

COARSENING STRATEGIES FOR UNSTRUCTURED MULTIGRID TECHNIQUES WITH APPLICATION TO ANISOTROPIC PROBLEMS*

E. MORANO[†], D. J. MAVRIPLIS[‡], AND V. VENKATAKRISHNAN[‡]

Abstract. Over the years, multigrid has been demonstrated as an efficient technique for solving inviscid flow problems. However, for viscous flows, convergence rates often degrade. This is generally due to the required use of stretched meshes (i.e., the aspect ratio $AR = \Delta y/\Delta x \ll 1$) in order to capture the boundary layer near the body. Usual techniques for generating a sequence of grids that produce proper convergence rates on isotropic meshes are not adequate for stretched meshes. This work focuses on the solution of Laplace's equation, discretized through a Galerkin finite-element formulation on unstructured stretched triangular meshes. A coarsening strategy is proposed and results are discussed.

Key words. multigrid method, unstructured meshes, semicoarsening, anisotropic problems

AMS subject classifications. 65M55, 65M60, 76M10

PII. S1064827595287638

Introduction. Multigrid methods are well known to be successful for solving elliptic problems. This is mainly due to their good damping properties which result from two very simple principles. A usual Fourier analysis demonstrates that most of the commonly used solvers effectively damp the high frequencies of a signal. A low frequency component of a given signal on a fine mesh becomes a high frequency on a coarser one, hence the idea of solving the same problem on a sequence of meshes where all frequencies can be damped equally and, if enough grids are available, only a few iterations will be required to produce a converged solution (for more details see [4]). Despite these rather simple considerations, the multigrid algorithm is complex and difficult to implement. One of the difficulties resides in the generation of the sequence of grids for unstructured meshes. The convergence properties of the multigrid method depend upon the "quality" of these grids.

A sequence of meshes may be produced through two different methods. First, starting from a mesh that is not too fine but correctly represents the problem, finer meshes may be generated through refinement. A global refinement, performed through local subdivision of the triangles of the discretization, tends to preserve the geometrical features required to obtain an efficient multigrid method. However, this will clearly not be efficient in terms of computational cost, hence the local refinement technique where specific regions of the mesh are refined and then possibly adapted [20]. Although this method seems more reasonable, it increases the computational time and the complexity of the multigrid algorithm.

Another method consists in coarsening an existing fine mesh, which has been created to represent accurately the different phenomena to be observed. One of the techniques available consists in removing, through a coarsening criterion, a certain

*Received by the editors June 12, 1995; accepted for publication (in revised form) January 12, 1997; published electronically September 10, 1998. This research was supported under NASA contract NAS1-19480 while the authors were in residence at ICASE.

<http://www.siam.org/journals/sisc/20-2/28763.html>

[†]Graduate Aeronautical Laboratories, MS 205-45, California Institute of Technology, Pasadena, CA 91125 (eric@galcat.caltech.edu).

[‡]Institute for Computer Applications in Science and Engineering, MS 132C, NASA Langley Research Center, Hampton, VA 23681-0001 (dimitri@icase.edu, venkat@icase.edu).

number of nodes from the initial mesh and reconnecting (retriangulating) the remaining set of nodes. Guillard in [8] developed a coarsening algorithm that only relies on a node proximity criterion coupled with a Delaunay reconnection technique which was successfully applied in [16] for nonstretched mesh computations. However, the Delaunay technique tends to produce the “most equilateral” triangulation for the given point distribution and therefore is not easily applicable to stretched meshes. Similar works were done by Bank and Xu in [1, 2] and Chan and Smith in [5]. The coarsening algorithms developed by Bank or Chan also rely on node proximity. Bank assumes that the fine grid is obtained from global refinement which provides a set of triangle subdivision rules. By reversing the process he builds a sequence of coarse grids. Chan’s initial algorithm is very similar to Guillard’s but he proposes a “dual” algorithm where the nodes of the coarse grid lie at the center of a fine grid element. The coarse grid is then generated by removing the neighboring elements and reconnecting the centers of the remaining elements. All these techniques provide nested, structured-like multigrid convergence rates. However, in the case of Chan’s dual algorithm not only are the triangles not nested but the vertices as well. Practical applications for solving the Euler equations [12] with nonnested meshes do not produce in general as good convergence rates as those obtained when the grids are at least node nested [16]. However and as mentioned previously, those techniques cannot be directly applied to highly stretched meshes or to anisotropic problems.

In order to avoid retriangulation, the so-called agglomeration technique (see Lallemand, Stève, and Dervieux [11]) is of interest. The generation of coarser meshes consists in the agglomeration, or fusion, of the control volumes of the discretization. However, for consistency considerations, when it comes to viscous flows, more accurate intergrid transfer operators are required [10, 15].

The following study focuses on the 2D Laplace equation $\Delta u(x, y) = 0$, since the poor convergence properties of the multigrid technique, observed when solving the Navier–Stokes equations on stretched meshes, also appear for the solution of this simpler equation. The purpose of this work is to propose new coarsening strategies that will preserve the convergence rate of the usual isotropic multigrid technique. The semicoarsening method introduced in this work relies on the discretization of the equation and is related to algebraic multigrid (see [21]). This study will show how this process may be extended from the case of regular structured grids to totally unstructured meshes. This work is restricted to solely Dirichlet boundary conditions, since the coarsening methodology shall not change the well posedness of the fine grid problem. In a recent work [7], Francescatto successfully applied an algebraic-like semicoarsening method to the multigrid agglomeration technique.

The organization of the paper is as follows: the discretization of the 2D Laplace equation is introduced in section 1 along with an edge-based data structure. Section 2 recalls the essential multigrid convergence properties. The generation of stretched grids is addressed in section 3. A semicoarsening algorithm, extended to unstructured meshes, is presented in section 4. Finally, numerous experiments are discussed in section 5.

1. The Laplace equation. The problem consists in solving Laplace’s equation:

$$(1) \quad \begin{cases} \Delta u(x, y) = u_{xx} + u_{yy} = 0 & \text{on } \Omega \text{ convex polygonal domain,} \\ u = u_0 & \text{on } \Gamma. \end{cases}$$

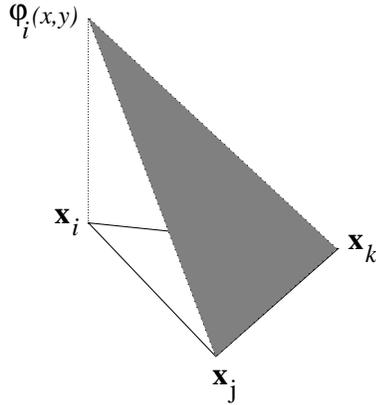


FIG. 1. Linear basis function φ_i .

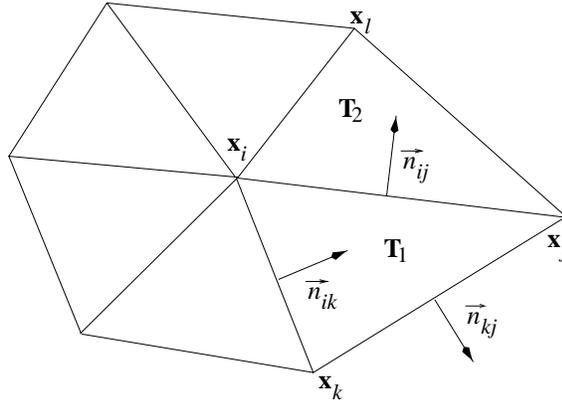


FIG. 2. Vertex i and connecting neighbors.

A Galerkin finite-element formulation is used on unstructured triangular meshes. An integration by parts results in

$$(2) \quad \int_{\Omega_i} \Delta u \varphi_i \, d\omega = - \int_{\Omega_i} \vec{\nabla} u \cdot \vec{\nabla} \varphi_i \, d\omega + \int_{\Gamma_i} \vec{\nabla} u \cdot \vec{n} \varphi_i \, d\sigma,$$

where φ_i is the linear basis function as depicted in Fig. 1. If u is piecewise linear, then the Green formula and the notation of Fig. 2 result in

$$(3) \quad \begin{cases} (\vec{\nabla} \varphi_i)_{T_1} = \frac{-1}{2A_1} \vec{n}_{kj}, \\ (\vec{\nabla} u)_{T_1} = \frac{-1}{2A_1} (u_i \vec{n}_{kj} + u_k \vec{n}_{ij} - u_j \vec{n}_{ik}), \end{cases}$$

where $u_i = u(\mathbf{x}_i) = u(x_i, y_i)$ is the value of the solution u at vertex i , A_1 is the area of triangle T_1 , and \vec{n}_{ij} is the vector normal to the edge $[i, j]$ and is of magnitude equal to the length of the edge. Equation (2) can be rewritten as

$$(4) \quad \int_{\Omega_i} \Delta u \varphi_i \, d\omega = \sum_i \int_{T_i} (\vec{\nabla} \varphi_i)_{T_i} \cdot (\vec{\nabla} u)_{T_i} \, d\omega.$$

Moreover, for the considered triangle T_1 , (3) can be rewritten as

$$(5) \quad \begin{cases} (u_x)_{T_1} = \frac{1}{2A_1} (\delta u_{ij} \delta y_{jk} - \delta u_{jk} \delta y_{ji}), \\ (u_y)_{T_1} = \frac{-1}{2A_1} (\delta u_{ij} \delta x_{jk} - \delta u_{jk} \delta x_{ji}), \end{cases}$$

where $\delta(\cdot)_{ij} = (\cdot)_j - (\cdot)_i$. A similar formulation can be written for triangle T_2 .

