

# A Newton-Krylov Algorithm for the Euler Equations Using Unstructured Grids

L. M. Manzano,<sup>\*</sup> J. V. Lassaline,<sup>†</sup> P. Wong,<sup>‡</sup> and D. W. Zingg<sup>§</sup>

*Institute for Aerospace Studies*

*University of Toronto*

*4925 Dufferin St.*

*Toronto, Ontario, Canada M3H 5T6*

**A Newton-Krylov flow solver is presented for the Euler equations on unstructured grids. The algorithm uses a preconditioned matrix-free GMRES method to solve the linear system that arises at each Newton iteration. The preconditioner is an incomplete lower-upper factorization of an approximation to the Jacobian matrix after applying the reverse Cuthill-McKee reordering. The algorithm successfully converges for a wide range of steady two- and three-dimensional aerodynamic flows. A ten-order reduction of the density residual is obtained in a computing time equivalent to fewer than 520 and 1,800 residual evaluations for the two-dimensional and three-dimensional cases, respectively.**

## Introduction

Advances in automated aerodynamic optimization have increased the need for improvements in the speed and reliability of algorithms for computing aerodynamic flows. The Newton-Krylov family of algorithms provides a promising option. Since the work of Wigton et al.,<sup>1</sup> a wide variety of such algorithms has been proposed, including both approximate-Newton<sup>2</sup> and inexact-Newton<sup>3</sup> approaches. The generalized minimal residual (GMRES) Krylov subspace method<sup>4</sup> for nonsymmetric linear systems has been the linear solver of choice. GMRES requires only matrix-vector products, which can be obtained without forming or storing the flow Jacobian matrix in what is termed the matrix-free or Jacobian-free approach. Some approximation to the Jacobian is often formed for use in preconditioning the linear systems. An efficient preconditioner is a key component in a Newton-Krylov algorithm. Preconditioning strategies include lower-upper symmetric Gauss-Seidel (LU-SGS),<sup>5</sup> incomplete lower-upper (ILU) factorization,<sup>3</sup> and use of an existing solver.<sup>1</sup> Newton-Krylov algorithms have been applied to compressible and incompressible flows using both structured and unstructured grids and to other physical problems, such as non-equilibrium radiation diffusion.<sup>1-3,5-12,14</sup>

One of the more efficient Newton-Krylov algorithms was developed by Pueyo and Zingg,<sup>3</sup> who demonstrate a twelve-order reduction in the residual in a computing

time equivalent to fewer than 1000 residual evaluations for a range of two-dimensional inviscid, laminar, and turbulent flows over airfoils. Pueyo and Zingg use an inexact-Newton strategy with matrix-free GMRES and an ILU preconditioner based on an approximation to the flow Jacobian which has been tuned for optimal performance. The start-up phase is handled using an approximate-factorization algorithm on a coarse grid initially, followed by five approximate-factorization iterations on the fine grid to provide a good initial solution for the inexact-Newton algorithm. The algorithm of Pueyo and Zingg was limited to single-block structured grids (single-element airfoils), scalar numerical dissipation, and the Baldwin-Lomax algebraic turbulence model. Nemec and Zingg<sup>15</sup> and Chisholm and Zingg<sup>16</sup> extended the algorithm to multi-block structured grids and the one-equation Spalart-Allmaras turbulence model, thus permitting application to multi-element airfoils. Chisholm and Zingg further incorporated a matrix numerical dissipation model and developed a start-up strategy which does not rely on approximate factorization. The two start-up strategies have yet to be compared, but the approximate factorization approach is restricted in applicability to structured meshes. The algorithm of Nemec and Zingg has been applied extensively to aerodynamic optimization.<sup>17,18</sup>

The objective of this paper is to apply the Newton-Krylov strategy developed by Pueyo and Zingg<sup>3</sup> with the extensions reported in Refs. 15 and 16 to the solution of the two- and three-dimensional Euler equations on unstructured grids. In particular, the start-up techniques developed by Chisholm and Zingg<sup>16</sup> are needed for unstructured grids, since approximate factorization cannot be used. Fast solution of the two-dimensional

---

<sup>\*</sup>Research Engineer

<sup>†</sup>Graduate Student

<sup>‡</sup>Graduate Student

<sup>§</sup>Professor, dwz@oddjob.utias.utoronto.ca, Senior Member AIAA

Euler equations using a similar Newton-Krylov algorithm was reported by Blanco and Zingg<sup>7</sup> for unstructured grids consisting strictly of triangles. Since the present effort aims eventually to solve the Reynolds-averaged Navier-Stokes equations, the current algorithm is applicable to grids consisting of arbitrary polygons in two dimensions and arbitrary polyhedra in three dimensions.

