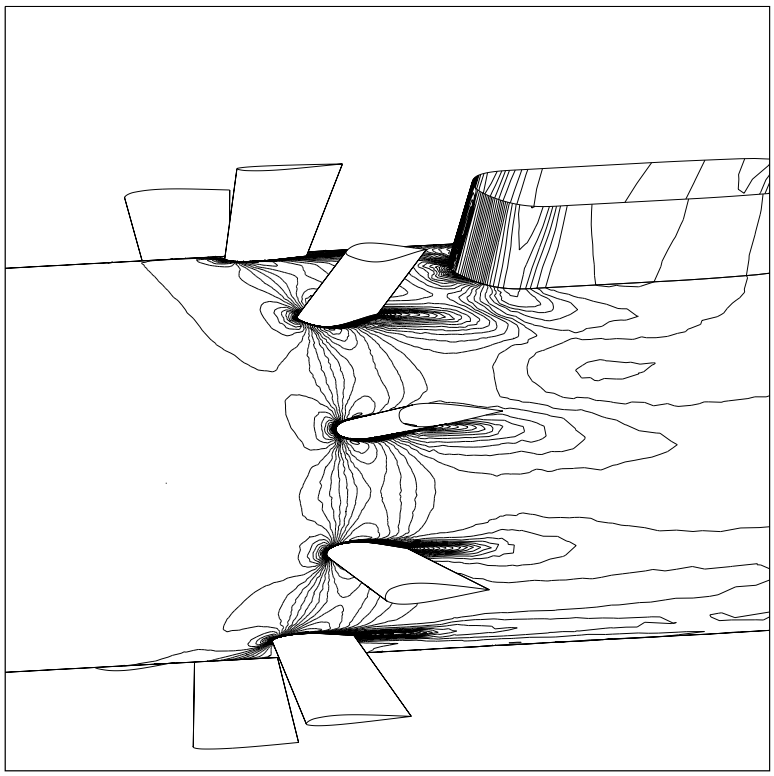
AIAA 99–3339

Edge-based Multigrid and Preconditioning for Hybrid Grids

Pierre Moinier
Jens-Dominik Müller
Michael B. Giles
Oxford University Computing Laboratory
Oxford, United Kingdom OX1 3QD



14th Computational Fluid Dynamics Conference
28 June – 1 July, 1999 / Norfolk VA

Edge-based Multigrid and Preconditioning for Hybrid Grids

Pierre Moinier*
Jens-Dominik Müller†
Michael B. Giles‡
Oxford University Computing Laboratory
Oxford, United Kingdom OX1 3QD

A multigrid method has been developed for the Euler and Navier-Stokes equations on unstructured hybrid grids in two and three dimensions. The coarse grids are automatically generated from the finest grid through element collapsing. This has been used in preference to a previous edge-collapsing technique to preserve as much structure as possible within semi-structured grids. The performance of the multigrid is significantly improved through the use of Jacobi preconditioning within a Runge-Kutta iterative smoother. Results are presented for a variety of two-dimensional and three-dimensional problems, both inviscid and viscous with the Spalart-Allmaras turbulence model.

Introduction

CFD has to respond to the need for accurate, efficient and robust algorithms for solving complete descriptions of fluid motion over complex geometries. When using the Reynolds-averaged Navier-Stokes equations with an appropriate turbulence model, the computational mesh has to be highly resolved in the direction normal to the wall to accurately represent the steep gradients in a high Reynolds number boundary layer. This results in highly stretched computational cells which limit the effectiveness of the numerical algorithms, and increase considerably the size of the problem to solve, both in term of memory requirements and computational cost. With the continuing rapid development of computers, the size of the problems being addressed is becoming ever larger. Therefore, the challenge is to obtain an iterative convergence rate which is grid-independent. Multigrid is the most popular approach to achieve this, and it has been very successful for the Euler equations as well as most elliptic equations. It is also the most attractive approach for solving the Navier-Stokes equations, despite the problems caused by the presence of highly stretched cells in the boundary layer.

The central idea of multigrid is to transfer the low frequency solution errors onto a sequence of coarser meshes where they become high frequency errors that are more effectively smoothed by traditional iterative methods. The choice of the iterative smoother is very important, and a popular explicit multigrid smoother is the semi-discrete scheme proposed by Jameson *et al*[8] which uses multi-stage Runge-Kutta time-stepping with coefficients chosen to promote rapid damping and propagation of error modes. However, for Navier-Stokes computations of high Reynolds number flows, the convective error modes in the boundary layer are not efficiently eliminated because of the high aspect ratio cells inside the boundary layer. The resulting numerical stiffness in conjunction with the stiffness associated with the turbulence model, results in much poorer convergence than obtained with the Euler equations.