In evaluating the coefficient for the edge joining vertices i and j , only the triangles T_1 and T_2 will yield nonzero contributions. The discrete formulation of (1) is thus an edge-based formulation:

$$(6) \quad \begin{aligned} \Delta u &\sim \sum_{\{i,j\} \in \mathcal{E}} \text{coef}_{\{i,j\}} \delta u_{ij} \\ &= \sum_{\{i,j\} \in \mathcal{E}} \frac{1}{4} \left[\left(\frac{\delta y_{ik} \delta y_{jk}}{A_1} + \frac{\delta y_{il} \delta y_{jl}}{A_2} \right) + \left(\frac{\delta x_{ik} \delta x_{jk}}{A_1} + \frac{\delta x_{il} \delta x_{jl}}{A_2} \right) \right] \delta u_{ij} \\ &= 0 \end{aligned}$$

with the notation of Fig. 2 and where \mathcal{E} denotes the set of the $n_{\mathcal{E}}$ edges $\{i, j\}$ of the mesh. The geometrical anisotropy is reflected in the coefficient associated with each edge. If the length $\|\overline{\mathbf{x}_i \mathbf{x}_j}\|$ increases (the nodes k and l being fixed) then the value of the coefficient decreases. Therefore, considering the domain $\Omega_i = \bigcup_i T_i$, the maximum coefficient is associated with the smallest connecting edge and the minimum with the longest.

Remark. In the case of the anisotropic equation

$$(7) \quad \begin{cases} u_{xx} + \varepsilon u_{yy} = 0 \text{ on } \Omega, \varepsilon \in [0, 1], \\ u = u_0 \text{ on } \Gamma, \end{cases}$$

the edge coefficient would read

$$coef_{\{i,j\}} = \frac{1}{4} \left[\left(\frac{\delta y_{ik} \delta y_{jk}}{A_1} + \frac{\delta y_{il} \delta y_{jl}}{A_2} \right) + \varepsilon \left(\frac{\delta x_{ik} \delta x_{jk}}{A_1} + \frac{\delta x_{il} \delta x_{jl}}{A_2} \right) \right].$$

In all that follows only geometric anisotropy is considered, i.e., $\varepsilon = 1$.

2. Some definitions and convergence results. Multigrid theory relies on the use of a sequence of nested meshes for solving (1). These meshes represent the different spaces where the equation is discretized. In what follows, only two meshes are considered: \mathcal{H}_h and \mathcal{H}_H with $H = 2h$ and $\mathcal{H}_H \subset \mathcal{H}_h \subset H_1^0$. The discrete problem on the fine grid is written as

$$(8) \quad A_h u_h = 0.$$

A weighted Jacobi relaxation is considered as the basic iterative process or smoother:

$$(9) \quad u_h^{n+1} = S_h u_h^n = (I - \omega D_h^{-1} A_h) u_h^n, \text{ where } D_h = (A_h)_{ii}.$$

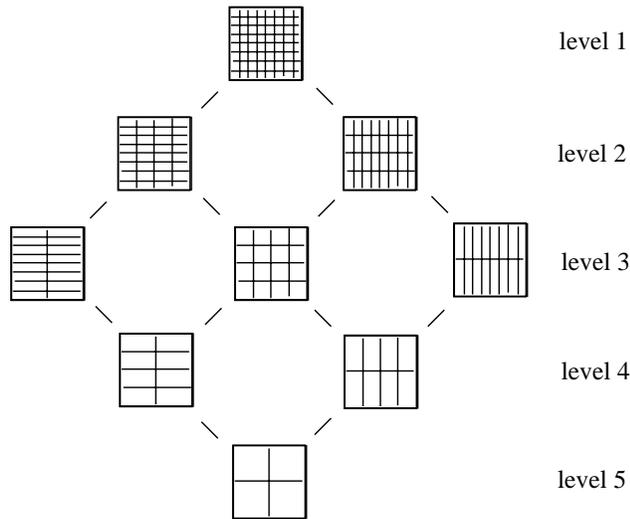
In order to use both spaces for solving (8) it is necessary to use transfer operators. A linear interpolation $P : \mathcal{H}_H \rightarrow \mathcal{H}_h$ defines the prolongation operator, and its transpose $R = P^* : \mathcal{H}_h \rightarrow \mathcal{H}_H$ defines the restriction. The two-grid iterative operator M_h is then defined by

$$(10) \quad \begin{aligned} u_h^{n+1} &= M_h u_h^n = S_h^{\nu_2} (I - P A_H^{-1} R A_h) S_h^{\nu_1} u_h^n \\ &= (A_h^{-1} - P A_H^{-1} R) (A_h S_h^{\nu_1}) u_h^n \end{aligned}$$

with $\nu_1 = \nu$ prerelaxations and $\nu_2 = 0$ postrelaxations.

One very important feature of a multigrid (MG) algorithm is its mesh-independent convergence. According to Hackbusch [9], mesh independence for elliptic operators is achieved through the smoothing property ($\|A_h S_h^{\nu}\| \leq h^{-2} \eta(\nu)$, where $\lim_{\nu \rightarrow \infty} \eta(\nu) = 0$) and the approximation property ($\|A_h^{-1} - P A_H^{-1} R\| = O(h^2)$). Because of its nature, the MG algorithm converges linearly with respect to the number of MG cycles.

Morano and Dervieux, in [16], showed that this may also be achieved for the Euler and low Reynolds number Navier–Stokes equations where the employed meshes are not stretched. However, when highly stretched elements are used (mandatory for high Reynolds number solutions, see [15] for example), this convergence greatly deteriorates with classical fully coarsened (FC) grids. It is no longer linear nor mesh independent. The deterioration in convergence is also observed when the resolution of Laplace’s equation is attempted with highly stretched elements, that is, when the mesh is anisotropic.

FIG. 3. *Sequence of grids for MSG.*

3. A sequence of grids. When very stretched elements are used, the damping properties of the smoother are negligible in the stretching direction. Thus using a full-coarsening strategy will certainly not improve the damping properties, since the stretching is fully preserved on larger elements. Moreover, the distribution of nodes in the stretching direction will correctly represent the low frequencies of the signal, whereas, in the direction normal to the stretching, it will represent the high frequencies. Because of the nature of the smoothers commonly used, the multigrid technique damps mainly the high frequencies, hence the idea of semicoarsening in the direction normal to the stretching.

The semicoarsening technique is well known and used especially in the structured mesh community. For complex geometries, however, multiple directions within the mesh require semicoarsening. A process named multiple semicoarsened grid (MSG) algorithm was introduced by Mulder [18]. This technique relies on the generation of numerous grids that are semicoarsened from the finer grid in all possible directions as depicted in Fig. 3. This ensures proper dissipation of the signal. A multigrid scheme is then implemented using all the grids, which is complex and costly, especially for 3D problems [19]. Moreover, there is no possible extension of this technique to unstructured grids.

The complexity of the usual multigrid technique also relies on the full-coarsening method. This technique consists in removing every second vertex in each direction on a regular structured mesh, which results in a number of nodes of the coarse grid decreased by a factor 4 in two dimensions or 8 in three dimensions. The V -cycle complexity of such a method tends to $4/3$ work units (WUs) in two-dimensions (a WU corresponds to the computation of one residual on the fine grid), whereas the semicoarsening technique produces coarse grids with a number of nodes decreasing by a factor 2 and the overall complexity tends to 2. Therefore, such a method will cost more per cycle. However, it will be shown that this technique allows a much better damping factor than a regular full-coarsening technique in the case of stretched meshes.

The smoothing property is valid for the weighted Jacobi relaxation scheme applied in this study. The effect of the approximation property is emphasized since it determines the mesh independence of the convergence. This property is verified when the discretized subspaces, defined by the sequence of coarser meshes, utilized within the MG algorithm are nested. In this paper, the sequence of meshes is created through a semicoarsening technique followed by a retriangulation. When this strategy is applied to unstructured meshes, the nestedness of the meshes is rather difficult to preserve. The nodes of the coarse grid form a subset of the nodes of the fine grid which produces node-nested, but not element-nested, grids. The example depicted in Fig. 4 shows how the convergence varies with respect to the nestedness of the meshes. A nonstretched 89-node Cartesian mesh defines the fine grid (Fig. 4a). The boundary conditions are those defined in section 5. Three different coarse grids are considered. Each of them is a node-nested grid and comprises 25 nodes. Figure 4b shows a usual fully nested grid. Figure 4c and 4d depict randomly coarsened grids. On the right side of the grid shown in Fig. 4c a few elements are not nested. Finally, Fig. 4d depicts a coarsened grid where the elements are anything but nested. Two-grid experiments (see section 5) are performed, and Fig. 4e depicts the respective convergence histories. The convergence rate ranges from 0.15 to 0.31 for such a simple test case. Therefore, the nestedness of the grids is of extreme importance in the quality of the MG performance. Further results may be obtained in [23].