## Numerical Algorithm

### Spatial Discretization

The algorithm employs a cell-vertex finite-volume spatial discretization of the Euler equations. The spatial discretization is applied to the centroidal-median dual constructed from a given source grid. The semi-discrete form of the equations can be written as follows

$$\frac{dQ}{dt} + R(Q) = 0 \quad (1)$$

where the residual operator  $R(Q)$  is formed primarily as face-based operations on the solution vector  $Q$ . Advective contributions to the residual operator are formed by the surface integral of the flux contributions over the boundary of each cell. A blend of first- and third-order matrix artificial dissipation<sup>20</sup> is applied through a combination of Laplacian and bi-harmonic operators controlled by a switch based on pressure.

### Inexact Newton's Method

For steady flows, the discretized system of equations can be written as

$$R(Q) = 0 \quad (2)$$

An inexact Newton method is a generalization of Newton's method, where at each iteration  $k$ , we find  $\Delta Q_k$  in an iterative fashion such that

$$\|R(Q_k) + A(Q_k)\Delta Q_k\| \leq \eta_k \|R(Q_k)\| \quad (3)$$

where  $A(Q_k)$  is the Jacobian of  $R(Q_k)$ . This involves finding  $\Delta Q_k$  such that the residual of the linear problem is reduced by a factor  $\eta_k$ . The solution vector is updated by  $Q_{k+1} = Q_k + \Delta Q_k$ , which is known as a Newton update or outer iteration. The process is repeated until the desired convergence is reached. If  $\eta_k = 0$ , Newton's method is recovered. The inexact-Newton method is motivated by the prohibitive cost of direct solvers when solving large problems. In this paper, the linear system is reduced by  $\eta_k = 0.1$  for all test cases, as suggested by Pueyo and Zingg.<sup>3</sup> Although this approach yields linear, rather than quadratic, convergence, it is very efficient in terms of computing time.

### Linear Iterative Solver

The linear system of equations that arises by applying Newton's method to Eq. 2 is solved using the

GMRES algorithm of Saad and Schultz.<sup>4</sup> The GMRES algorithm requires the computation of matrix-vector products, which can be calculated using first-order forward differences as follows

$$Av = \frac{R(Q + \epsilon v) - R(Q)}{\epsilon} \quad (4)$$

The parameter  $\epsilon$  is calculated at every GMRES iteration as a function of machine zero  $\epsilon_{mz}$  and  $\|v\|_2$ .

$$\epsilon = \frac{\sqrt{\epsilon_{mz}}}{\|v\|_2} \quad (5)$$

This matrix-free approach reduces the memory required to solve the linear system, since the Jacobian matrix does not need to be formed explicitly. Furthermore, it provides an effective linearization even though  $R(Q)$  is not differentiable (since the numerical dissipation includes the absolute value function).

### Preconditioning

The Jacobian matrix of the Euler equations is often off-diagonally dominant and ill-conditioned. As a result, the GMRES algorithm can converge slowly or stall. A preconditioner is used to overcome this difficulty. With right preconditioning, the system  $Ax = b$  becomes

$$AM^{-1}Mx = b \quad (6)$$

where  $M$  is an approximation to  $A$  which is much easier to invert than  $A$ . The idea is that the preconditioner  $M^{-1}$  transforms the original matrix into another one as close as possible to the identity matrix, thus improving the performance of GMRES.

The preconditioner is formed using an approximate Jacobian based on a first-order spatial discretization, which reduces the memory requirements and computational cost. It tends to be more diagonally dominant and better conditioned than the true Jacobian matrix, thereby improving the performance of the linear solver. The coefficient of second-order dissipation in the preconditioner  $\epsilon_2^{PC}$  is formed by combining the second- and fourth-difference dissipation coefficients as follows<sup>3</sup>

$$\epsilon_2^{PC} = \epsilon_2^{RHS} + \sigma \epsilon_4^{RHS} \quad (7)$$

where the superscript RHS denotes the right-hand side, and  $\sigma$  is a user-specified parameter. Increasing the value of  $\sigma$  makes the system more diagonally dominant, but results in an inferior approximation to the true Jacobian. Consequently, large values of  $\sigma$  are robust but possibly slow. Values of  $\sigma$  equal to 5 or 6 are effective for a wide range of flow problems. Note that the approximation given in Eq. 7 is applied to the preconditioner only. Hence it affects neither the convergence of the Newton iterations nor the accuracy of the converged solution.

