An Unstructured Algorithm for High Reynolds Number Flows on Highly-Stretched Grids

P.I.Crumpton P.Moinier M.B.Giles

January 9, 1998

Abstract

This paper describes a robust and efficient algorithm for the numerical solution of the steady compressible Reynolds-averaged Navier-Stokes equations, with a one equation turbulence model. Highly stretched triangular and tetrahedral unstructured grids are used to integrate the equations to the wall.

An edge-collapse multigrid approach is introduced which is shown to be efficient, robust and applicable to complex geometries in two and three dimensions, for inviscid and viscous flows.

Special attention is paid to the treatment of the highly stretched grids in both the discretization and the edge-collapse multigrid method.

1 SPATIAL DISCRETIZATION

This paper is concerned with solving the steady compressible Reynolds-averaged Navier-Stokes equations in conjunction with the Spalart Allmaras turbulence model [1]. Collectively, these may be expressed as

$$\frac{\partial}{\partial x_i}\mathcal{F}(\mathbf{e_i}, \mathcal{Q}, \nabla\mathcal{Q}) = \mathcal{S}(\mathcal{Q}, \nabla\mathcal{Q}). \tag{1}$$

Here $\mathcal{Q}(\mathbf{x})$ is the vector of conserved variables, $(\rho, \rho u, \rho v, \rho w, E, \tilde{\nu})^T$, where ρ is the density, u, v, w are the velocity components, E is the energy per unit volume and $\tilde{\nu}$ is the turbulence variable in the Spalart-Allmaras model [1]. $\mathcal{F}(\mathbf{n}, \mathcal{Q}, \nabla\mathcal{Q})$ is the flux in the direction of the unit vector $\mathbf{n}$, and $\mathbf{e_i}, i = 1, 2, 3$, are the unit vectors in a Cartesian coordinate system. The source term $\mathcal{S}$ is of the form $(0, 0, 0, 0, 0, g(\mathcal{Q}, \nabla\mathcal{Q}))^T$. [1]

The discretization described here is appropriate for both 2D triangular and 3D tetrahedral grids. [2] Using the finite volume approach, equation (1) is integrated over some control volume, which after the application of the divergence theorem gives the expression

$$\mathcal{R}_j = \frac{1}{V_j}\left(\int_{\partial V_j}\mathcal{F}(\mathbf{n}, \mathcal{Q}, \nabla\mathcal{Q})ds - \int_{V_j}\mathcal{S}(\mathcal{Q}, \nabla\mathcal{Q})dv\right) = 0, \quad \forall j \tag{2}$$

[1] Generally, in this paper calligraphic letters are used to denote analytic functions and variables, whereas Roman letters are used to denote discrete quantities. Bold quantities are vectors in Cartesian coordinates.

[2] In fact, the same code runs on either grid type. This is a major benefit for algorithm development allowing methods to be designed and debugged in 2D, and then applied and tested in 3D.

where V_j is the area or volume of the control volume associated with index j. Here the unknowns are stored at the nodes of a given triangular or tetrahedral grid, and the control volume is the 'median-dual' [2] which is constructed around each node $\mathbf{x}_j$ of the grid. The flux integration in equation (2) is approximated by using pre-computed weights for each edge of the grid, see [3, 4], giving

$$\int_{\partial V_j} \mathcal{F}(\mathbf{n},,) \approx \sum_{i \in E_j} \mathcal{F}(\mathbf{n}_{ij},,)|_{\mathbf{x}=\frac{1}{2}(\mathbf{x}_i+\mathbf{x}_j)} \triangle s_{ij} + \sum_{k \in B_j} \mathcal{F}(\mathbf{n}^b_{kj},,)\Big|_{\mathbf{x}=\mathbf{x}^b_{jk}} \triangle s^b_{kj} \quad (3)$$

where E_j is the set of all nodes connected to node j via an edge, $\mathbf{n}_{ij}$ a unit vector and $\triangle s_{ij}$ a distance (2D) or area (3D) associated with the edge connecting nodes i, j. The second term in the RHS correspond to additional contributions due to boundary faces. B_j is the set of all boundary edges (2D) or faces (3D) connected to node j, with corresponding normal $\mathbf{n}^b_{kj}$ and distance or area $\triangle s^b_{kj}$. The boundary flux is evaluated at point $\mathbf{x}^b$ which is defined as

$$\mathbf{x}^b_{jk} = \frac{1}{2N+2} \sum_{i \in BF_k} (1 + (N+2))\delta_{ik})\mathbf{x}_i \quad (4)$$

where N is the dimension of the problem, BF_k the set of all nodes in boundary face k, and δ the Kronecker delta [2, 3].