To overcome this, one approach is to use a matrix time step, or preconditioner, to cluster the eigenvalues of the residual operator away from the origin into a region of the complex plane for which the multi-stage scheme provides rapid damping and propagation of the corresponding error modes.[1] Pierce and Giles have shown that for turbulent Navier-Stokes calculations on structured grids, the combination of a block-Jacobi preconditioner and a multigrid method with semi-coarsening across the boundary layer provides very effective damping of all modes inside the boundary layer, both in theory and in practice. The preconditioner damps all of the convective modes, while the multigrid strategy, in which the grids are coarsened only in the direction across the boundary layer, ensures that all acoustic modes disappear.[19,20]

However, the task of automatically generating block-structured grids for complex geometries is very challenging. An alternative is to use unstructured grids which are much more easily generated, but using purely tetrahedral grids in 3D leads to reduced accuracy and greater computational cost per grid point. The compromise, which to some extent offers the best of both approaches, is to use hybrid grids in which the grid is treated as an unstructured collection of different cell types (tetrahedra, pyramids, prisms and hexahedra). This gives maximum geometric flexibility, and at

*Graduate Student, email: moinier@comlab.ox.ac.uk
†Postdoctoral Research Fellow, email: jdm@comlab.ox.ac.uk
‡Professor, email: giles@comlab.ox.ac.uk

the same time allows one to use grids which in certain regions are structured or semi-structured, giving improved accuracy and cost per grid point. For example, Kallinderis *et al* have developed viscous grid generation methods which start with a surface triangulation and then use advancing normals to produce a boundary layer grid composed of prismatic elements.[9] These very thin prismatic elements are very suitable for the accurate evaluation of the normal shear stress in the high Reynolds number boundary layer. Outside the boundary layer, the grid reverts to tetrahedra to fill the rest of the domain in which the flow is essentially inviscid.

The drawback of using hybrid grids is that it can complicate the multigrid procedure, depending on the details of the multigrid strategy. If one follows the approach of Mavriplis in using an edge-based discretisation and an agglomeration multigrid which is closely related to algebraic multigrid, the discrete equations on coarser grid levels are assembled automatically without the explicit creation of a coarse grid. For this method, the use of a hybrid grid does not introduce any significant additional difficulties. Agglomeration multigrid is a very powerful approach offering simplicity and robustness, but with the Navier-Stokes equations there is the problem that the sum of the order of accuracy of multigrid restriction and prolongation violates the condition established by Hackbusch[7] (see section Scheme Description) as being necessary for grid-independent convergence. Using an *ad hoc* fix, Mavriplis has nevertheless obtained impressive results,[11–13] but in our research we have preferred the alternative approach of explicitly constructing coarse grids from the finest grid, so that we can achieve first order restriction and second order prolongation in the multigrid.

Our research started from the edge-collapsing multigrid approach introduced by Crumpton and Giles, which has been shown to be highly successful for inviscid flows on tetrahedral meshes.[3,4] For high Reynolds number viscous flows, modifications had to be introduced to prevent the grid in the boundary layer becoming over-coarsened in the direction across the boundary layer. The result is essentially equivalent to a semi-coarsening strategy as used on highly stretched structured grids,[19,23] and it gives a multigrid algorithm which is efficient, robust and applicable to complex geometries in two and three dimensions.[5,14]

The subject of this paper is the extension of the edge-collapsing idea to hybrid grids. This proved to be more difficult than initially expected. Ideally, what one would like is a collapsing technique which preserves as much as possible of the structure within the grid. For example, if the fine grid consists of prisms in the boundary layer, then one would ideally like the first coarse grid to consist of semi-coarsened prisms in the boundary layer. However, Crumpton's edge-collapsing algorithm works by collapsing an edge, combining its two nodes into one, and connecting the new node to the faces of the cavity formed by the removal of the edge and all associated cells. When starting from a prismatic grid, this quickly results in a grid consisting solely of tetrahedra. Even worse, with a hexahedral grid it can result in a coarse grid which has fewer nodes, but an increased number of cells, many of which are tetrahedra.[16]

We will show that these problems are avoided by modifying the algorithm to collapse cells rather than edges, ensuring that the number of cells and edges is always reduced, as well as the number of nodes. The use of heap-based dynamic sorting to select the next cell for collapse, and limits to prevent excess collapsing, lead to patterns of collapse which result in the coarsened grid retaining most of the inherent structure of the original grid in highly stretched regions. The multigrid CFD algorithm is also described and results are presented for a range of test cases to show the multigrid convergence rates which are achieved.

Edge-collapsing Multigrid

The coarser meshes used in the calculations have been generated using an element-collapsing algorithm[16] that primarily considers the graph of edges of the mesh. In this graph, any set of edges can be collapsed if the geometry is still valid after the collapse and none of the neighboring edges exceeds a certain multiple of its original length. The first criterion is obvious, we cannot tolerate negative volumes due to folded grids. The second criterion expresses the design principle of multigrid: the coarsening approximately doubles the length of the edges in a mesh.

In a triangular or tetrahedral mesh, collapsing a single edge removes all elements that are formed with that edge. In a hybrid mesh this is not the case; in order to make an element disappear we may have to collapse several edges. In our current implementation we choose to collapse the shortest edge of an element and the other edges which are topologically "parallel" that is they connect the same two faces. An element collapse then happens by two faces of an element falling onto each other.