Incomplete lower-upper factorization with level of fill  $p$  (ILU( $p$ )) is applied to form the preconditioner

for the GMRES algorithm. In an incomplete factorization, the matrix  $M$  is approximated by the product of triangular matrices  $L$  and  $U$  computed using the Gaussian elimination process by dropping elements that exceed a certain level of fill. When the level of fill is zero, the non-zero pattern of the preconditioner corresponds to that of the Jacobian matrix. Both memory and CPU cost to form and apply the factorization increase by allowing a higher level of fill. On the other hand, a high level of fill can increase efficiency and robustness by reducing the number of linear iterations.

### Reordering

The ordering of the unknowns can affect the quality of the incomplete factorization.<sup>21</sup> The reverse-Cuthill-McKee ordering (RCM)<sup>22</sup> has been found to be effective when the root node is chosen to lie on the downstream boundary.

### Start up

Newton's method is susceptible to slow convergence or divergence during the initial iterations, especially for transonic cases. To overcome this difficulty, a time step is added which effectively damps the Newton updates. This also improves the conditioning of the Jacobian, hence reducing the number of inner iterations required during the early Newton iterations. The full Newton method, which corresponds to an infinite time step, is applied once the  $L_2$ -norm of the density residual has been reduced to  $5 \times 10^{-4}$  for the 2D cases and  $8 \times 10^{-4}$  for the 3D cases.

Convergence of Newton's method can also be improved using grid sequencing, which consists of iterating on one or several coarser grids to obtain a better initial guess for the fine grid instead of starting from uniform flow. Two-level grid sequencing is used to converge the ONERA M6 wing test case below. The coarse grid was generated from the fine mesh by agglomerating the fine grid cells using the algorithm described by Zingg and Lassaline.<sup>19</sup> Using grid sequencing, the computing time is reduced by roughly 50% for these cases.

## Results

Several two- and three-dimensional test cases have been studied in order to characterize the speed, accuracy, and robustness of the Newton-Krylov algorithm. These include flows around the NACA 0012 airfoil, the ONERA M6 wing, and the M100 wing-body configuration. Numerical results are compared to a structured solver in 2D and to experimental data in 3D.

### NACA0012 Airfoil

Three subsonic and four transonic test cases were examined. The grid is composed of triangular elements and has 14,193 nodes. Results were obtained using matrix dissipation with  $V_l = V_n = 0.25$ ,  $k_4 = 0.1$ ,  $\sigma = 5.0$  and ILU(3), where  $k_4$  is the fourth-difference dissipa-

tion coefficient, while  $V_l$  and  $V_n$  control the eigenvalues used in forming the dissipation matrix  $|A|$ . Values of  $V_n$  and  $V_l$  equal to unity give scalar (spectral radius) dissipation. The second-difference numerical dissipation coefficient  $k_2$  was set to 5 for the transonic cases and 0 for the subsonic cases.

The residual convergence histories of all cases are shown in Fig. 1 as a function of the CPU time required for one residual evaluation. This work unit is determined by dividing the total CPU time by the CPU time required for a residual evaluation. We use this work unit in order to permit relatively straightforward comparison between different algorithms and different computing hardware. The Mach number and angle of attack of each case are shown in the legend. In all cases, the  $L_2$ -norm of the residual is reduced by 10 orders of magnitude in a CPU time equivalent to less than 520 residual evaluations. Note that the applicability to arbitrary polygons and polyhedra increases the cost of a residual evaluation, thus reducing the apparent cost expressed in terms of equivalent residual evaluations. During the full Newton phase, the GMRES solver took an average of 20 iterations for each outer or Newton iteration.

In Fig. 2, the computed surface pressure coefficient is compared to that from a 2D structured solver, CYCLONE, based on the ARC2D flow solver,<sup>23</sup> for the test case  $M = 0.8$  and  $\alpha = 1.25^\circ$ . The structured solution was obtained on a C-mesh with approximately 89,000 nodes.

### ONERA M6 Wing

Three flows were computed about the ONERA M6 wing:<sup>24</sup> a)  $M = 0.7003$  and  $\alpha = 1.08^\circ$ , b)  $M = 0.7019$  and  $\alpha = 5.06^\circ$ , c)  $M = 0.84$ ,  $\alpha = 3.06^\circ$ . The grid has roughly 235,000 nodes and is composed of tetrahedral cells. Fig. 3 shows a view of a coarser grid with 50,000 nodes. The computed Mach contours for the  $M = 0.84$ ,  $\alpha = 3.06^\circ$  case are depicted in Fig. 4.