The discrete equivalent to equation (2) thus becomes

$$R_j = \frac{1}{V_j} \left(\sum_{i \in E_j} F_{ij} \triangle s_{ij} + \sum_{k \in B_j} F^b_{kj} \triangle s^b_{kj} - S_j V_j \right) \quad \forall j \quad (5)$$

where F_{ij} is the flux in the direction $\mathbf{n}_{ij}$ associated with an edge (i, j), and F^b_{kj} is the flux associated with a boundary face.

It now remains to define the discrete flux functions F and F^b, and then the spatial discretization is complete. The flux $\mathcal{F}$ can be split into an inviscid and viscous part

$$\mathcal{F}(\mathbf{n}, \mathcal{Q}, \nabla\mathcal{Q}) = \mathcal{F}^I(\mathbf{n}, \mathcal{Q}) + \mathcal{F}^V(\mathbf{n}, \mathcal{Q}, \nabla\mathcal{Q})$$

for any unit normal $\mathbf{n}$. The discrete approximation of each of these parts is presented in the next two sections.

1.1 Evaluation of F^I

The inviscid flux discretization is based on the flux-differencing ideas of Roe [5], combining central differencing of the nonlinear invisicd fluxes with a smoothing flux based on one-dimensional characteristic variables. In regions in which the flow is smooth it takes the form

$$F^I_{ij} = \frac{1}{2}\left(\mathcal{F}^I_{ij}(Q_i) + \mathcal{F}^I_{ij}(Q_j) + \frac{1}{3}|A_{ij}|(L_i(Q) - L_j(Q))\right) \quad (6)$$

where $\mathcal{F}_{ij} = \mathcal{F}(\mathbf{n}_{ij},,)$ and $A_{ij} = \partial\mathcal{F}^I/\partial\mathcal{Q}$.

L is an undivided pseudo-Laplacian operator with unit central coefficient, a generalization to unstructured grids of the second difference operator used in many structured grid discretizations [6].

This simplest possible definition of this operator is the following

$$L_j(Q) = \frac{1}{\#(E_j)} \sum_{i \in E_j} (Q_i - Q_j) \,, \tag{7}$$

in which $\#(E_j)$ represents the number of elements in set E_j. A similar smoothing formulation has been used successfully on unstructured grids [4], but it has been found to give poor results in general because of the fact that $L_j(Q)$ is not identically zero when Q is a linear function [7]. Thus, in this work a modified version of the pseudo-Laplacian is used which has the properties of admitting linear solutions and at the same time being suited to highly stretched grids.

The first step is to define

$$\hat{L}_j(Q) = \left(\sum_{i \in E_j} \frac{1}{|\mathbf{x}_i - \mathbf{x}_j|} \right)^{-1} \sum_{i \in E_j} \frac{(Q_i - Q_j)}{|\mathbf{x}_i - \mathbf{x}_j|}.$$

The second step is to define a 'linearly-preserving' pseudo-Laplacian

$$\hat{L}_j^{lp}(Q) = \hat{L}_j(Q) - \nabla Q_j . \hat{L}_j(\mathbf{x}).$$

The calculation of ∇Q_j is approximated using the edge weights

$$\nabla Q_j = \sum_{i \in E_j} \frac{1}{2}(Q_i + Q_j)\mathbf{n}_{ij} \triangle s_{ij} + \sum_{k \in B_j} Q^b \mathbf{n}^b_{kj} \triangle s^b_{kj} \quad , \tag{8}$$

which is exact when Q is a linear function of $\mathbf{x}$. Therefore, by construction, $\hat{L}^{lp}(Q)$ is identically zero if Q varies linearly.

If the first step is omitted, replacing $\hat{L}(Q)$ by $L(Q)$, the resulting discretization is both accurate and robust for inviscid flows [8, 9, 10]. However, for the highly stretched grids used in high Reynolds number viscous flow computations, the anisotropic scaling of the contributions to the pseudo-Laplacian from each edge is required for numerical stability.