The implementation of this algorithm for isotropic meshes is straightforward. Given a fine mesh, we tag each edge with its length times a growth factor, say 2, as maximum length. The elements are sorted in a heap list for smallest volume and we try to collapse the shortest edge and its parallel siblings. Fixing a certain maximum angle for the elements in the collapsed geometry, in the 2D examples 135°, guarantees a minimum quality of the coarser mesh as well as positive volumes. This test is done by looping over all elements that are formed with any of the collapsed vertices and considering the "remaining" element of each. Other edges on these elements may have been collapsed in

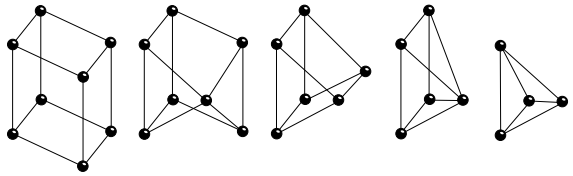


Fig. 1 Collapsing edges on a hexahedron.

earlier steps. E.g. a quadrilateral with one collapsed edge becomes a triangle, a doubly collapsed quadrilateral vanishes. Various collapsed shapes derived from a hexahedron are shown in figure 1. The algorithm terminates once there are no edges left to be collapsed. All remaining elements and nodes are then identified and a coarsened grid is created.

The algorithm has to be modified to achieve directional coarsening in stretched layers. All long edges in stretched regions have to be prevented from collapsing. For this we need to identify short edges in stretched regions. A first criterion is that these edges are shorter by a given factor, say 3, compared to the largest neighboring edge. Additionally we require that there is at least one other neighboring edge that is short and points in the same direction, to within some tolerance. This criterion ensures that single short edges in very irregular unstructured grids do not define a stretched region.

If an element is in a stretched region, all neighboring long edges of the ones to be collapsed are prevented from any collapse. Once the stretched regions have been directionally coarsened in this way, the isotropic process collapses the rest of the domain. Figures 9 and 10 show the results of the grid coarsening for a hybrid grid around a RAE 2822 airfoil. It can be seen that the stretched part of the grid close to the airfoil remains regular and is coarsened exactly 1:2. The outer part of the structured region which is not stretched loses some regularity and the quadrilaterals collapse into larger quadrilaterals and triangles.

Scheme Description

The pre-conditioned semi-discrete equation appears as

$$P^{-1}\frac{dQ}{dt} + R(Q) = 0,$$

where Q denotes the set of conservative variables, $R(Q)$ the residual vector of the spatial discretisation and P^{-1} the local preconditioner which is a point-implicit block-Jacobi preconditioner.[1,15]

Using a finite volume approach, the discrete approximation of the residual for an interior grid point is

$$R_j = \frac{1}{V_j}\sum_{i\in E_j} F_{ij} \triangle s_{ij} \quad \forall j,$$

where V_j is the measure of the control volume (the median-dual) associated with index j, E_j the set of all nodes connected to node j via an edge, $\triangle s_{ij}$ a distance (2D) or area (3D) associated with the edge, and F_{ij} is the numerical flux. In a previous paper,[17] we argued that a desirable feature of discretisations on hybrid grids was that all spatial operators should be 'linear preserving', giving the exact integral for any linear function. However, to obtain this for grids which are not tetrahedral requires the addition of many new edges, and so here we have assumed that in structured or semi-structured regions there is sufficient smoothness in the grid to use the simple median-dual edge weights with negligible loss of accuracy.

At a solid wall, an extra term from the boundary faces associated with the node is added. Tangency and no-slip wall conditions are enforced by zeroing out the momentum components associated with the corresponding wall condition, whereas the far field boundary is treated by adding an extra upwinded flux difference. The inviscid flux discretization is based on the flux-differencing ideas of Roe,[21] combining central differencing of the non-linear inviscid fluxes with a smoothing term based on one-dimensional characteristic variables. This numerical dissipation is a blend of second and fourth characteristic differences with a limiter.[5] The viscous flux is approximated half-way along each edge and uses the usual integration rule around each volume, giving a consistent finite volume treatment of the inviscid and viscous terms. To account for the effect of turbulence, the one equation turbulence model of Spalart and Allmaras[22] is used with a first order spatial discretisation. Other than an implicit treatment of the source term, it is solved using the same 5-stage Runge-Kutta method[10] as used for the flow equations.

The preconditioner P is based on a local linearisation of the 3D Navier-Stokes equations about a uniform flow, and built by extracting the terms corresponding to the central node. As the flux can be split into inviscid and viscous parts, the matrix preconditioner has contributions coming from both, and is written as

$$P_j^{-1} = \left(P_j^I\right)^{-1} + \left(P_j^V\right)^{-1},$$

where the superscripts I and V stand for Inviscid and Viscous, respectively. Even though a high-order method with limiters is used to define the residual, the preconditioner is based on a first order characteristic smoothing. This approximation is acceptable since the resulting matrix timestep will be only underestimated. This is slightly different from the structured approach, where the block-Jacobi preconditioner remains the same for both schemes.[19] The viscous contribution is only calculated for interior grid points since there is no viscous contribution for a node which lies on a adiabatic solid wall. Following a linearizing procedure, a key factor for the implementation is that all cross derivatives are neglected in the precon-

ditioner.