4. Semicoarsening and unstructured meshes. In what follows a semicoarsening technique is presented that is applicable to unstructured meshes as well as to structured meshes. The technique may be seen as a variant of the algebraic multigrid (see [21]) in the sense that it necessitates a preprocessing stage that relies on the discretization of the equation for generating the coarse grids. As shown previously, the Galerkin discretization of Laplace's equation amounts to a sum over all edges. The value of the coefficient associated with each edge is determined by the geometry of the two surrounding elements (triangles). The smaller the length of the edge, the larger the value of the coefficient. The semicoarsening technique proceeds as follows: amongst the initial set of nodes forming the fine grid some are selected to form the coarse grid and others are removed. Once a node is selected to remain on the coarse grid, its neighbors must be scanned to determine which one of them has to be removed. The usual technique on structured meshes appeals to the fact that high frequencies must be damped as much as possible. The closer two nodes are, the better they will represent the high frequency components of the solution. Therefore, on such grids, it is natural to eliminate every other vertex in the direction normal to the stretching. This will obviously balance the overall dissipation of the MG scheme. In the case of unstructured meshes no such tool as the Fourier analysis are available and justification of the method through functional analysis would be a horrendous work, which is not the purpose of this paper. However, it is expected that the algorithm will be able to reproduce the usual technique when applied to regular structured-like meshes. A more thorough study of the usual semicoarsening technique shows that if anisotropy is also present in the formulation of the equation, $\varepsilon < 1$ in (7), pure geometric considerations are no longer sufficient to provide a correct coarse grid. It is now necessary to consider the discretization itself. It is well known that equation (7) is equivalent to equation (1) where the aspect ratio would be $1/\varepsilon$. Therefore, in order to generate a proper coarse grid it is necessary to coarsen in the direction normal to the "equivalent" stretching direction, which amounts to coarsening the nodes in the direction where $coef_{\{i,j\}}$ is maximum.

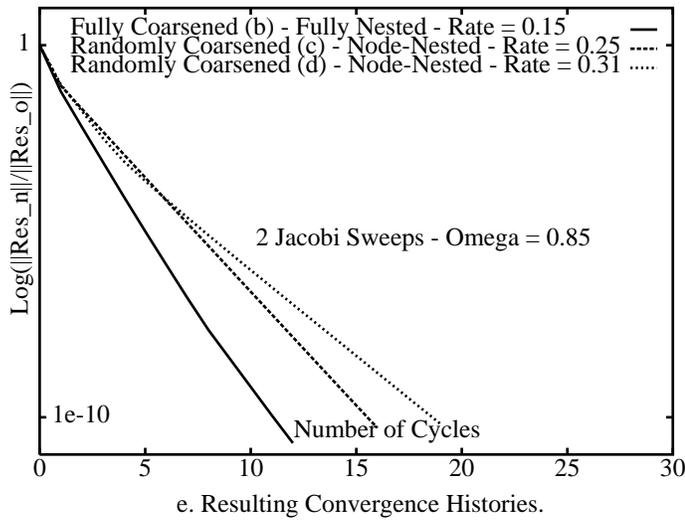
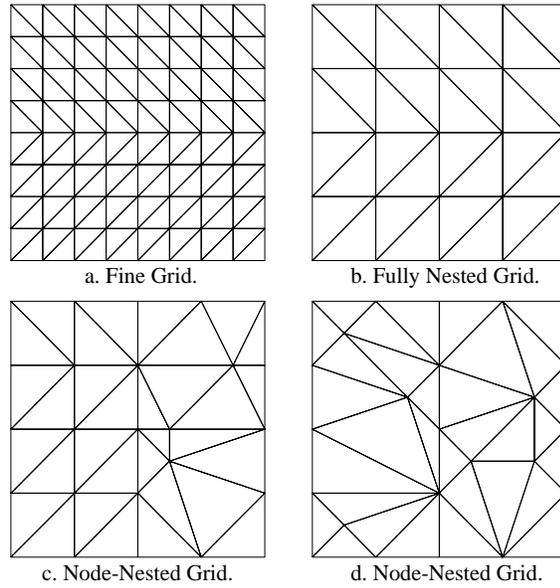


FIG. 4. Coarse grid discretizations $AR = 1$.

Unstructured meshes for high Reynolds number flow computations are essentially composed of two regions: one where the aspect ratio is (very) small, where the viscous effects are dominant, and another one, where the aspect ratio is close to 1, far from the viscous effects (the farfield for example). In order to preserve the low complexity of an MG algorithm it might be desirable to perform the semicoarsening only in the low aspect-ratio region, whereas a full coarsening may be applied elsewhere. Again, this is similar to an algebraic multigrid as described in [21]. This should provide a slightly better complexity than the one obtained through semicoarsening only.

In summary, the algorithm necessitates a preprocessing stage where the coefficients $coef_{\{i,j\}}$ are computed. Then the semicoarsening algorithm *per se* takes place. In order to account for the changing anisotropy within the mesh it is decided (in this release of the code) that the edges of the mesh be “inspected” in a somewhat orderly fashion starting from the anisotropic region ending with the more isotropic region. As previously stated only geometric anisotropy is to be considered.

4.1. The algorithms. The two essential algorithms used to create the coarse grids are presented here.

ALGORITHM 1. Preprocessing stage.

```
do i=1,N
  do j=1,K(i)
    Compute  $coef_{\{i,j\}}$ ;
  enddo
   $avg_i = \frac{1}{K(i)} \sum_{j=1}^{K(i)} coef_{\{i,j\}}$ ;
   $max_i = \max_{1 \leq j \leq K(i)} (coef_{\{i,j\}})$ ;
enddo
```

where N is the total number of nodes on the fine grid and $K(i)$ the number of neighbors of a node i .

ALGORITHM 2. Semicoarsening with heapsort.

The setup employed for coarsening is the same as that used for agglomeration in [22, 15]. It is necessary to define a distance such that the coarsening progresses from the maximum anisotropy region (usually at physical boundaries) to the minimum anisotropy region. Let $\mathcal{W} = \{j, \dots, k\}$ be the set of nodes that comprises the wall boundary (defined by the user), let i be any node in the mesh that is not in \mathcal{W} , and let

$$\mathcal{C}_{ij} = \{\text{edges that consecutively connect } i \text{ to } j, j \in \mathcal{W}, \text{ such that } |\mathcal{C}_{ij}| \text{ is minimum}\}.$$

The distance is given by

$$\text{dist}(i, \mathcal{W}) = \text{dist}_i = - \min_{j \in \mathcal{W}} (|\mathcal{C}_{ij}|).$$

In order for the algorithm to progress through the mesh, the set of edges \mathcal{E} is sorted out through a heapsort technique [6] based on the above-defined distance of the edge end points from the boundary. A dynamic heaplist is then maintained for determining the order in which the mesh vertices are considered for inclusion in the coarse grid. At any time the heaplist contains all edges which delimit the boundaries between processed and unprocessed mesh vertices. (A processed mesh vertex is one which has already been included or deleted from the coarse grid subset.) As new vertices are processed, edges are continuously added to and deleted from the heap. By basing the heaplist priority on the distance from the boundary, the algorithm advances through the grid from boundaries toward regions of isotropic mesh triangles. For each edge chosen from the heaplist, both vertices are examined.

The second part of the algorithm consists of determining the appropriate number of neighboring vertices to be deleted for the chosen vertex. This is based on the relative values of avg_i and max_i , as can be seen from the pseudo-code description of the algorithm below:

```

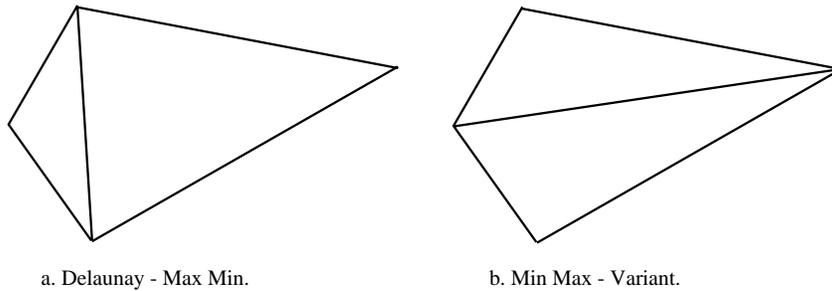
     $\mathcal{L} = \mathcal{E}$ ;
(1)  $\mathcal{L} \leftarrow \text{Heapsort}(\mathcal{L})$ ;
     $e_1 = \mathcal{L}(1)$ ;
%% Pick coarse grid vertex (seed point) from one end
%% of current front edge.
    do j=1,2
        if (.not.Processed( $e_1(j)$ )) then
             $jpick = e_1(j)$ ;
            Processed( $jpick$ ) = .true.;
             $\mathcal{C} \leftarrow \mathcal{C} \cup \{jpick\}$ ;
%%  $\beta > 0$  is a user defined parameter.
            if ( $max_{jpick} \geq \beta avg_{jpick}$ ) then
                 $n_{coarse} = 1$ ;
            else
                 $n_{coarse} = 3$ ;
            endif
(2)  $loc_{max} = \max_{i \in \mathcal{N}_{jpick}} (coef_{\{jpick,i\}})$ ;
%% Delete appropriate number of neighboring vertices.
             $n_{delete} = 0$ ;
            do k=1,  $|\mathcal{N}_{jpick}|$ 
                 $ngbh = \mathcal{N}_{jpick}(k)$ ;
                if ( ( $coef_{\{jpick,ngbh\}} = loc_{max}$  ) .and.
                    ( $loc_{max} \geq avg_{jpick}$  ) ) then
                    Processed( $ngbh$ ) = .true.;
                     $\mathcal{N}_{jpick} \leftarrow \mathcal{N}_{jpick} \setminus \{ngbh\}$ ;
                     $n_{delete} \leftarrow n_{delete} + 1$ ;
                endif
            enddo
            if ( $n_{delete} < n_{coarse}$ ) Goto (2);
        endif
    enddo
%%  $e_1$  is removed from the list.
     $\mathcal{L} \leftarrow \mathcal{L} \setminus \{e_1\}$ ;
    Goto (1);
enddo

```

where \mathcal{N}_i is the list of the unprocessed neighbors of i , $n_{coarse} = 1$ provides semi-coarsening, and $n_{coarse} = 3$ full coarsening. A neighbor of the seed point $jpick$ is “removed” if the value of their connecting edge coefficient is equal to the local maximum coefficient and if this local maximum is greater than the average value of all the surrounding edge coefficients.

