eqn. $\frac{\partial^2 U}{\partial \xi^2} = f$ in 2D.

6 level algorithm;

Exact continuous solution: $U \equiv 0$

					asymptotic convergence rate					
					conditional coarsening			conditional relaxation		
h_y	m	t	ν_1	ν_2	cycles	final	aver.	cycles	final	aver.
0.01562	1	0.100	1	1	15	2.1	14.2	12	2.23	25.7
0.01562	1	0.300	1	1	21	3.09	5.3	24	3.19	5.82
0.01562	1	0.500	1	1	31	4.13	5.65	26	5.01	7.41
0.01562	1	0.700	1	1	21	3.1	5.2	28	3.16	5.33
0.01562	1	0.900	1	1	14	2.09	13.5	13	2.22	23
0.00781	2	0.100	1	1	26	2.66	5.4	20	2.71	7.09
0.00781	2	0.300	1	1	33	4.07	6.18	27	5.82	8.95
0.00781	2	0.500	1	1	21	4.45	7.8	22	10.9	16.1
0.00781	2	0.700	1	1	26	4.15	6.79	24	5.81	9.17
0.00781	2	0.900	1	1	27	2.66	5.27	26	2.72	5.95
0.00391	4	0.100	1	1	25	3.99	6.36	34	5.07	7.33
0.00391	4	0.300	1	1	21	4.15	7.21	26	8.17	24.6
0.00391	4	0.500	1	1	32	4.37	5.66	29	11	14.6
0.00391	4	0.700	1	1	23	4.18	6.91	27	8.23	24.4
0.00391	4	0.900	1	1	25	3.98	6.31	28	5.06	7.69
0.00195	8	0.100	1	1	20	4.09	7.84	27	6.82	22.2
0.00195	8	0.300	1	1	22	4.23	5.49	25	8.04	19.9
0.00195	8	0.500	1	1	18	3.69	4.19	26	9.93	13.2
0.00195	8	0.700	1	1	25	4.25	5.41	29	8.01	18.6
0.00195	8	0.900	1	1	20	4.06	7.8	27	6.77	22.2
0.00098	16	0.100	1	1	25	4.08	5.42	24	6.63	15.2
0.00098	16	0.300	1	1	14	1.64	2.46	41	10.9	27.6
0.00098	16	0.500	1	1	15	1.43	2.35	24	11.7	36.8
0.00098	16	0.700	1	1	14	1.64	2.46	37	10.8	29.6
0.00098	16	0.900	1	1	25	4.09	5.41	26	6.66	14.6

Table 3. Multigrid 1FMG-solver of the eqn. $\frac{\partial^2 U}{\partial \xi^2} = f$ in 2D.

$$\xi = \frac{x+ty}{\sqrt{1+t^2}}; \quad \eta = \frac{-tx+y}{\sqrt{1+t^2}}$$

6 levels; basic cycle: conditional coarsening V(1,1);

Exact continuous solution: $U = \sin(\beta_\xi \xi + \beta_\eta \eta) = \sin(\theta x + \omega y)$

t	$\beta_\xi \sqrt{1+t^2} h_y$	ωh_y	θh_x	Discretization error	Algebraic error Before cycles	After 1 cycle	After 2 cycle
0.100	0.05131	0.09817	0.04149	0.00056	0.00019	$1.92 \cdot 10^{-05}$	$9.05 \cdot 10^{-06}$
0.300	0.05330	0.09817	0.02385	0.000796	0.001	$5.14 \cdot 10^{-05}$	$1.18 \cdot 10^{-05}$
0.500	0.05708	0.09817	0.00799	0.00065	0.00156	0.000101	$9.23 \cdot 10^{-06}$
0.800	0.06538	0.09817	−0.01316	0.000395	0.000298	$4.19 \cdot 10^{-05}$	$9.71 \cdot 10^{-06}$
0.100	0.05131	0.98175	−0.04687	0.0789	0.0693	0.0179	0.00554
0.300	0.05330	0.98175	−0.24123	0.409	0.0542	0.00939	0.00225
0.500	0.05708	0.98175	−0.43380	0.477	0.022	0.00397	0.000819
0.800	0.06538	0.98175	−0.72002	0.402	0.0527	0.0109	0.00242
0.100	0.05131	2.45437	−0.19413	0.604	0.0424	0.0111	0.00429
0.300	0.05330	2.45437	−0.68301	0.676	0.0523	0.00697	0.00136
0.500	0.05708	2.45437	−1.17011	0.681	0.0478	0.00473	0.000373
0.800	0.06538	2.45437	−1.89812	0.67	0.0507	0.00926	0.00266
0.100	0.72025	0.09817	0.71043	0.0384	0.0462	0.00206	0.000944
0.300	0.74823	0.09817	0.71878	0.041	0.0539	0.00203	0.000205
0.500	0.80127	0.09817	0.75218	0.0539	0.0699	0.00316	0.000129
0.800	0.91779	0.09817	0.83925	0.063	0.112	0.00188	0.000332
0.100	0.72025	1.07992	0.61226	0.111	0.0817	0.014	0.00438
0.300	0.74823	1.07992	0.42425	0.156	0.159	0.00758	0.00185
0.500	0.80127	1.07992	0.26131	0.135	0.257	0.0145	0.0022
0.800	0.91779	1.07992	0.05385	0.0881	0.152	0.00232	0.000534
0.100	0.72025	2.06167	0.51408	0.394	0.125	0.0038	0.00142
0.300	0.74823	2.06167	0.12973	0.44	0.533	0.0896	0.0179
0.500	0.80127	2.06167	−0.22957	0.118	0.494	0.0535	0.00584
0.800	0.91779	2.06167	−0.73154	0.00653	0.258	0.00665	0.00146
0.100	2.20022	0.09817	2.19040	0.433	1.36	0.00272	0.000964
0.300	2.28569	0.09817	2.25624	0.513	1.46	0.041	0.00159
0.500	2.44771	0.09817	2.39862	0.685	1.63	0.0792	0.00067
0.800	2.80367	0.09817	2.72513	0.836	2.02	0.134	0.0337
0.100	2.20022	0.88357	2.11186	0.476	1.43	0.0108	0.00308
0.300	2.28569	0.88357	2.02062	0.551	1.65	0.0255	0.00419
0.500	2.44771	0.88357	2.00592	0.655	1.85	0.0622	0.00596
0.800	2.80367	0.88357	2.09681	0.894	3.26	0.242	0.0722
0.100	2.20022	2.25802	1.97441	0.682	1.7	0.0267	0.00768
0.300	2.28569	2.25802	1.60829	1.33	2.63	0.0459	0.00377
0.500	2.44771	2.25802	1.31870	1.59	2.33	0.104	0.0106
0.800	2.80367	2.25802	0.99725	1.26	5.46	1.72	0.668