To form the block-Jacobi preconditioner, the inviscid and viscous Jacobians need to be calculated at each node of the grid. However, at the wall, as already mentioned, the viscous Jacobian does not have to be evaluated. In fact, only a no-slip condition has to be satisfied which is achieved by setting all momentum components in the residual to zero. For Euler calculations, the procedure is slightly different. In addition to the corrections made on the residual, the preconditioner is modified at the wall in order that the condition $\mathbf{u}.\mathbf{n} = \mathbf{0}$ is satisfied; $\mathbf{u}$ and $\mathbf{n}$ denote respectively the velocity vector and the unit normal vector to the wall. This is accomplished by re-evaluating the matrix in the coordinate system (x_n, x_{t_1}, x_{t_2}), by using a rotation matrix T from the original (x, y, z) coordinate system to the new one. x_n is the coordinate in the direction normal to the surface and the other two are mutually orthogonal tangential coordinates. Once done, it is transformed back to the original coordinate system. Thus, the original equation becomes

$$\left[P^{-1} - T^{-1}ST(P^{-1} - I)\right] \frac{dQ}{dt} = MR(Q) \quad ,$$

where $M = (I - T^{-1}ST)$ with S the matrix which sets the normal momentum component to zero. $T^{-1}ST$ only involves the unit normal vector, and so is easily constructed.[6]

As explained in the previous section, the multigrid uses a sequence of coarse grids generated from an initial fine grid by an automatic element removal algorithm. This algorithm produces a pointer from each fine grid node to the coarse grid node into which it has collapsed. The multigrid restriction uses this to compute a volume-weighted average of the flow and a limited volume-weighted restriction of the residual. For the prolongation, a linear interpolation is used through the reconstruction of the gradient of the corrections. The accuracy of the transfer operators thus defined is sufficient to guarantee good convergence rates, since it satisfies the necessary relation for ensuring multigrid efficiency[7]

$$\mathcal{O}_P + \mathcal{O}_R > \mathcal{O}_E \quad ,$$

where $\mathcal{O}_P$ and $\mathcal{O}_R$ are defined as the highest degree plus one of the polynomials that are interpolated exactly by the prolongation and restriction operator and $\mathcal{O}_E$ is the order of the differential equation, which equals 2 for the Navier-Stokes equations.

All of the results to be presented were obtained using V-cycle multigrid, with Full Multigrid startup, and one iteration of the Runge-Kutta smoother before restriction and after prolongation, except where otherwise stated. Thus, one multigrid cycle on the finest grid level has a computational cost which is approximately double that of a single 5-stage time-step on the finest grid.

Results

In this section we present results for a set of inviscid and viscous flows over geometries of varying complexity. First, results are presented for a two-dimensional problem. The performance of the collapsing multigrid algorithm with the hybrid approach is compared with previous results obtained by Pierce using structured grids.[18] The test case is a standard transonic NACA0012 case with $M_\infty = 0.8$ and $\alpha = 1.25^o$, giving a strong shock on the suction surface and a weak shock on the pressure surface. The fine grid, shown in Figure 2, contains 20480 quadrilaterals and is exactly the same as used by Pierce. A sequence of 4 coarser hybrid meshes was generated following the element collapsing procedure; the first and last of these is shown in Figures 3 and 4. It can been seen that the coarsening procedure maintains the general topology of the domain, with the coarse grids composed mainly of quadrilaterals. The computed pressure distribution and the convergence history using both methods are presented in figures 5 and 6. Both methods converge similarly to machine accuracy with very little difference in the asymptotic convergence rate.

We next present a calculation over a geometry of increased complexity. It involves the solution of inviscid transonic flow over a business jet. The geometry consists of a half complete aircraft configuration bounded by a symmetry plane. The fine grid has 156000 vertices and 847000 tetrahedra. 2 coarser grids are derived by the element collapsing algorithm and contain respectively, 58500 and 9800 grid points. The collapsing algorithm is based on several criterion driving the collapsing procedure. In this case, the low coarsening ratio between the finest and the first coarser mesh is due to the poor quality of the initial mesh which has elements with a dihedral angle of more than 180^o. The freestream conditions are $M_\infty = 0.85$ and $\alpha = 2^o$. Figures 7 and 8 show the convergence history and the Mach contour plot where the shock patterns are evident. Convergence to machine accuracy is achieved in 249 multigrid iterations.

The first test case involving a turbulent flow is over a single airfoil. The geometry is an RAE2822 airfoil, with $M_\infty = 0.73$, $Re = 6.5$ million, and $\alpha = 2.8^o$. The mesh employed for this computation contains a total of 19100 grid points, and is depicted in Figure 9. This hybrid grid has a structured part, with elements stretched in the direction along the airfoil, the rest of the domain being filled with triangles. Four coarser grids have been generated, the first one is depicted in Figure 10. Figure 12 shows the computed pressure distribution which compares well with the experimental data,[2] although the turbulence model produces a shock location forward of the experimen-

tal location, behavior which has been previously observed.[18,22] The shock induces a separation bubble measuring about 5% of chord. Convergence history is shown, in Figure 11. Here we compare the convergence when using the block-Jacobi preconditioner and when using the standard approach of scalar preconditioning, with semi-coarsened multigrid. Both methods converge to machine accuracy, along with the turbulence model. The Jacobi approach converges quite smoothly and rapidly to engineering accuracy in approximately 90 multigrid iteration, but after 4 orders of magnitude, a severe degradation in convergence occurs, making this method only as efficient as the standard one. This behavior is attributed to the turbulence model whose convergence is asymptotically the limiting factor in both calculations. However, for this case the preconditioned calculation required two smoothing iterations before restriction of the residual and after prolongation of the multigrid correction, whereas the standard approach required three. Thus, the preconditioned calculation is almost 50% faster.