It is clear that the starting point of the front will determine the quality of the subset of nodes which constitute the coarse grid. In the present release of the code, technical considerations make the algorithm start with both boundaries. The body and farfield extrema are retained on the coarse grid in order to preserve the general geometry of the discretized domain.

This algorithm clearly provides a semi/full-coarsening (S/FC) technique. Yet, if appropriate, the algorithm only performs semicoarsening ($n_{coarse} = 1$ always) or full coarsening ($n_{coarse} = 3$ always). Such an algorithm may be applied to unstructured meshes as well as to structured meshes provided the considered discretization relies

FIG. 5. *Retriangulation techniques.*

on an edge-based data structure. This algorithm relies on the discretization of the equation to be solved rather than on simple geometrical considerations.

4.2. Reconnecting the coarse grid nodes. Once the subset of nodes of the fine grid is obtained after coarsening, a retriangulation of these vertices is required. The reconnection relies here on a Delaunay method. This method has proved useful and efficient when used in conjunction with equilateral triangle types of meshes. The coarsening technique utilizing such an algorithm was introduced in [17]. Unfortunately, this method does not apply to highly stretched meshes. It usually results in a poor reconnection in the region where the nodes of the mesh are not regularly distributed. In order to overcome this difficulty, an edge-swapping technique may be employed [3, 13]. The Delaunay reconnection of a set of four nodes results in two triangles where the minimum angle is maximized (Fig. 5a). In lieu of preserving this connectivity it is possible to swap the edges by minimizing the maximum angle of the two triangles (Fig. 5b). This technique has proved very efficient when used with an advancing front technique for generating meshes and is thus employed for the unstructured test case in this paper. The reconnection of the structured coarse grids is performed through the usual Delaunay method.

5. Results and comments. In order to validate the previous concept, various test cases are performed for solving the Laplace equation. Results are presented on structured and unstructured meshes. The discretization domain for the structured cases is defined by a square of surface 1, while the unstructured case is defined by a pentagon enclosed in a semicircular domain. A nonstretched structured test case serves as the standard test case since it provides the best MG convergence. The relaxation parameter ω is equal to 0.85 and no optimization is performed here. Two sweeps are performed on the fine grid. The transfer operators are linear and were introduced in [14]. All cases are performed with Dirichlet boundary conditions. For the structured test cases they are defined by $u(0, y) = 1$, $u(x, 1) = 2$, $u(1, y) = 3$, and $u(x, 0) = 4$, and for the unstructured case they are equal to -1 on the body and to 1 on the farfield. For all test cases, the different grids used are presented along with the convergence histories of the various schemes. The convergence histories depict the logarithm of the norm of the normalized residual versus the number of cycles. This convergence is carried over until a residual decrease on the fine grid equal to 10^{-10} .

5.1. Two-grid experiments. These experiments require a residual decrease on the coarse grid equal to 10^{-10} . The semicoarsening-only ($nb_{max} = 1$) option of the algorithm is used for the generation of the coarse grids.

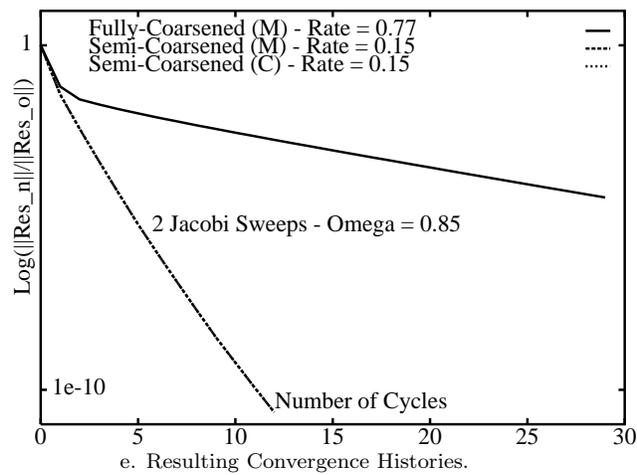
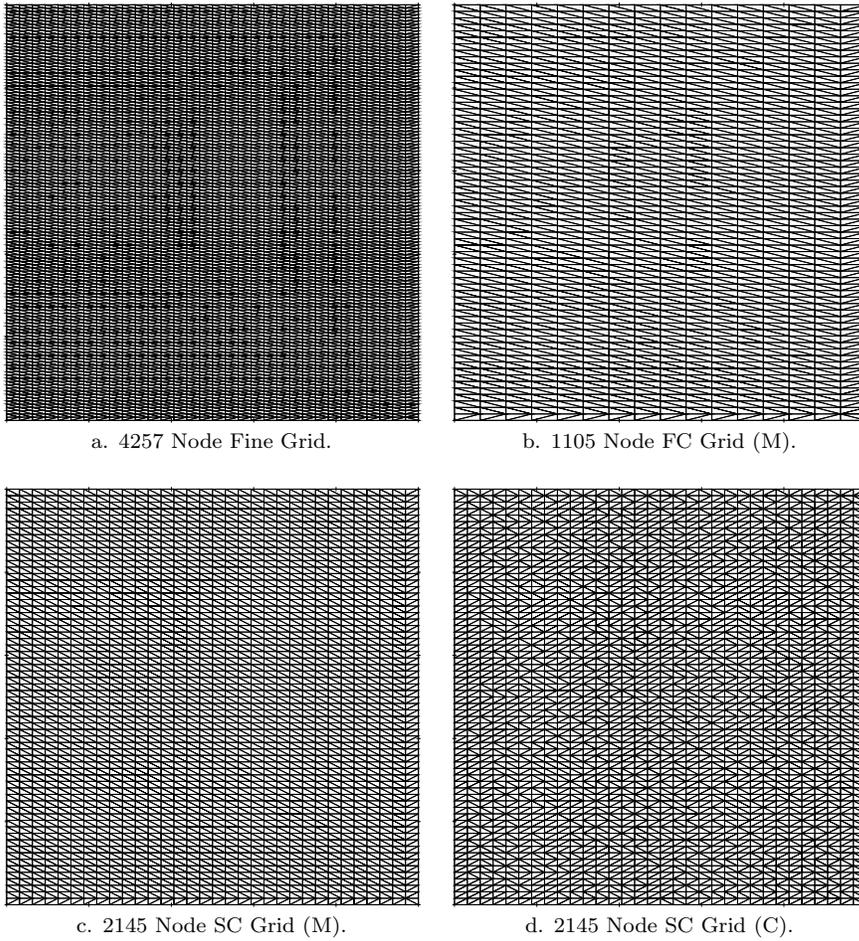


FIG. 6. Linear meshes— $AR = 1/4$.