Figure 1 demonstrates the effect of the linear preserving modification for an RAE2822 2D airfoil; wiggles in the Mach contours are evident when using $\hat{L}$, but do not appear when $\hat{L}^{lp}$ is used. This extra smoothness in the solution because of the linear preserving dissipation is thought to be especially important for the turbulence model, which uses highly non-linear functions of the solution to identify different regions of the boundary layer.

The formulation above applies when the flow is smooth. To handle shocks, a limiter is introduced for each variable so that the smoothing reverts to first order characteristic upwinding at shocks [11].

1.2 Evaluation of F^V

The viscous flux is approximated half-way along each edge (ie. $\mathcal{F}^V_{ij}$) and then use the usual integration rule around each volume, equation (2), thus giving a consistent finite volume treatment of the inviscid and viscous terms. This requires an approximation of ∇Q at the midpoint of each edge which is obtained by a straightforward average,

$$\overline{\nabla Q}_{ij} = \frac{1}{2} (\nabla Q_i + \nabla Q_j) \quad .$$

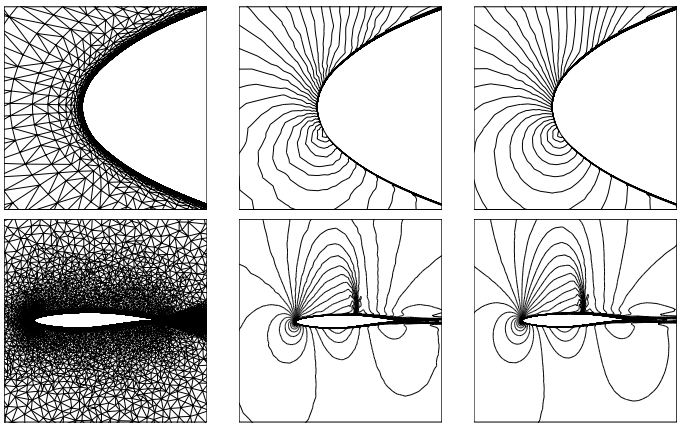


Figure 1: Grid and Mach contours for RAE2822 (case 9)

However, as this is the average of two central differences, it will not damp high frequency modes. Although the inviscid flux includes numerical dissipation terms that will damp these modes, this is insufficient inside the boundary layer where the viscous terms dominate. To remedy this, the component of ∇Q in the direction along the edge is replaced by a simple difference along the edge, giving

$$\nabla Q_{ij} = \overline{\nabla Q}_{ij} - \left(\overline{\nabla Q}_{ij} \cdot \delta \mathbf{s}_{ij} - \frac{(Q_i - Q_j)}{|\mathbf{x}_i - \mathbf{x}_j|} \right) \delta \mathbf{s}_{ij} \tag{9}$$

where

$$\delta \mathbf{s}_{ij} = \frac{\mathbf{x}_i - \mathbf{x}_j}{|\mathbf{x}_i - \mathbf{x}_j|} \quad .$$

Without the addition of the edge-derivative terms the algorithm failed to converge satisfactorily.

2 MULTIGRID METHOD

Multigrid has had a major impact on CFD, and has become an essential part of any successful algorithm. However, the best approach in conjunction with unstructured grids is not yet established. The fundamental concept behind any multigrid method is to have a sequence of successively coarser grids that can represent the "smooth" error modes of the finer grid, while some iterative "smoothing" procedure removes all the high frequency error modes; thus all modes are being dealt with. This, along with transfer operations of restriction (fine to coarse) and prolongation (coarse to fine) defines a multigrid method [12]. The generation of a sequence of grids is achieved, as in [13, 10], by using an edge-collapsing algorithm whose methodology is to remove points to coarsen a given fine grid. This is completely automatic, needing no interaction with any grid generation process. The resulting grid sequence can be used by any grid based algorithm, including those which use an edge based data structure.

2.1 Point-removal

The strategy is to replace two nodes connected by an edge, by a single node at the mid-point of the original edge. The basic edge-collapsing method for an edge i, j in a triangular (2D) or tetrahedral (3D) mesh is summarized below.

1. construct T a unique list of all the cells connected to i and j;
2. construct G the list of boundary faces that appear only once in T;
3. construct S a set of new cells connecting the node $\mathbf{x}^{new}$ to each face in G, where
$$\mathbf{x}^{new} = \begin{cases} \frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j) & \text{if } d_i = d_j \\ \mathbf{x}_i & \text{if } d_i > d_j \\ \mathbf{x}_j & \text{if } d_j > d_i \end{cases}$$
where d_k is the number of boundary surfaces touching node k;
4. if all the volumes in S are positive, replace T with S in the global mesh.