The final example is the flow through the 3D bypass duct of a turbofan engine. The geometry is composed of ten struts and a pylon. The fine grid has 274000 grid points and is constructed by stacking a sequence of 2D grids. Convergence history and Mach contours can be seen in Figures 13 and 14. From the fine grid, 2 coarser grids are produced containing respectively 138000 and 79300 vertices. The coarsening ratio is low because the multigrid semi-coarsening strategy is essentially only removing points in one-dimension in the areas of high stretching, which is both through the boundary layer and radially. The radial stretching is a consequence of the grid being composed of stacked 2D grids with a fixed radial step. This leaves a high aspect ratio in the radial direction in all regions of the 2D grid that have a much smaller mesh spacing than the radial step. For an inflow Mach number of 0.55, with zero incidence and a Reynolds number of 6 million around the struts, convergence to 6 orders of magnitude is reached in 250 multigrid cycles (the pylon is here treated as inviscid, because the purpose of studying this geometry did not require the pylon boundary layer to be resolved).

Conclusions

In this paper we have presented a new multigrid method for the solution of the Euler and Navier-Stokes equations on unstructured hybrid grids. Unlike the agglomeration multigrid method, it involves the construction of a sequence of coarse hybrid grids on which the same residual operator can be applied. The benefit of this approach is that linear corrections can be prolonged exactly, thereby satisfying a key requirement for grid independent convergence of the Navier-Stokes equations.

The coarse grids are generated through an element collapsing procedure which preserves much of the structure in semi-structured grids, and in particular gives semi-coarsening in boundary layer grids which are created by the method of advancing normals.

Numerical results for a variety of two-dimensional and three-dimensional problems show that the multigrid convergence rate is excellent. An important contributing factor is the use of Jacobi preconditioning for the Runge-Kutta iteration, which overcomes much of the numerical stiffness associated with the Navier-Stokes equations in the highly stretched cells in the boundary layer.

Acknowledgements

This research was funded by EPSRC Research Grant GR/L18280, MOD and Rolls-Royce plc, with technical monitor Dr. Leigh Lapworth whose assistance is greatly appreciated. The first author also gratefully acknowledges support from an EC Research Fellowship.

References

[1]S. R. Allmaras. Analysis of a local matrix preconditioner for the 2-D Navier-Stoke s equations. AIAA 93-3330-CP, 1993.

[2]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.

[3]P. I. Crumpton, M.B. Giles, and G.N. Shrinivas. Design optimisation for complex geometries, 1996. Keynote presentation at International Conference on Numerical Methods in Fluid Dynamics.

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

[5]P.I. Crumpton, P. Moinier, and M.B. Giles. An unstructured algorithm for high Reynolds number flows on highly stretched grids, 1997. Tenth International Conference on Numerical Methods for Laminar and Turbulent Flows.

[6]J. Elliott. *Aerodynamic optimization based on the Euler and Navier-Stokes equations using unstructured grids.* PhD thesis, MIT Dept. of Aero. and Astro., 1998.

[7]W. Hackbusch. Multi–grid convergence theory. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Method Proceedings of the Conference Held at Köln–Porz, November 23–27, 1981*, pages 177–219. Springer–Verlag, 1981.

[8]A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the Euler equations by finite volume methods with Runge–Kutta time stepping schemes. AIAA Paper 81-1259, 1981.

[9]A. Khawaja, H. McMorris, and Y. Kallinderis. Hybrid grids for viscous flows around complex 3-D geometries including multiple bodies. AIAA Paper 95-1685, 1995.

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

[11]D.J. Mavriplis. Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes. AIAA Paper 97-1952, 1997.

[12]D.J. Mavriplis. On convergence acceleration techniques for unstructured grids. Technical report, ICASE, 1998. Report No. 98-44.

[13]D.J. Mavriplis and V. Venkatakrishnan. A unified multigrid solver for the Navier-Stokes equations on mixed element meshes. *International Journal of Computational Fluid Dynamics*, 8:247–263, 1997.

[14]P. Moinier and M.B. Giles. Preconditioned Euler and Navier-Stokes calculations on unstructured meshes. In M.J. Baines, editor, *Numerical Methods for Fluid Dynamics VI*, pages 417–424. ICFD, 1998.

[15]E. Morano, M.H. Lallemand, M.P. Leclercq, H. Steve, B. Stoufflet, and A. Dervieux. Local iterative upwind methods for steady compressible flows. In *Third European Conference on Multigrid Methods*, page 227. Springer-Verlag, Berlin 1991, 1990.