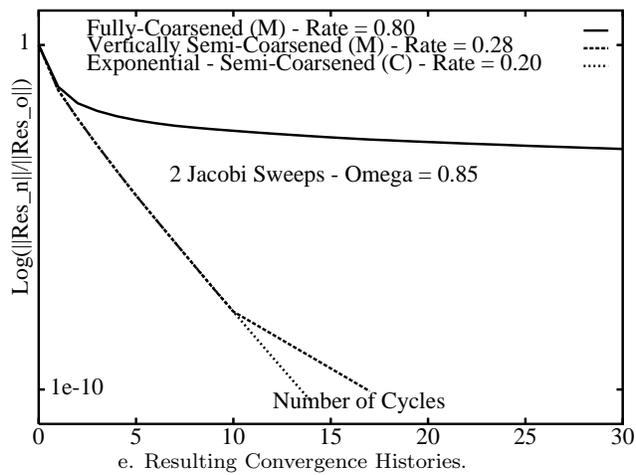
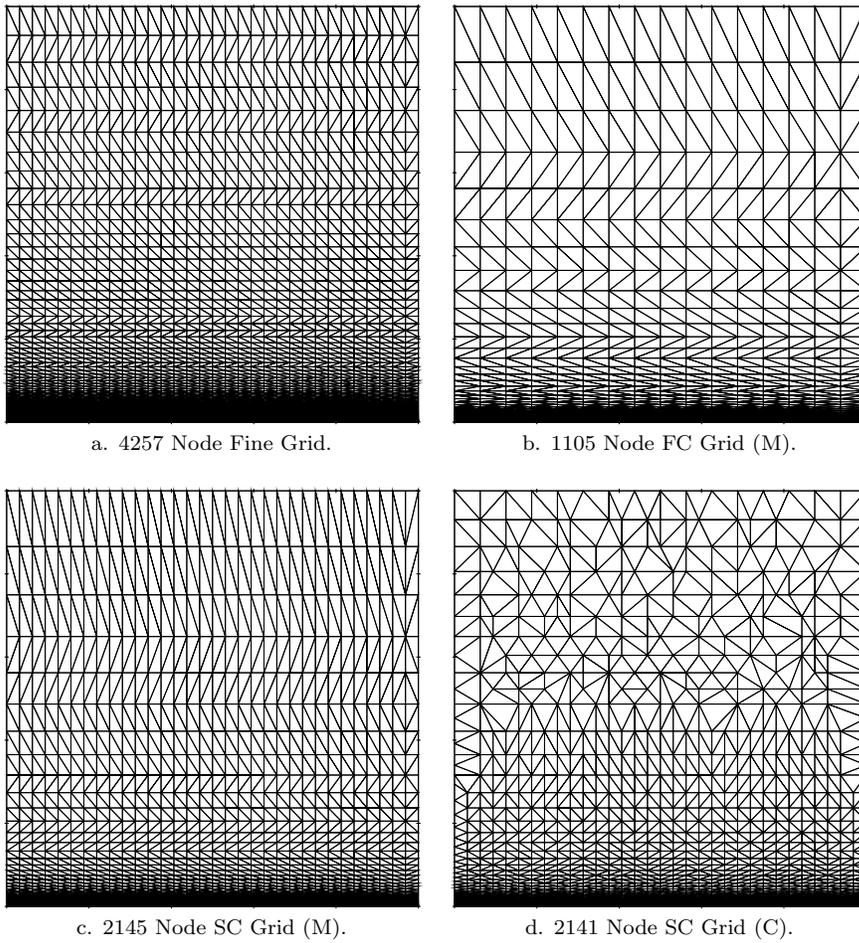
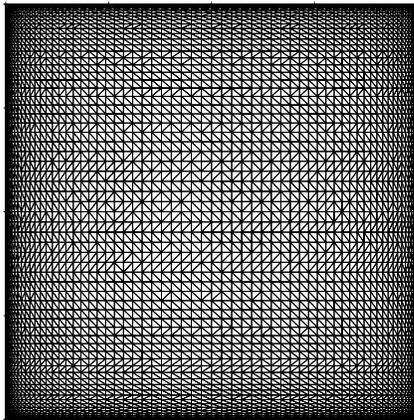
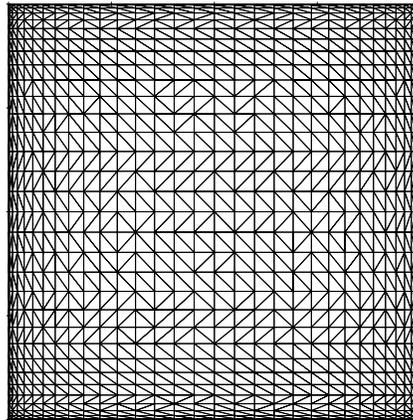


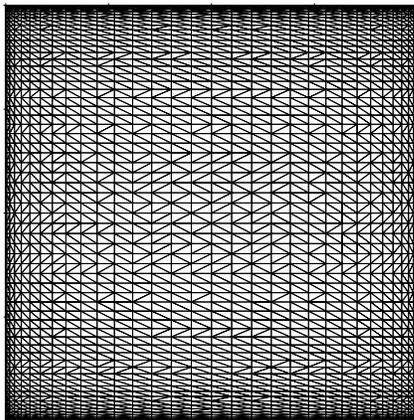
FIG. 7. Exponential meshes— $AR = 2.4 \times 10^{-4}$.



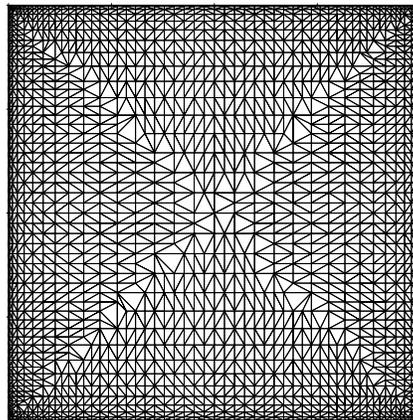
a. 4225 Node Fine Grid.



b. 1089 Node FC Grid (M).



c. 2145 Node SC Grid (M).



d. 2115 Node SC Grid (C).

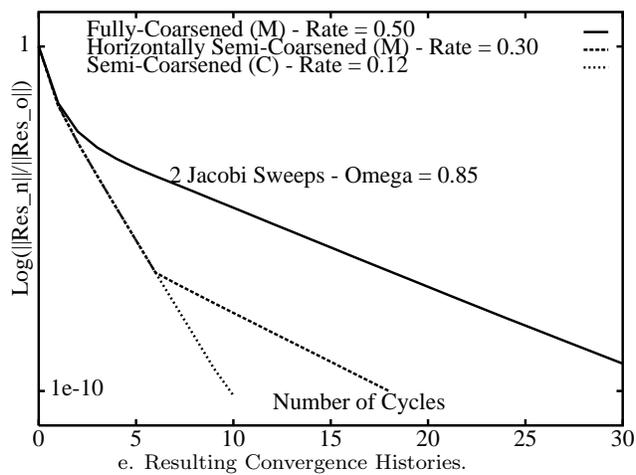
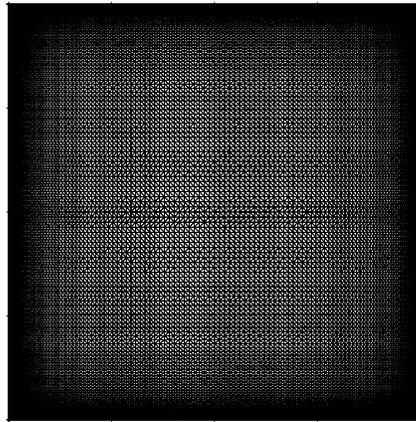
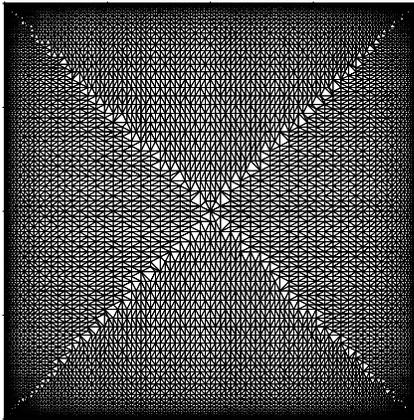


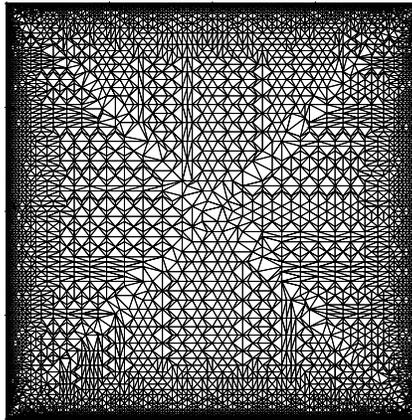
FIG. 8. Chebyshev meshes—AR = 0.024.



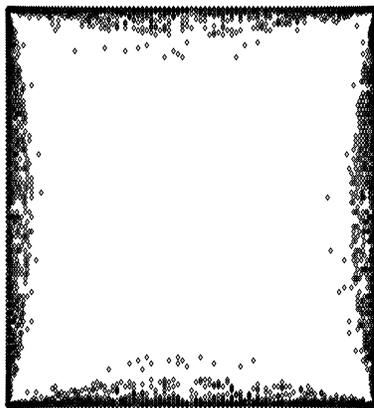
a. 16641 Node Fine Grid.



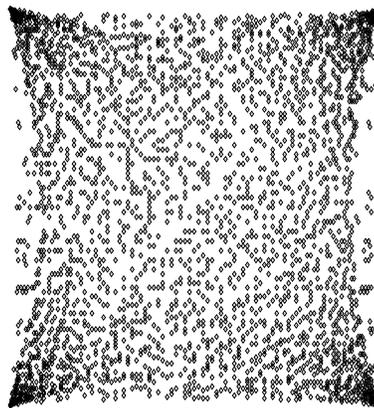
b. 8324 Node SC Grid.



c. 5968 Node S/FC Grid.



d. Semicoarsened Region.



e. Fully coarsened Region.

FIG. 9. *Multigrid Chebyshev meshes—AR = 0.012.*

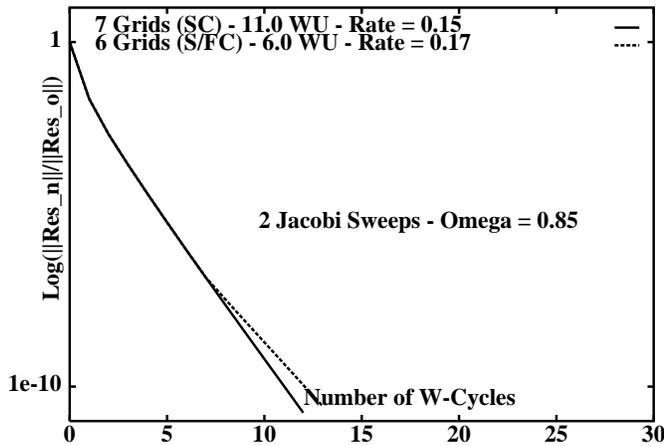


FIG. 10. Multigrid Chebyshev meshes—resulting convergence histories.

Nonstretched meshes. The aspect ratio is equal to one and the grids are fully nested. The fine and coarse grid, respectively, are similar to those depicted in Fig. 4a and 4b with 4225 (65×65) and 1089 (33×33) nodes, respectively. The coarse grid is a manually (M) fully coarsened grid (i.e., the coarsening algorithm is not involved). No anisotropy is encountered here, and a solution is obtained after 12 cycles, which corresponds to a convergence rate of 0.15.