An illustration of this procedure for a 2D case can be seen in Fig. 2.

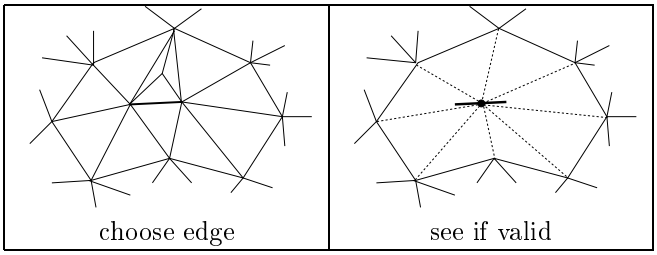


Figure 2: An example of a collapsed edge

This full edge-collapse procedure is applied to a grid in the following manner.

```
do until stopping criterion is satisfied
        color nodes ;
        do over each color
                do over nodes of color
                        if possible, collapse the shortest edge;
                end do
        end do
end do
```

The node coloring is used as a method of removing points evenly across the grid, to prevent all of the nodes being removed from a particular part of the grid. The shortest edge is chosen to make the collapsed grid as isotropic as possible. Isotropic grids, without any highly stretched cells, enable the smoother to damp equally in all direction.

For inviscid grids this has been highly successful [13, 10]. However, for the highly stretched grids required for high Reynolds number flows this procedure was inadequate, since the choice of the shortest edge for edge-collapsing resulted in the highly stretched part of a grid in the boundary layer becoming over-coarsened in the direction across the boundary layer. The procedure adopted is to limit the degree of grid coarsening dependent on the grid stretching. For example, for regions of the grid where the aspect ratio is big, only half the points will be removed,

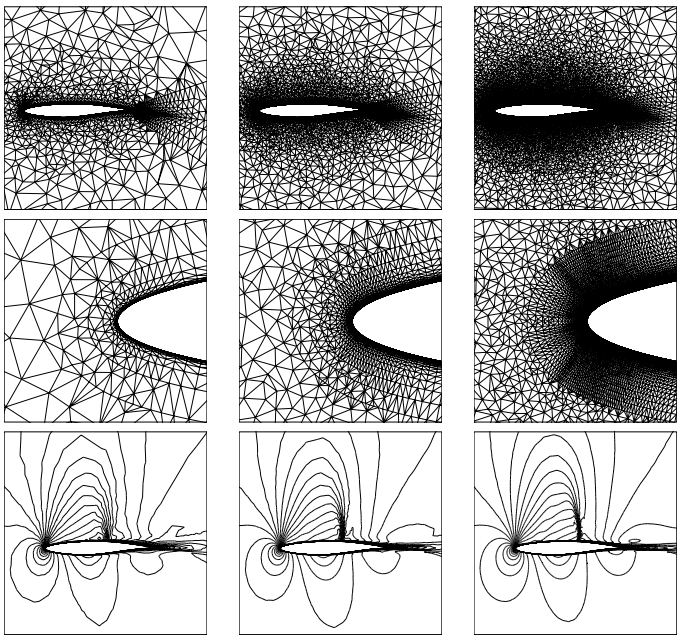


Figure 3: RAE2822 grids (1)-(3) and Mach contours for case 9

approximately, regardless of the dimension of the problem. This is essentially a semi-coarsening strategy, which is common for explicit smoothers on structured grids [12, 14].

The iterative scheme used to converge the discrete residuals to zero is pseudo time-stepping using the 5-stage Runge-Kutta method developed by Martinelli [15].

3 RESULTS

Firstly, we consider the standard 2D RAE2822 airfoil test case 9 [16] ($M_\infty = 0.73$, $\alpha = 2.8$, $Re = 6.5 \times 10^6$). Three grids are considered, and each grid is collapsed twice to produce a multigrid grid sequence; all the base grids can be seen in Fig. 3, and details of all grids are summarized in the table below, with the number in parentheses being the ratio of fine to coarse nodes. This ratio is expected to be lower than that for isotropic inviscid grids because of the semi-coarsening strategy.

	number of nodes		
grid	base grid	collapsed once	collapsed twice
(1)	3089	1161(2.7)	541(2.2)
(2)	9030	3563(2.5)	1633(2.2)
(3)	30087	11061(2.7)	4648(2.4)