The runs are started on a coarse grid from initial conditions with the scalar first-order dissipation scheme ( $V_l = V_n = 1.0$ ,  $k_2 = 1.0$ ,  $k_4 = 0$ ), ILU(2), and a local time step corresponding to a CFL number of 2. After 5 iterations, the CFL number is increased to 500, and the iterations are continued until the  $L_2$ -norm of the density residual is reduced to  $10^{-2}$ . After interpolating the solution to the fine grid, 8 iterations are performed on the fine grid using the first-order scalar dissipation scheme with a CFL number of 2. The CFL number is increased to 500, and the iterations are continued until the  $L_2$ -norm is reduced to  $10^{-2}$ . At this point, third-order scalar dissipation is applied. Five iterations are performed with a CFL number of 5. A CFL number of 30 is then used until the  $L_2$ -norm of the residual is again reduced to  $10^{-2}$ . The final stage of convergence is then performed using matrix dissipation with  $V_l = V_n = 0.25$ ,  $k_2 = 5.0$ ,  $k_4 = 0.1$ ,

$\sigma = 6.0$  and ILU(1). Five iterations are performed with a CFL number of 5, then a CFL number of 100 is used until the residual reaches  $8 \times 10^{-4}$ . The full Newton method is then used until the residual is reduced to  $10^{-10}$ . During the final phase, an average of 18 GMRES iterations are needed per Newton iteration. The convergence histories are shown in Fig. 5. The  $L_2$ -norm of the residual was reduced 10 orders of magnitude in a computing time equivalent to fewer than 1,800 residual evaluations in each case.

To assess the accuracy of the 3D solver, the  $M = 0.84$ ,  $\alpha = 3.06^\circ$  test case was solved on three grids with 50,000, 111,000 and 235,000 nodes. A comparison of the coefficient of pressure for the three grids is presented in Fig. 6. The results agree with the experiment as well as can be expected from an inviscid solution.

### M100 Wing-Body Configuration

A subsonic flow was computed for the M100 wing-body configuration<sup>25</sup> with  $M = 0.6$ ,  $\alpha = 1.733^\circ$ . The grid, illustrated in Fig 7, has approximately 177,000 nodes and is composed of tetrahedral cells. Computed Mach number contours are depicted in Fig. 8.

The start-up strategy is similar to that used for the ONERA M6 wing, except that grid sequencing and first-order scalar dissipation are not needed for this subsonic flow and relatively coarse grid. The scalar dissipation scheme is used until the residual norm is reduced to  $10^{-2}$ , and then the matrix scheme is used. An infinite time step is initiated once the residual norm reaches  $8 \times 10^{-4}$ . The parameters used are  $V_l = V_n = 0.25$ ,  $k_2 = 0.0$ ,  $k_4 = 0.1$ ,  $\sigma = 6.0$ , and ILU(1). As shown in Fig. 9, the density residual is reduced 10 orders of magnitude in a CPU time equivalent to approximately 450 residual evaluations.

## Conclusions

A preconditioned matrix-free inexact Newton-Krylov algorithm for solving the Euler equations on 2D and 3D unstructured grids has been presented and its performance demonstrated for a number of test cases. Specific start-up strategies have been described, including grid sequencing during the start-up phase, which lead to fast and robust convergence. Future work will concentrate on the development of simpler start-up strategies. The results presented are sufficiently promising to justify extension of the Newton-Krylov algorithm to three-dimensional turbulent flows.

## References

<sup>1</sup>L.B. Wigton, D.P. Young, and N.J. Yu, "GMRES Acceleration of Computational Fluid Dynamics Codes," AIAA Paper 85-1494, 1985.  
<sup>2</sup>S.E. Rogers, "A Comparison of Implicit Schemes for the Incompressible Navier-Stokes Equations," AIAA Paper 95-0567, June 1995.