[16]J.-D. Müller. Coarsening 3-d hybrid meshes for multigrid methods. In *9th Copper Mountain Multigrid Conference*, 1999.

[17]J-D. Müller and M.B. Giles. Edge-based multigrid schemes for hybrid grids. In M.J. Baines, editor, *Numerical Methods for Fluid Dynamics VI*, pages 425–432. ICFD, 1998.

[18]N.A. Pierce. *Preconditioned Multigrid Methods for Compressible Flow Calulations on Stretched Meshes*. PhD thesis, Oxford University, 1997.

[19]N.A. Pierce and M.B. Giles. Preconditioned multigrid methods for compressible flow calculations on stretched meshes. *Journal of Computational Physics*, 136:425–445, 1997.

[20]N.A. Pierce, M.B. Giles, A. Jameson, and L. Martinelli. Accelerating three-dimensional Navier-Stokes calculations. AIAA Paper 97-1953, 1997.

[21]P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.

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

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

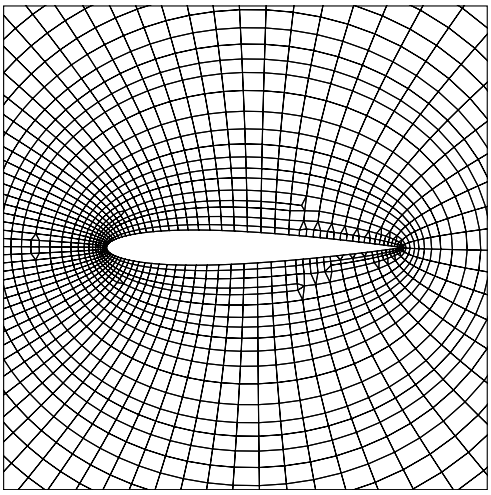


Fig. 3 NACA0012; first coarsening

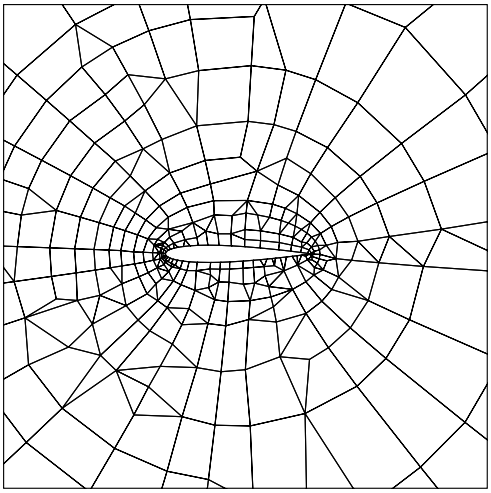


Fig. 4 NACA0012; fourth coarsening

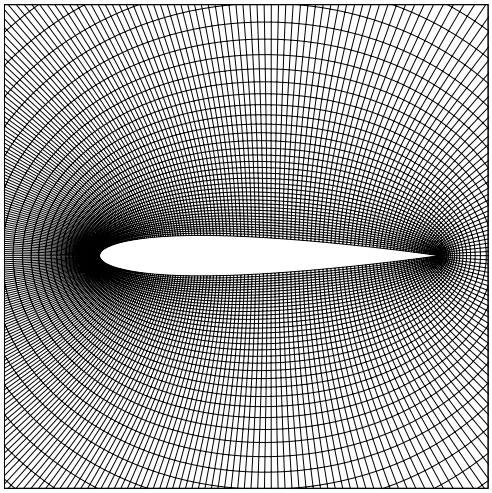


Fig. 2 NACA0012; fine grid

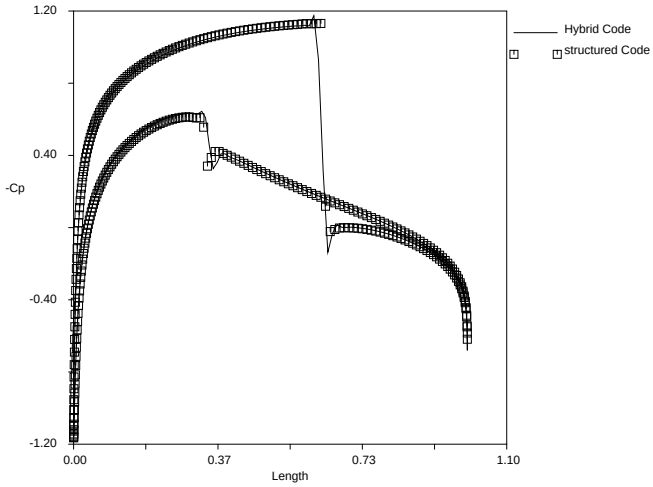


Fig. 5 NACA0012; coefficient of pressure

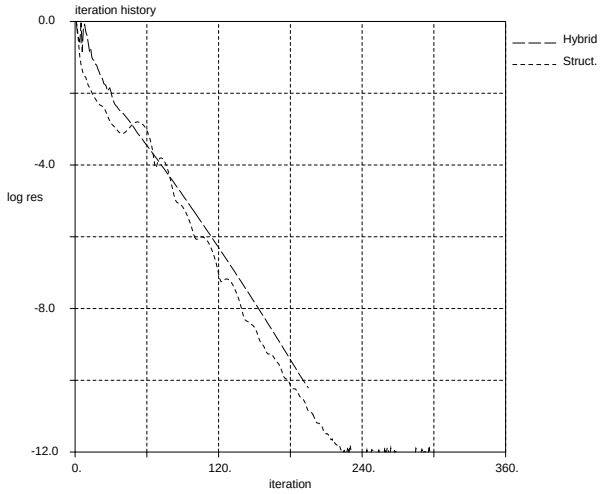


Fig. 6 NACA0012; convergence history

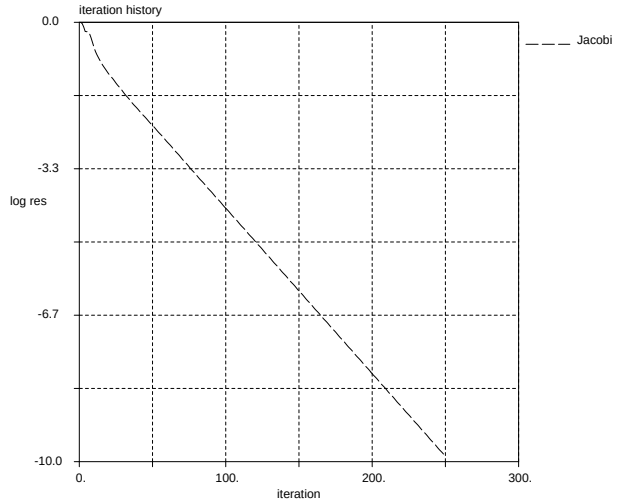


Fig. 7 Business jet, convergence history

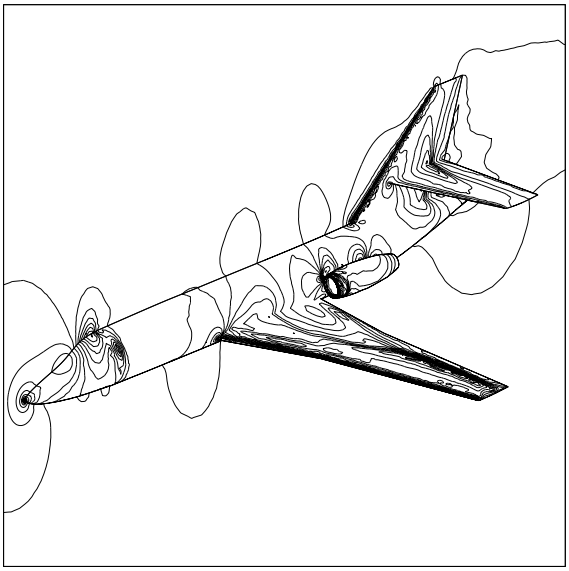


Fig. 8 Business jet, Mach number contours

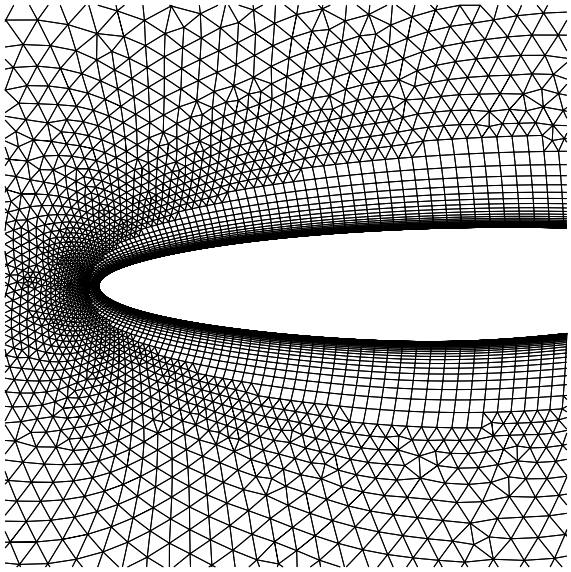


Fig. 9 RAE2822; fine grid.