Figure 4 shows the edge-collapse grid sequence used for grid (2). As expected, where the base grid (2) has high aspect ratio the coarse grid is only halved, and for the essentially inviscid isotropic part of the grid, the number of nodes is reduced

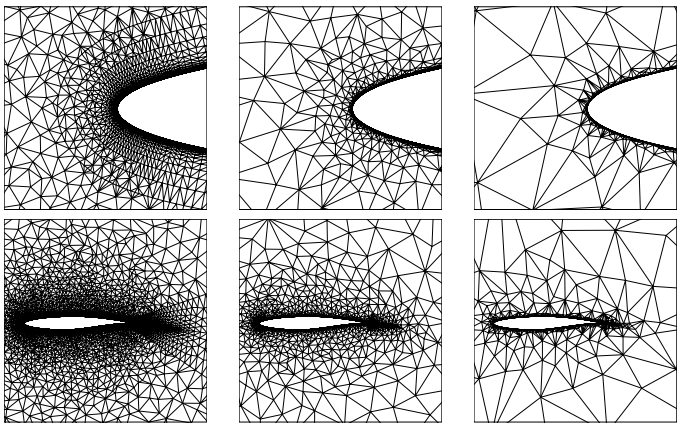


Figure 4: Multigrid sequence used for grid (2)

by three quarters. Since for grids (1)-(3) approximately half the points are of high aspect ratio, then the relative number of nodes on the coarse grid is $\frac{1}{2}\left(\frac{1}{2}\right)+\frac{1}{2}\left(\frac{1}{4}\right)$ giving a fine to coarse ratio of $8/3 \approx 2.7$ which is similar to that achieved.

Grids (1)-(3) have been generated using Delauney insertion [17]. The viscous part of the grid is generated by inserting points in a line normal to the wall in a geometric progression up to 5% of chord away from the airfoil. The first point nearest to the wall is fixed for each grid at 5×10^{-6}, which for a unit chord airfoil at this Reynolds number has been found to give adequate resolution of the laminar sub-layer, that is $y^+ < 1$ for the first point. The grid sequence is generated by halving the chord-wise grid distribution in conjunction to doubling the number of points normal to the wall. Grids (1)-(3) have 13, 25, and 50 points in a line normal to the wall from 5×10^{-6}-to-5% chord respectively. The wake grid is added in order to mimic a C-grid. To avoid any spurious far-field effects, the far field was set 1000 chords away from the airfoil.

Figure 3 shows Mach contours on the grids (1)-(3); a clear qualitative agreement is observed. Figure 5 shows the comparison of pressure coefficient and skin friction with the experimental data of reference [16]. It is evident from the skin friction comparison that the solution from grid (1) does not achieve the experimental friction levels pre- and post-shock. In addition, grids (1) and (2) predict the shock slightly upstream of the experimental data. However, it is worth emphasizing that all these grids are relatively coarse, grids (1)-(3) are equivalent in terms of number of nodes to (128×24), (256×35), (256×117) structured grids respectively. None of these structured grids would be thought particularly fine for a 2D calculation, and it is thought that grid (1) gives good accuracy considering the mesh resolution. In 3D, however, this type of chord-wise grid resolution in grids (1)-(2) is all that can be realistically afforded.

The convergence can be seen in Fig. 6, where multigrid iteration is plotted against log residual. As is often the case with a highly non-linear turbulence equation, the convergence is not monotonic as with the inviscid case. However, a satisfactory level of convergence is achieved, and the asymptotic rates are similar for the three grids. To illustrate the benefits of this multigrid approach over the