<sup>3</sup>A. Pueyo, and D.W. Zingg, "Efficient Newton-Krylov solver for aerodynamic computations," *AIAA J.*, vol. 36, no. 11, pp. 1991-1997, 1998.  
<sup>4</sup>Y. Saad and M. Schultz, "GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856-869, 1986.  
<sup>5</sup>H. Luo, J.D. Baum, and R. Lohner, "A fast, matrix-free implicit method for compressible flows on unstructured grids," *Journal of Computational Physics*, vol. 146, pp. 664-690, 1998.  
<sup>6</sup>V. Venkatakrishnan and D. Mavriplis, "Implicit solvers for unstructured meshes," *Journal of Computational Physics*, vol. 105, pp. 83-91, 1993.  
<sup>7</sup>M. Blanco and D. Zingg, "Fast Newton-Krylov method for unstructured grids," *AIAA Journal*, vol. 36, no. 4, pp. 607-612, 1998.  
<sup>8</sup>E. Nielsen, W. Anderson, R. Walters, and D. Keyes, "Application of Newton-Krylov methodology to a three-dimensional unstructured Euler code," AIAA Paper 95-1733, 1995.  
<sup>9</sup>P. Geuzaine, "Newton-Krylov strategy for compressible turbulent flows on unstructured meshes," *AIAA J.*, vol. 39, no. 3, pp. 528-531, 2001.  
<sup>10</sup>T. Barth and S. Linton, "An unstructured mesh Newton solver for compressible fluid flow and its parallel implementation," Tech. Rep. 95-0221, AIAA, 1995.  
<sup>11</sup>M. Delanaye, P. Guezaine, J.A. Essers, and P. Rogiest, "A second-order finite-volume scheme solving Euler and Navier-Stokes equations on unstructured grids," AGARD 77th Fluid Dynamics Panel Symposium on Progress and Challenges in Computational Fluid Dynamics Methods and Algorithms, Seville, Spain, Oct. 1995.  
<sup>12</sup>D. Mavriplis, "Parallel performance investigations of an unstructured mesh Navier-Stokes solver," Tech. Rep. 2000-13, ICASE, 2000.  
<sup>13</sup>D.W. Zingg, S. De Rango, and A. Pueyo, "Advances in algorithms for computing aerodynamic flows," Paper 18 in *Frontiers of Computational Fluid Dynamics 2002*, D.A. Caughey, M.M. Hafez, eds., World Scientific, 2002.  
<sup>14</sup>V.A. Mousseau, D.A. Knoll, and W.J. Rider, "Physics-based preconditioning and the Newton-Krylov method for non-equilibrium radiation diffusion," *Journal of Computational Physics*, vol. 160, pp. 743-765, 2000.  
<sup>15</sup>M. Nemeć and D.W. Zingg, "Newton-Krylov algorithm for aerodynamic design using the Navier-Stokes equations," *AIAA J.*, vol. 36, no. 1, pp. 1146-1154, 2002.  
<sup>16</sup>T. Chisholm and D.W. Zingg, "A fully-coupled Newton-Krylov solver for turbulent aerodynamic flows," Paper 333, ICAS 2002 Congress, Toronto, 2002.  
<sup>17</sup>M. Nemeć, D.W. Zingg, and T.H. Pulliam, "Multi-point and multi-objective aerodynamic shape optimization," AIAA Paper 2002-5548, September 2002.  
<sup>18</sup>M. Nemeć and D.W. Zingg, "From analysis to design of high-lift configurations using a Newton-Krylov algorithm," Paper 173, ICAS 2002 Congress, Toronto, 2002.  
<sup>19</sup>J. Lassaline and D. Zingg, "Development of an agglomeration multigrid algorithm with directional coarsening," AIAA Paper 99-3338, June 1999.  
<sup>20</sup>R. Swanson and E. Turkel, "On central-difference and upwind schemes," *Journal of Computational Physics*, vol. 101, no. 1, pp. 292-306, 1992.  
<sup>21</sup>L. Dutto, "The effect of ordering on preconditioned GMRES algorithm, for solving the compressible Navier-Stokes equations," *International Journal for Numerical Methods in Engineering*, vol. 36, pp. 457-497, 1993.  
<sup>22</sup>E. Cuthill and J. McKee, "Reducing the bandwidth of sparse symmetric matrices," in *Proc. 24th National Conference of the Association for Computing Machinery*, Brndon Press, 1980.  
<sup>23</sup>Pulliam, T. H., "Efficient solution methods for the Navier-Stokes equations." Lecture Notes For The Von Karman Institute

For Fluid Dynamics Lecture Series: *Numerical Techniques For Viscous Flow Computation In Turbomachinery Bladings*, Jan. 1986.

<sup>24</sup>V. Schmitt and F. Charpin, "Pressure distributions on the ONERA M6 wing at transonic Mach numbers," *Experimental Data Base for Computer Program Assessment*, AGARD Advisory Report 138, B1, May 1979.

<sup>25</sup>M.P. Carr and K.C. Pallister, "Pressure distributions measured on research wing M100 mounted on an axisymmetric body," *Experimental Data Base for Computer Program Assessment*, AGARD Advisory Report 138, B8, May 1979.