Linear meshes. A 4257 (33×129) node fine grid is built (Fig. 6a) where the distribution of nodes is linear in the vertical (normal to the stretching) direction and the aspect ratio is equal to $1/4$. Three types of coarser meshes are presented. In Fig. 6b is depicted a manually fully coarsened 1105 (17×65) node coarse grid that represents the classical coarsening technique. In Fig. 6c and 6d are depicted two semicoarsened grids. The first grid is obtained manually through a vertical semicoarsening in a 2145 (33×65) node coarse grid. The second grid is the result of the coarsening algorithm (C) applied to the fine grid. It is a 2145 node coarse grid. The triangulations of the two semicoarsened grids appear to be different while the subset of nodes are the same. Yet similar convergences are expected. In Fig. 6e the various convergence histories are depicted. The full-coarsening technique results in a convergence rate of 0.77, while the semicoarsening techniques provide both a convergence rate equal to 0.15, which is identical to the convergence rate of the nonstretched test case.

Exponential meshes. A 4257 (33×129) node fine grid is depicted in Fig. 7a. The distribution of nodes is exponential in the vertical direction. The minimum aspect ratio is equal to 2.4×10^{-4} and the maximum to 2.2. This grid is manually fully coarsened which produces a 1105 (17×65) node coarse grid (Fig. 7b). A manually vertically semicoarsened 2145 (33×65) node coarse grid is depicted in Fig. 7c. Where the stretching follows the horizontal direction (where the distribution of nodes is more dense) this technique will provide the expected result, while the stretching deteriorates in the vertical direction (where the distribution of nodes is less dense). A 2141 node coarse grid obtained with the coarsening algorithm is depicted in Fig. 7d. In this case the coarsening follows the direction normal to the stretching everywhere in the mesh, as can be seen in the less dense region. The full-coarsening technique

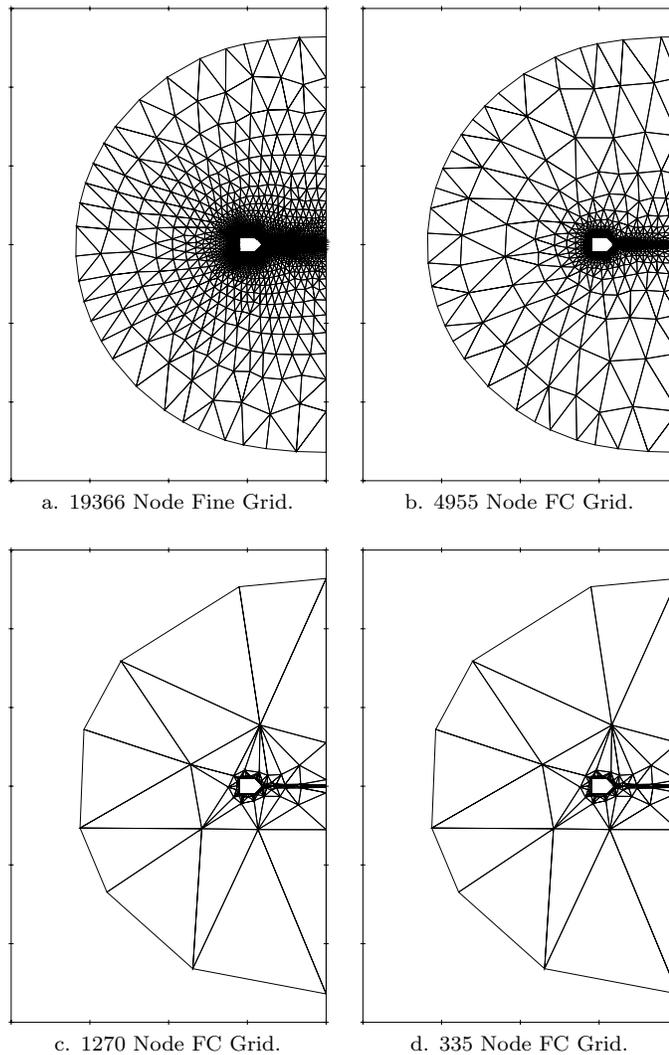


FIG. 11. *Multigrid unstructured—full coarsening— $AR = 3.7 \times 10^{-5}$.*

results in a 0.80 convergence rate (Fig. 7e). The manually semicoarsened grid proves to have a much better convergence rate of 0.28, but the best convergence rate of 0.20 corresponds to the automatically semicoarsened grid. Moreover, the vertically semicoarsened grid shows a change of slope at the end of the convergence. This means that the MG algorithm does not perform optimally and does not damp low frequencies correctly, whereas the code semicoarsened grid provides a linear type of convergence rate. Therefore, and although both semicoarsened grids have similar numbers of nodes, the coarse grid obtained through the automated coarsening algorithm results in more optimal convergence.

Chebyshev meshes. A 4225 (65×65) node fine grid is built where the distribution of nodes is a cosine function in both directions. The minimum aspect ratio is equal to 0.024 and the maximum to 40.73 (Fig. 8a). This grid comprises stretched and nonstretched elements. The minimum and maximum aspect-ratio cells are essentially

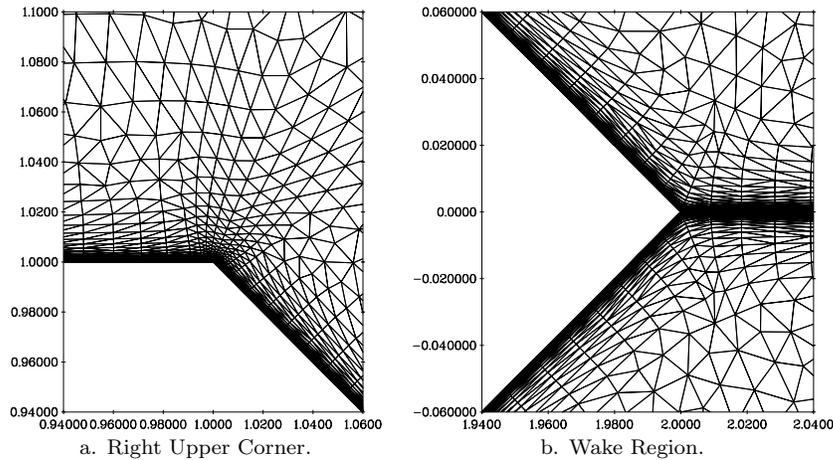


FIG. 12. *Multigrid unstructured—full coarsening— $AR = 3.7 \times 10^{-5}$.*

located at the boundary of the domain, while the isotropic cells are located in the bisectors and in the middle of the domain. A manually fully coarsened 1089 (33×33) node grid is depicted in Fig. 8b. Although no natural manual semicoarsening technique applies here, a horizontally semicoarsened 2145 node (33×65) coarse grid is built for comparison purposes (Fig. 8c). The coarsening algorithm resulted in a 2115 node coarse grid (Fig. 8d). It is again obvious that the semicoarsening follows the direction normal to the stretching, each region being clearly separated by the bisectors. The fully coarsened grid provided a convergence rate of 0.50, and 0.30 was achieved with the manually horizontally semicoarsened grid (Fig. 8e). A linear type of convergence resulting in a convergence rate of 0.12 was achieved with the code semicoarsened grid. It is interesting to note that, despite the similar number of nodes shared by the manually horizontally semicoarsened grid and the code semicoarsened grid, they provided different results, and therefore the good convergence rate of the code semicoarsening technique cannot be attributed solely to the number of nodes on the coarse grid.

5.2. Multigrid experiments. In this section, multigrid experiments are explored in order to demonstrate the robustness of the algorithm in producing a sequence of grids that permit efficient MG convergence. The number of grids will vary according to the test case. Two sweeps of the Jacobi relaxation are performed on each level and W-cycles are employed since they provide a better resolution of the coarse grid, resulting in better convergence rates. A structured Chebyshev and an unstructured test case are performed with both semicoarsening and S/FC techniques.