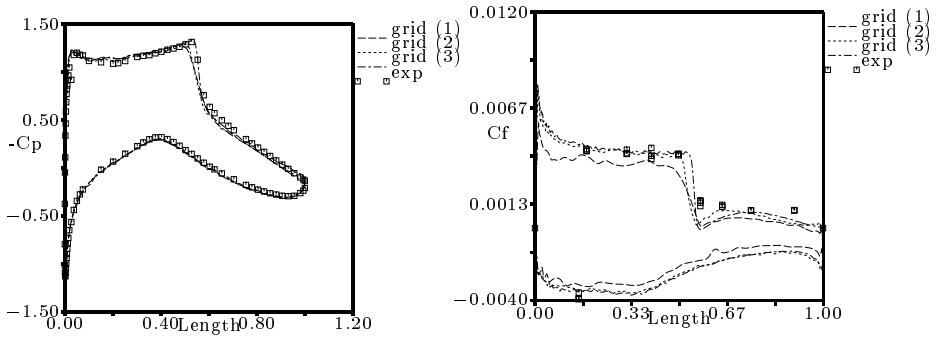
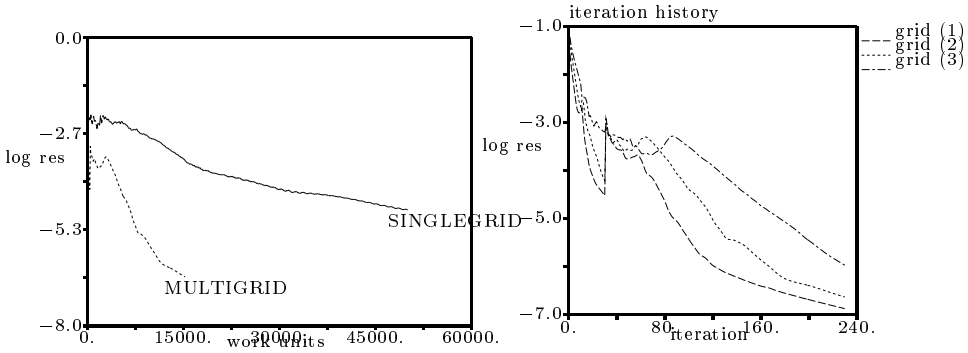


Figure 5: Comparison between solutions on grids (1)-(3) and experimental data [16] for RAE2822 test case 9



grids (2) grid (1)-(3)

Figure 6: Multigrid convergence for RAE2822 test case 9

single grid method, Fig. 6 plots the single and multigrid convergence, against work. Clearly a huge benefit is gained from the multigrid, even though the coarse grids contain many points.

To exemplify the robustness of the method a number of test cases are considered below.

Multi-Element Airfoil Figure 7 shows Mach contour plots and base grid for a multi-element airfoil. The grid is generated in the same manner as for the RAE2822, however, here the various elements of the complete airfoil are so close that the boundary layer grids from neighboring elements intersect. The Delauney method connects the points in what appears to be a highly undesirable fashion. The grid is thought to be very poor in the boundary layer, and wiggles are evident in the Mach plots. However, the solver converged despite both the low free stream Mach number ($M_\infty = 0.1$), and the highly distorted grid.

2D Bypass duct Figure 8 shows a bypass duct of a turbofan engine with Mach contour plot and the grid sequence used for multigrid acceleration. The flow is from left to right, with periodic boundary conditions top and bottom. Here

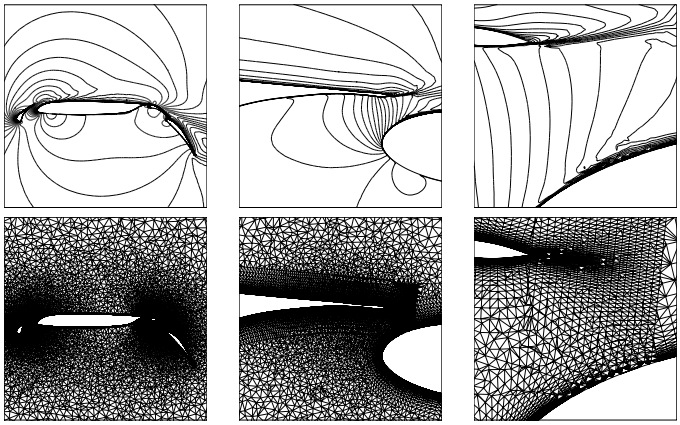


Figure 7: Mach contours and grid for a multi-element airfoil $M_\infty = 0.1$, $\alpha = 10^o$, $Re = 6 \times 10^6$.