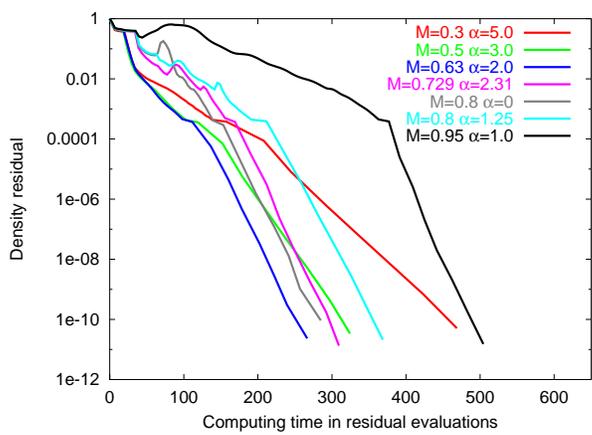


Fig. 1 Residual convergence histories for the NACA 0012 airfoil cases

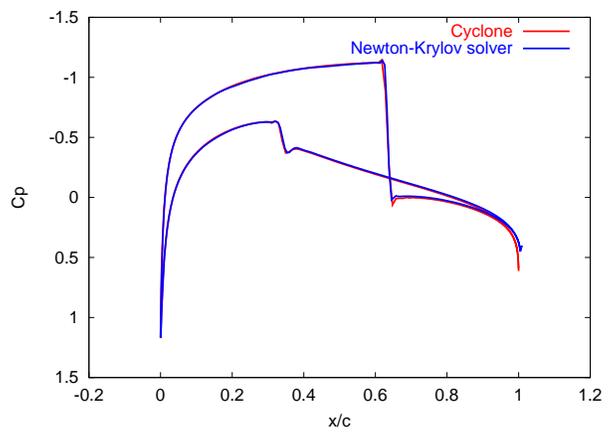


Fig. 2 Comparison of CYCLONE to Newton-Krylov solver for the NACA 0012 airfoil,  $M = 0.8$ ,  $\alpha = 1.25^\circ$

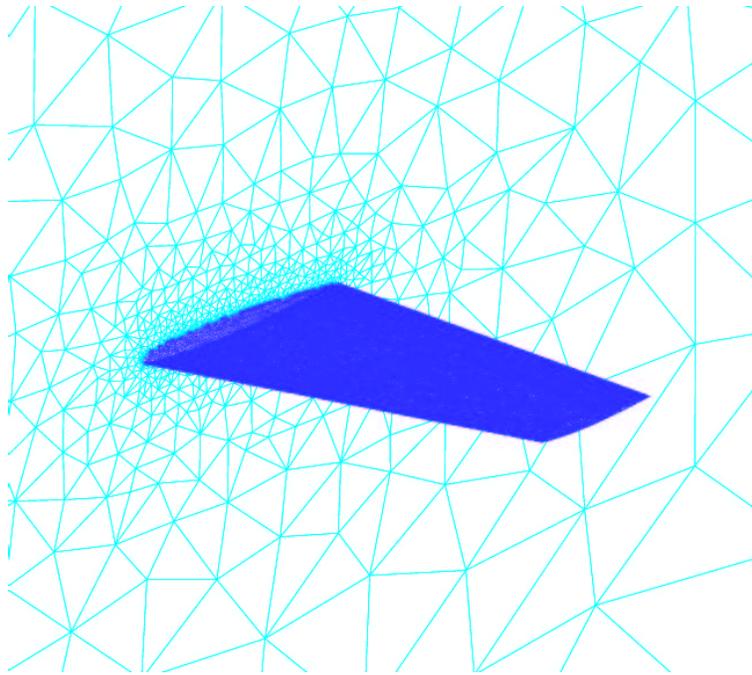


Fig. 3 Close-up of the Onera M6 wing (50,000 nodes)