The Chebyshev test case. A 16641 (129×129) node fine grid is constructed with a minimum aspect ratio value of 0.012 and a maximum value of 81.50 (Fig. 9a). The semicoarsening option provides a sequence of seven grids comprising 16641, 8324 (shown Fig. 9b), 4329, 2289, 1211, 652, and 352 nodes, and the S/FC technique a sequence of six grids comprising 16641, 5968 (shown Fig. 9c), 2976, 1077, 559, and 286 nodes. The respective W-cycle complexities are equal to 11 and 6 WUs. The region where the algorithm performs the semicoarsening is depicted nodewise in Fig. 9d, while Fig. 9e shows where the full coarsening is applied. It is clear that the semicoarsening is applied to the highly stretched element region as expected. The semicoarsening

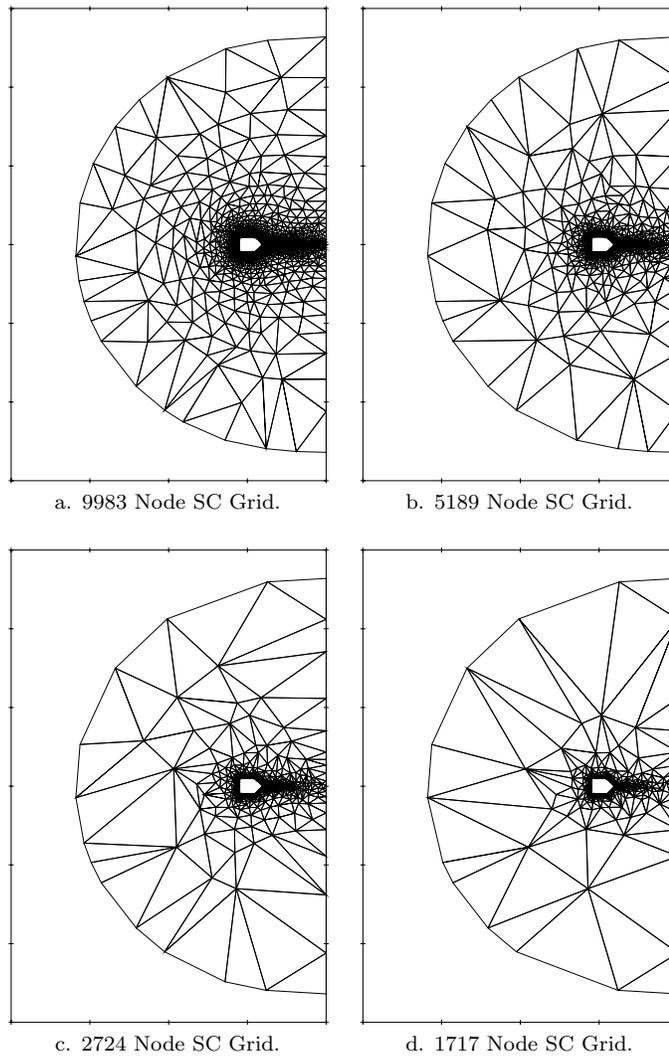


FIG. 13. *Multigrid unstructured—semicoarsening— $AR = 3.7 \times 10^{-5}$.*

technique results in a standard-like convergence rate of 0.15 (Fig. 10). When used only with six grids, this technique requires the coarsest grid to be converged completely, otherwise the process abruptly stalls at some low residual value. A convergence rate of 0.17 and a low complexity favor the S/FC technique. Yet the convergence history displays a (slight) change of slope. This indicates that the method is sensitive to the quality of the triangulation of the coarse grids. Mesh-independent convergence is the purpose of this study and is only truly achieved with the semicoarsening technique. The slightly poorer type of convergence associated with the S/FC technique may be explained by the quality of the triangulation of the coarse grid. Full coarsening in nonstretched regions tends to deteriorate the relative difference of the aspect ratio between the highly and nonstretched regions. Moreover, the addition of a seventh grid, or even converging the coarsest level, does not change the convergence.

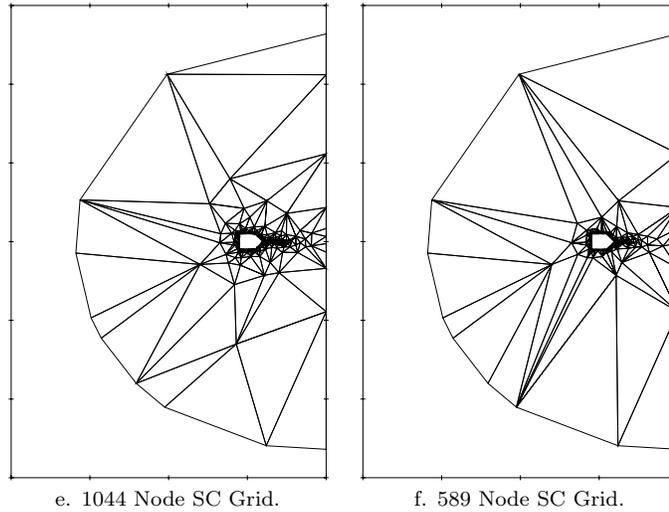


FIG. 13. (continued).

The unstructured test case. In this case (Fig. 11a), a grid-spacing $\Delta y = 10^{-6}$ on the body results in an average minimum aspect ratio of 3.7×10^{-5} . In Fig. 12a and 12b are depicted the zoom of the right upper corner and of the wake region, respectively, in order to show the different types of stretched and nonstretched elements that appear in these meshes. A first sequence of four fully coarsened meshes is manually constructed. The number of nodes for each level are 19366, 4955, 1270, and 335. These meshes are depicted in Fig. 11a to Fig. 11d. The complexity of a W-cycle is equal to 3.2 WUs.

The second sequence is obtained with the semicoarsening technique only. There are seven meshes that have 19366, 9983, 5189, 2724, 1717, 1044, and 589 nodes (Fig. 13a to Fig. 13f). The W-cycle complexity is equal to 12.5 WUs. The last sequence of meshes results from the S/FC technique and provides seven meshes (Fig. 14a to Fig. 14f): they comprise 19366, 9594, 4708, 2325, 1391, 794, and 424 nodes, resulting in an 11 WU W-cycle complexity. Semicoarsening and S/FC methods required all coarse point sets to be retriangulated using the min-max Delaunay variant. In order to maintain favorable convergence rates, it was found that the fine grid needed to be retriangulated according to the same technique. This can partially be explained by the quality of the nestedness of all the grids as seen in section 4. The fine grid is not depicted here for these last two sequences because it would appear similar to the original (Fig. 11a). However, the difference between the original and retriangulated fine grids, mostly confined to wake regions, is illustrated in Fig. 15a and 15b.

Converging the coarsest grid of the sequence of the fully coarsened grids does not change the convergence rates equal to 0.80 (Fig. 16). This indicates that the use of an additional coarser grid would not change the convergence. Besides, the retriangulation of the entire sequence of the fully coarsened grids does not change the convergence rate of the MG algorithm, whether or not the coarsest grid is converged. The S/FC and semicoarsened grids provide a clear improvement with respect to the usual fully coarsened grids with convergence rates equal to 0.23. The S/FC grids demonstrate a better behavior than in the Chebyshev case because they are very similar to the semicoarsened grids. Indeed, since most of the nodes are concentrated in the highly stretched regions, the algorithm performs essentially as a semicoarsening technique.

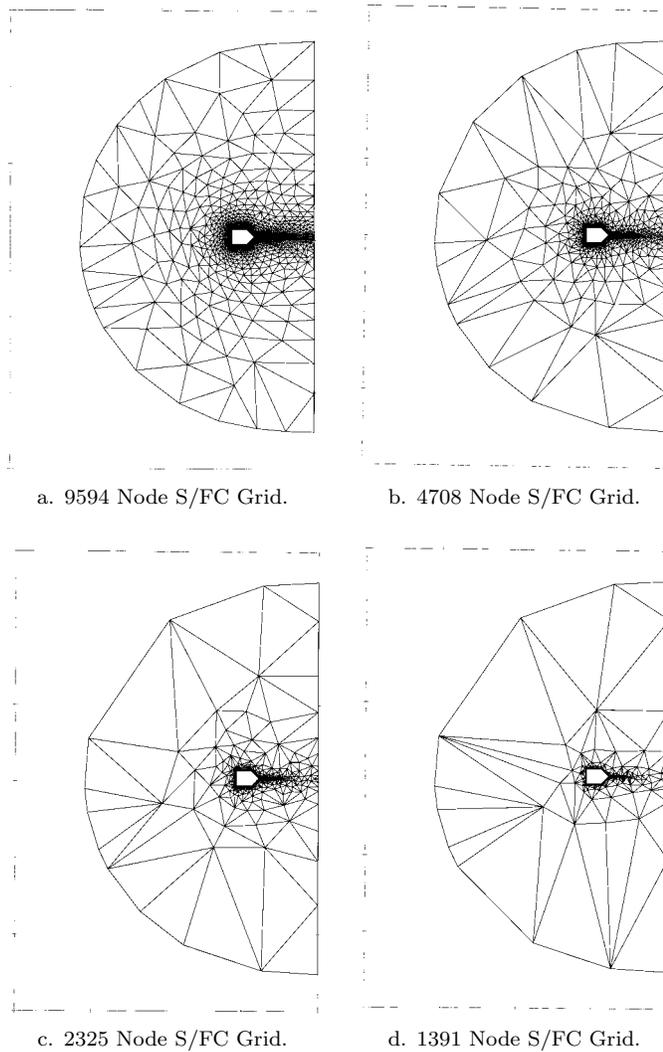


FIG. 14. *Multigrid unstructured—S/FC—AR = 3.7×10^{-5} .*

This type of mesh is more similar to exponential-type meshes rather than Chebyshev meshes.

6. Concluding remarks. In Fig. 17 are gathered the most significant results. They are separated in two different subsets. Curves 1 and 2 represent the spectrum of convergences within which the other convergence histories must fit. Indeed, curve 1 shows the best convergence and curve 2 shows what is expected when the discretization subspaces are only node nested. All other curves depict the convergence histories of the various test cases that employ the semicoarsening algorithm. The problem to be solved is the same for all test cases; only the geometries of the discretized spaces differ. The results are straight lines with similar slopes that fall within the predicted range. The difference of slopes may be explained by two essential reasons. First, the domain boundary of the structured and unstructured test cases differ. It is not

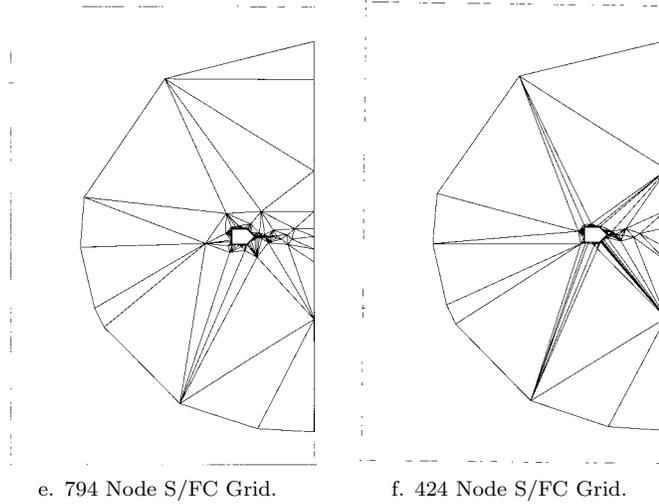


FIG. 14. (continued).