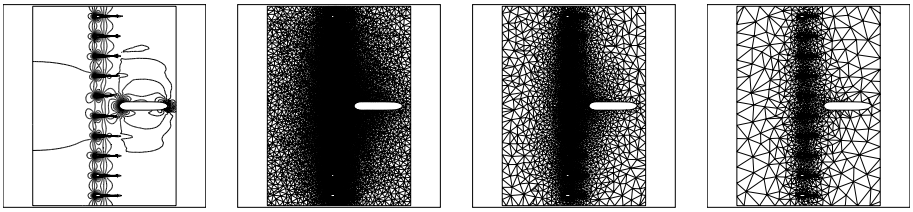
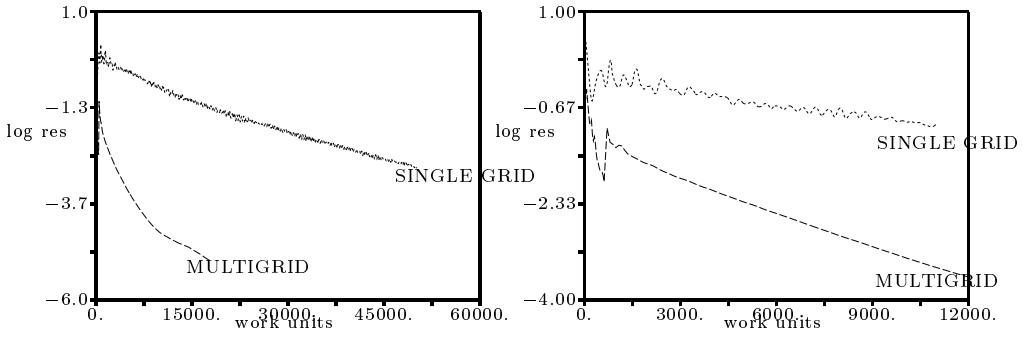


Figure 8: Mach contours and multigrid sequence for the bypass duct geometry, $M_{inlet} = 0.5$, $Re = 10^6$, $\alpha = 0$.

there is a blade row and a pylon; the latter is treated with inviscid boundary conditions to reduce the computational requirements of the calculation, and because the purpose of studying this geometry did not require the pylon boundary layer to be resolved. The grid resolution used for each of the blades in the row is roughly equivalent to that used in grid (2) for the RAE2822 airfoil. The Mach contours reveal that the turbulent wake emanating from the blade row disappears when the grid cannot resolve it. If the location of the wake was of paramount importance then grid adaption would be required. The multigrid sequence of grids illustrates the difference between the coarsening strategy in the isotropic inviscid part of the grid (pylon and surrounding area), and the anisotropic viscous part of the grid (the blade row). A point of interest is the treatment of the periodic boundary condition on the top and bottom of this geometry. The base grid is generated such that the nodes on the periodic boundary conditions have identical x co-ordinates, and $y_{top} - y_{bot} = const.$ The collapse procedure is constrained so that this is also true for the coarse grids. The benefit of using multigrid over the single fine grid iteration can be seen in Fig. 9. Clearly, for this complex geometry a large computational saving has been achieved.

Finally we concentrate on a 3D bypass duct. The grid is constructed by stacking a sequence of 2D grids, which is evident from inspection of the base grid shown in



(2D) (3D)

Figure 9: Multigrid convergence for the bypass duct configuration

Fig. 10. The grid sequence used for multigrid can be seen in Fig. 10; the sizes of the grids are tabulated below.

number of nodes		
# base	collapsed once	collapsed twice
274730	138595(2.0)	79308(1.7)

The ratio of fine to coarse grid nodes is low because the multigrid strategy is using semi-coarsening throughout most of the highly stretched grid. Figure 9 shows work plotted against log residual for the multigrid and single grid case. Clearly, despite the now very expensive multigrid iteration, a huge convergence benefit has been obtained.