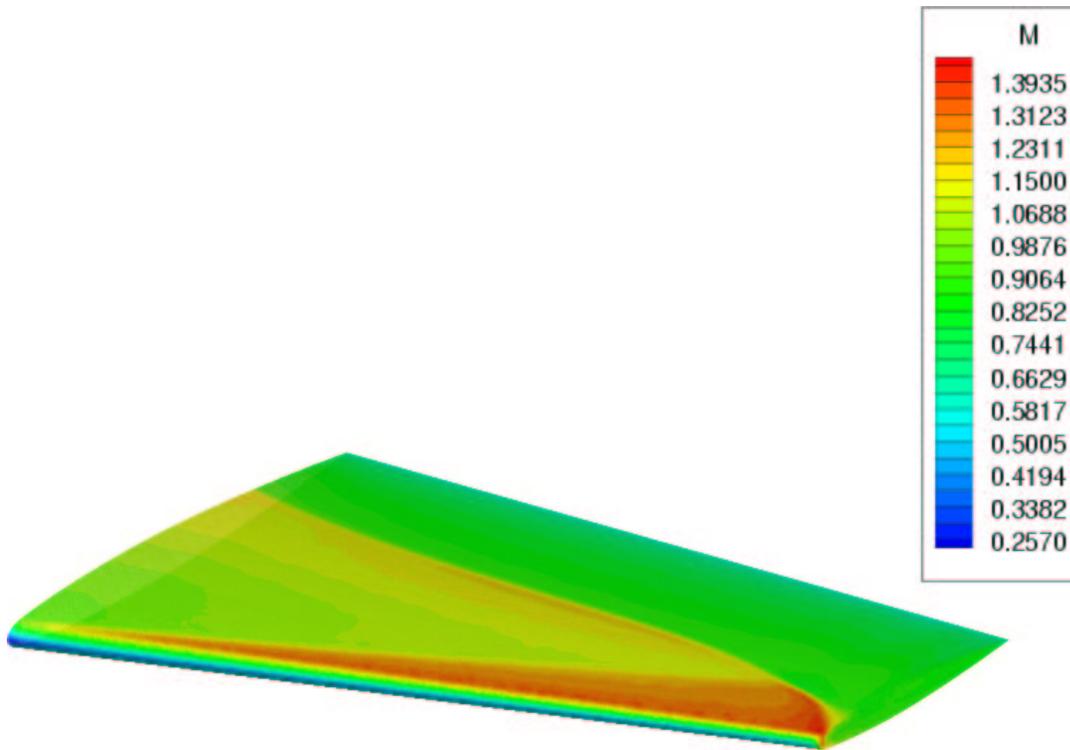


Fig. 4 Mach contours of the ONERA M6 wing for the  $M = 0.84, \alpha = 3.06^\circ$  case

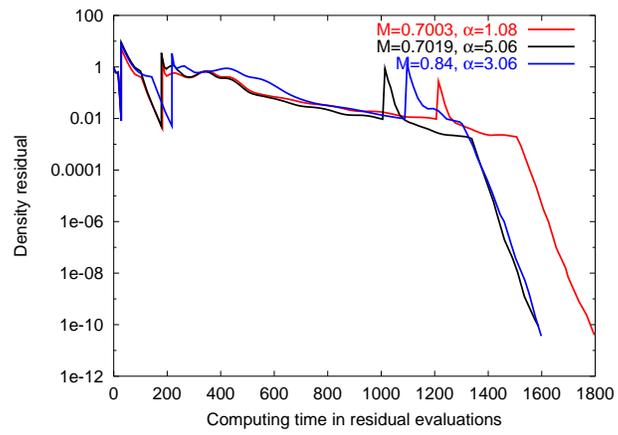
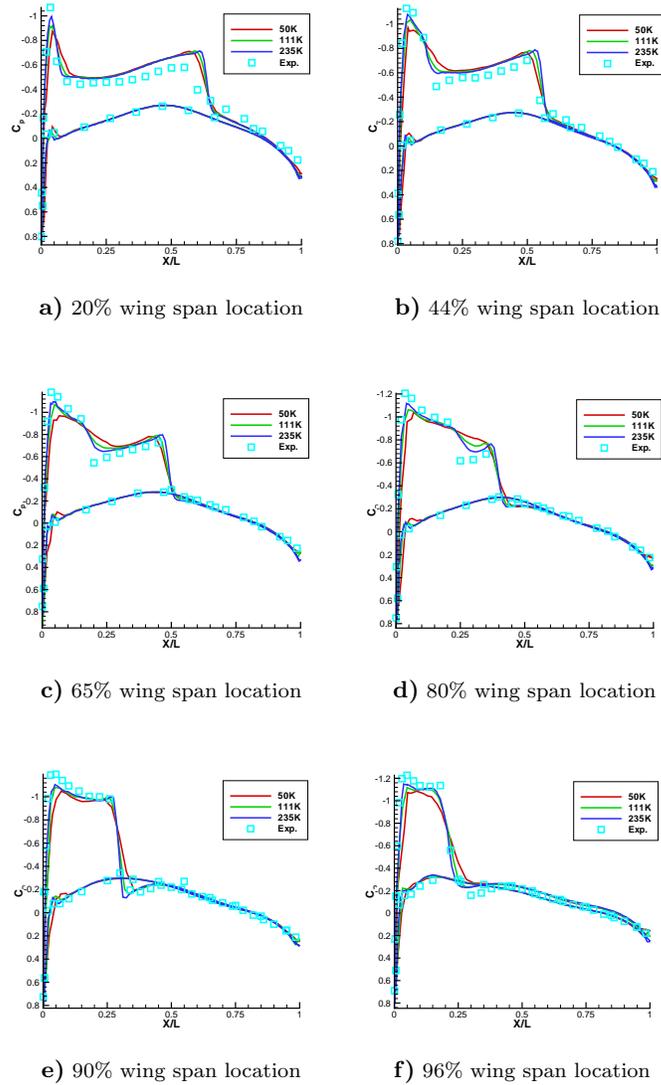