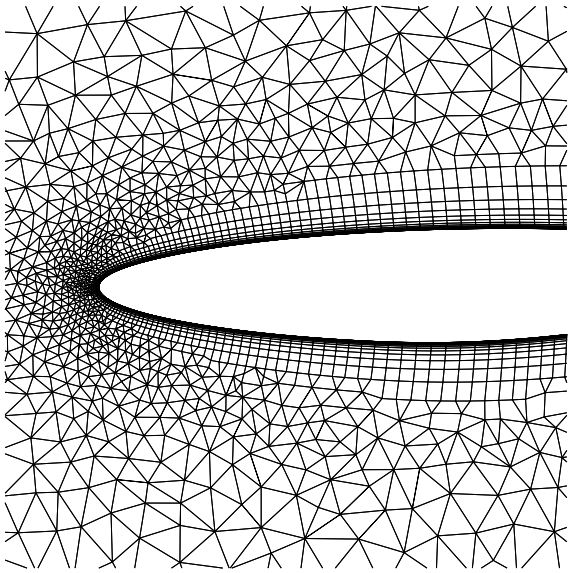


Fig. 10 RAE2822; first collapse

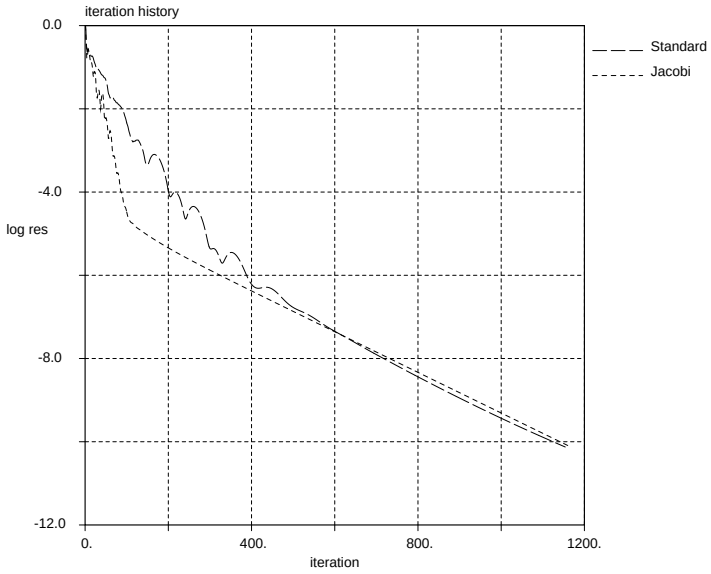


Fig. 11 RAE2822; convergence history

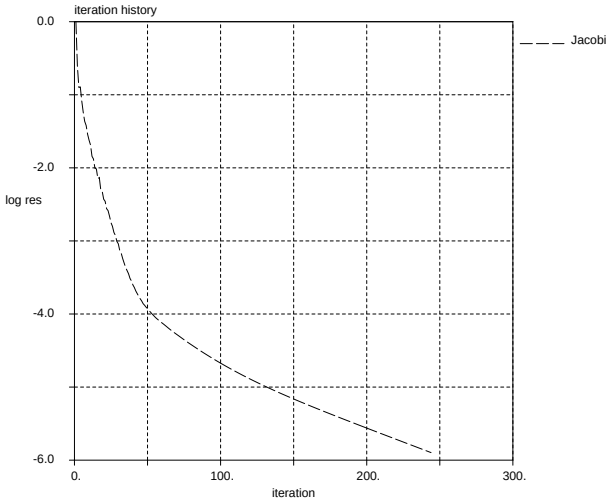


Fig. 13 3D Bypass duct; convergence history

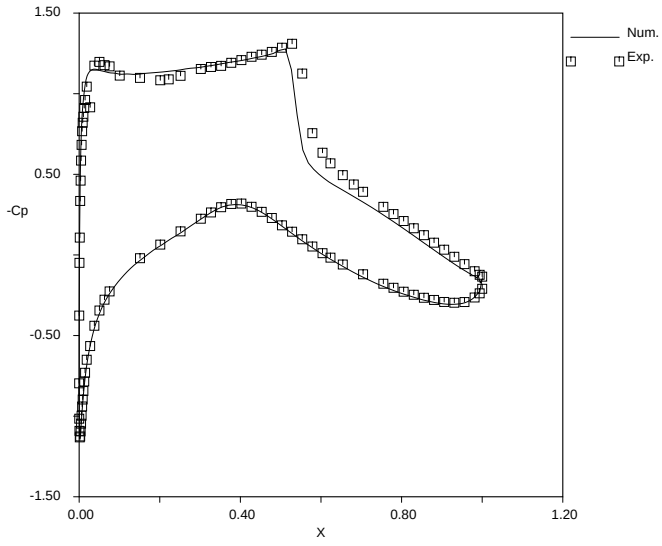


Fig. 12 RAE2822; coefficient of pressure

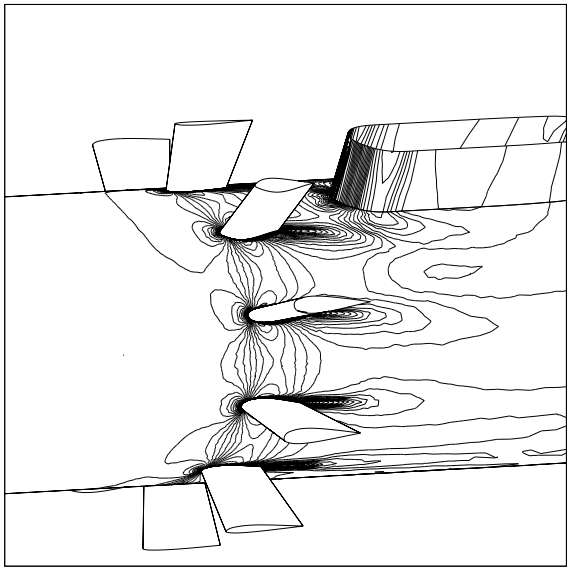


Fig. 14 3D Bypass duct; Mach number contours