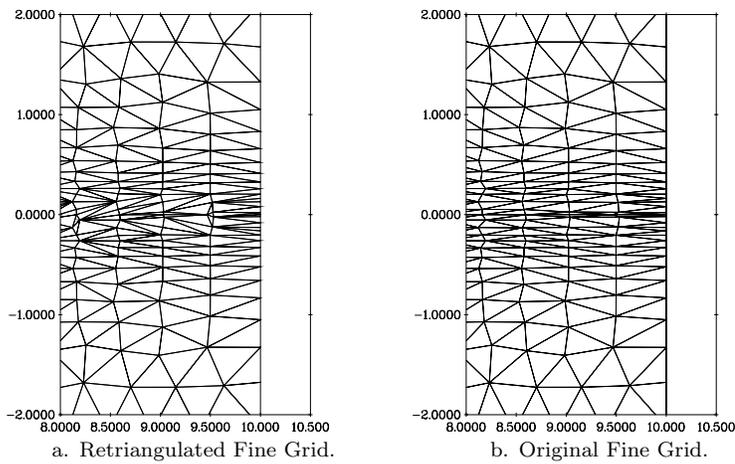


FIG. 15. Wake regions of unstructured grids.

possible, due to the geometry, to transpose exactly the same boundary for both types. Then it has been shown that the nestedness of the subspaces considerably influences the quality of the convergence. It cannot be expected that the unstructured grids be completely nested and the quality of the triangulation of each grid may also damage the convergence.

In this paper, a new semicoarsening algorithm relying on the discretization of the equation, which should enable flexible applications, has been introduced. Convergence rates for highly stretched unstructured meshes have been obtained similar to those for standard Cartesian structured nonstretched meshes. Finally, linear, hence mesh independent, convergence rates have been demonstrated. The extension of these unstructured semicoarsening techniques to the resolution of the Navier–Stokes equations is planned for the near future.

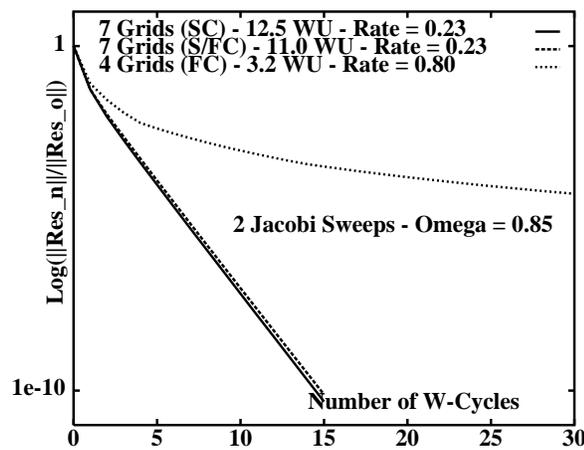


FIG. 16. Multigrid unstructured—resulting convergence histories.

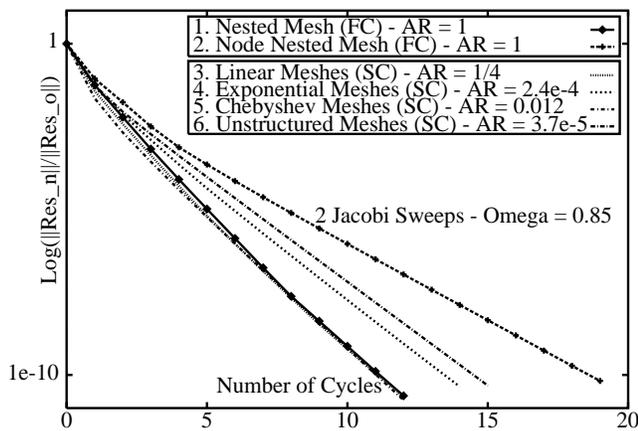


FIG. 17. Significant results.

REFERENCES

- [1] R. BANK AND J. XU, *The hierarchical basis multigrid method and incomplete LU decomposition*, in Domain Decomposition Methods in Scientific and Engineering Computing: Proceedings of the Seventh International Conference on Domain Decomposition, Contemp. Math. 180, AMS, Providence, RI, 1994, pp. 163–173.
- [2] R. E. BANK AND J. XU, *An algorithm for coarsening unstructured meshes*, Numer. Math., 73 (1996), pp. 1–36.
- [3] T. J. BARTH, *Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes*, AIAA Paper 91-0721, AIAA, New York, 1991.
- [4] W. BRIGGS, *A Multigrid Tutorial*, SIAM, Philadelphia, 1987.
- [5] T. CHAN AND B. SMITH, *Domain decomposition and multigrid algorithms for elliptic problems on unstructured meshes*, in Domain Decomposition Methods in Scientific and Engineering Computing: Proceedings of the Seventh International Conference on Domain Decomposition, Contemp. Math. 180, AMS, Providence, RI, 1994, pp. 175–189.

- [6] T. CORMEN, C. LEISERSON, AND R. RIVEST, *Introduction to Algorithms*, McGraw-Hill, New York, 1992.
- [7] J. FRANCESCATTO AND A. DERVIEUX, *Semi-coarsening for agglomeration MG: Application to turbulent flows*, *Internat. J. Numer. Methods Fluids*, 26 (1998), pp. 927–957.
- [8] H. GUILLARD, *Node Nested Multigrid with Delaunay Coarsening*, INRIA Research Report 1898, INRIA, Chesnay, France, 1993.
- [9] W. HACKBUSCH, *Multigrid Methods and Applications*, Springer-Verlag, Berlin, 1985.
- [10] B. KOOBUS, M.-H. LALLEMAND, AND A. DERVIEUX, *Unstructured volume-agglomeration MG: Solution of the Poisson equation*, *Internat. J. Numer. Methods Fluids*, 18 (1994), pp. 27–42.
- [11] M.-H. LALLEMAND, H. STÈVE, AND A. DERVIEUX, *Unstructured multigridding by volume-agglomeration: Current status*, *Comput. & Fluids*, 21 (1992), pp. 397–433.
- [12] M. P. LECLERCQ AND B. STOUFFLET, *Characteristic multigrid method application to solve the Euler equations with unstructured and unnested grids*, *J. Comput. Phys.*, 104 (1993), pp. 329–346.
- [13] D. MARCUM AND N. WEATHERHILL, *Unstructured Grid Generation Using Iterative Point Insertion and Local Reconnection*, AIAA Paper 94-1926, AIAA, New York, 1994.
- [14] D. MAVRIPLIS, *Multigrid solution of the two-dimensional Euler equations on unstructured triangular meshes*, *AIAA J.*, 26 (1988), pp. 824–831.
- [15] D. MAVRIPLIS AND V. VENKATAKRISHNAN, *Agglomeration multigrid for two dimensional viscous flows*, *Comput. & Fluids*, 24 (1995), pp. 553–570.
- [16] E. MORANO AND A. DERVIEUX, *Steady relaxation methods for unstructured MG Euler and Navier-Stokes solutions*, *Internat. J. Comput. Fluid Dynamics*, 5 (1995), pp. 137–167.
- [17] E. MORANO, H. GUILLARD, A. DERVIEUX, M.-P. LECLERCQ, AND B. STOUFFLET, *Faster relaxations for non-structured MG with Voronoï coarsening*, in *First European Computational Fluid Dynamics Conference*, Elsevier, New York, 1992, pp. 69–74.
- [18] W. MULDER, *A new multigrid approach to convection problems*, *J. Comput. Phys.*, 83 (1989), pp. 303–329.
- [19] A. OVERMAN AND J. V. ROSENDALE, *Mapping robust parallel multigrid algorithms to scalable memory architecture*, in *Sixth Copper Mountain Conference on Multigrid Methods*, NASA Conference Publication 3224, Part 2, NASA, Langley, VA, 1993, pp. 635–648.
- [20] K. RIEMSLAGH AND E. DICK, *A multigrid method for steady Euler equations on unstructured adaptive grids*, in *Sixth Copper Mountain Conference on Multigrid Methods*, NASA Conference Publication 3224, Part 2, NASA, Langley, VA, 1993, pp. 527–542.
- [21] J. RUGE AND K. STÜBEN, *Algebraic multigrid*, in *Multigrid Methods*, S. F. McCormick, ed., SIAM, Philadelphia, 1987, pp. 73–130.
- [22] V. VENKATAKRISHNAN AND D. MAVRIPLIS, *Agglomeration multigrid for the three dimensional Euler equations*, *AIAA J.*, 4 (1995), pp. 633–640.
- [23] S. ZHANG, *Optimal-order nonnested multigrid methods for solving finite element equations 1: On quasi-uniform meshes*, *Math. Comput.*, 55, pp. 23–36.