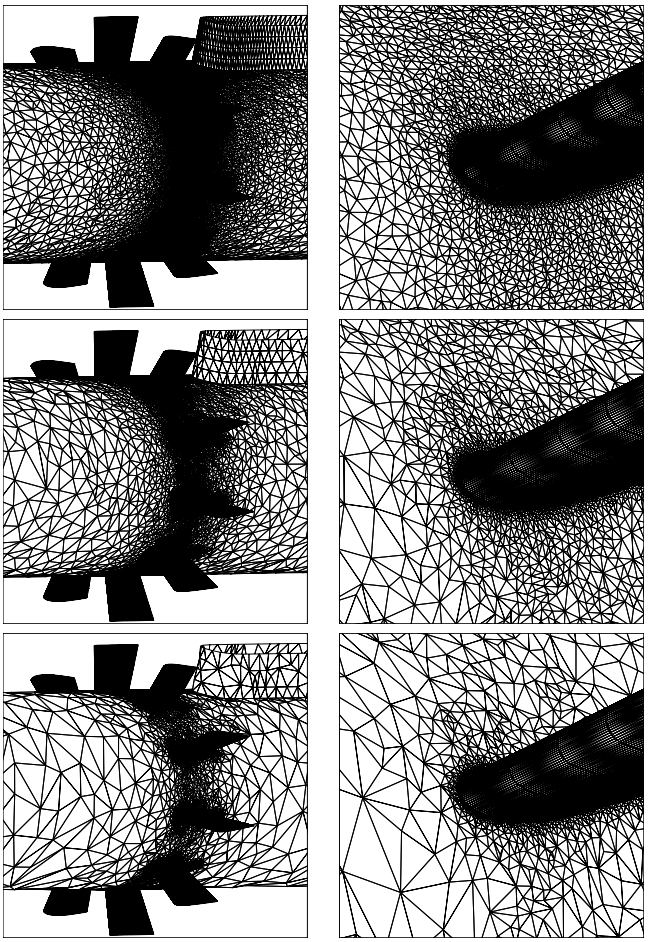


Figure 10: Multigrid sequence for 3D viscous bypass duct problem

4 CONCLUSIONS

A robust and efficient unstructured grid algorithm for both 2D triangular and 3D tetrahedral grids has been developed. The robustness arises from appropriate treatment of the highly stretched grids required for high Reynolds number flows, whilst the efficiency is achieved through an edge-collapse multigrid technique.

References

[1] P. R. Spalart and S. R. Allmaras. A one-equation turbulence model for aerodynamic flows. *La Recherche Aérospatiale*, 1:5–21, 1994.

[2] T. J. Barth. Aspects of unstructured grids and finite-volume solvers for the Euler and Navier-Stokes equations. VKI Lecture Series 1994–05, von Karman Institute for Fluid Dynamics, Belgium, 1994.

[3] K. Morgan, J. Peraire, J. Peiró, and O. Hassan. The computation of three dimensional flows using unstructured grids. *Comput. Methods Appl. Mech. Engrg.*, 87:335–352, 1991.

[4] A. Jameson and D. Mavriplis. Finite volume solution of the two-dimensional Euler equations on a regular triangular mesh. *AIAA Journal*, 24(4):611–618, Apr 1986.

[5] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.*, 43:357–372, 1981.

[6] A. Jameson. Transonic aerofoil calculations using the Euler equations. In P.L. Roe, editor, *IMA Conference on Numerical Methods in Aeronautical Fluid Dynamics*, pages 289–308, London, 1982. Academic Press.

[7] P.I. Crumpton. A cell vertex method for 3d Navier Stokes solutions. Technical Report NA93/09, Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD., 1993.

[8] G.N. Shrinivas and M.B. Giles. Application of sensitivity analysis to the redesign of OGV's. In *Proceedings of the IMECE 1995 Conference*, 1995.

[9] G.N. Shrinivas and M.B. Giles. OGV tailoring to alleviate pylon-OGV-fan interaction. In *Proceedings of the IGTI Turbo Expo*, 1995. ASME paper 95-GT-198.

[10] P. I. Crumpton, M. B. Giles, and G. N. Shrinivas. Design optimisation for complex geometries. 15th ICNMFD Conference, Monterey, USA, June 1996. To appear.

[11] H. Luo, J. D. Baum, and R. Lohner. Edge-based finite element scheme for the Euler equations. *AIAA J.*, 32(6), 1994.

[12] P. Wesseling. *An introduction to multigrid methods.* John Wiley, 1992.

[13] P.I. Crumpton and M.B. Giles. Implicit time accurate solutions on unstructured dynamic grids. AIAA Paper 95-1671, 1995.

[14] N. A. Pierce and M. B. Giles. Preconditioned multigrid methods for compressible flow calculations on stretched meshes. *J. of Computational Physics*, 1996.

[15] L. Martinelli. *Calculations of Viscous Flows with a multigrid method.* PhD thesis, Dept. of Mech. and Aerospace Eng., Princeton University, 1987.

[16] P.H. Cook, M.A. McDonald, and M.C.P. Firmin. Aerofoil RAE 2822 — pressure distributions and boundary layer and wake measurements. AGARD-AR-138, 1979.

[17] A. Bowyer. Computing Dirichlet tessellations. *Computer Journal*, 24:162–166, 1981.