Fig. 5 Residual convergence histories for the ONERA M6 wing cases



**Fig. 6**  $C_p$  plots for the ONERA M6 wing,  $M = 0.84, \alpha = 3.06^\circ$

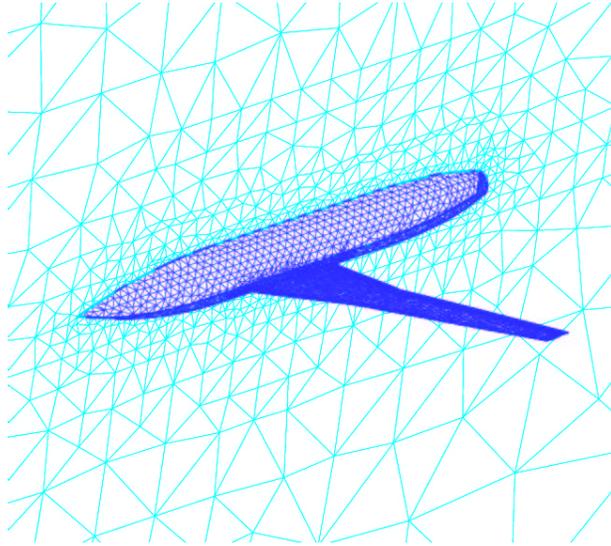


Fig. 7 Grid for the M 100 wing-body configuration

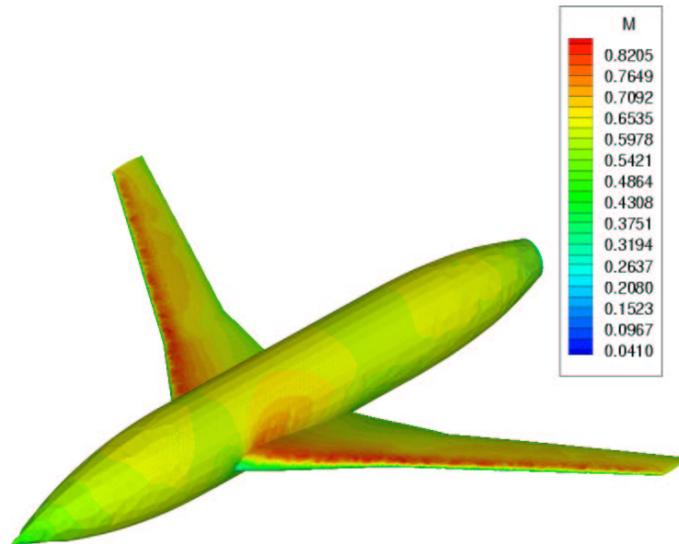


Fig. 8 Mach contours of the M 100 wing-body for the  $M = 0.6, \alpha = 1.733^\circ$  case

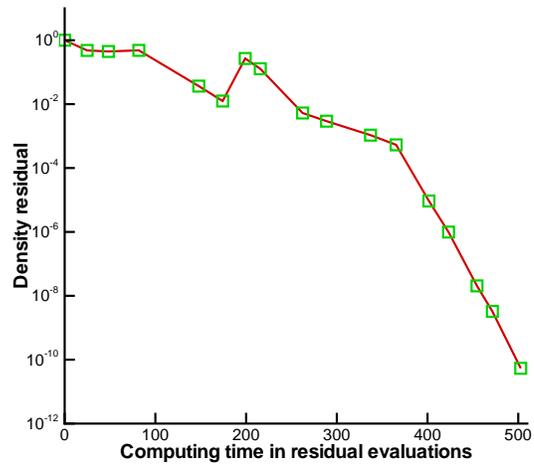


Fig. 9 Residual convergence history for the M100 wing-body configuration. The symbols show the Newton iterations.