A Vertex-Centroid (V-C) Scheme for the Gas-Dynamics Equations

Antony Jameson [*]
Department of Aeronautics & Astronautics
Stanford University
Stanford, CA 94305, USA

John C. Vassberg [†]
Aerodynamics, Phantom Works
The Boeing Company
Long Beach, CA 90807, USA

September 6, 2000

1 Introduction

While the majority of aerodynamic simulations continue to be performed on structured meshes, and the associated algorithms have realized a high level of maturity, mesh generation continues to be a bottleneck in the treatment of really complex configurations. The generation of a structured mesh can span one to two months of elapsed time, depending on the complexity of the geometry. The use of unstructured meshes offers the prospect of substantially reducing the time required to generate a grid about a complex shape. It is difficult, however, to achieve a comparable level of accuracy on an unstructured mesh, especially for Reynolds-Averaged Navier-Stokes (RANS) simulations. Further, the computational costs associated with the indirect-addressing algorithms of an unstructured-mesh solver are substantially higher.

Following classical finite element theory, a number of vertex based schemes have been developed [1, 2]. The upwind schemes which have proved so successful on structured meshes are typically formulated as cell-centered, finite-volume schemes, in which the integral form of the equations is used to update the cell-averaged values of the conserved flow variables. This approach has also been quite widely adopted on unstructured meshes [3, 4, 5, 6]. In order to obtain accurate values at the center of each cell interface, it requires a reconstruction procedure to estimate the gradients in each cell.

In the past, the present authors have developed cell-vertex schemes related to the Galerkin formulation [7, 8, 9]. Both scalar diffusion and upwind biased schemes have been devised in this context [10, 11]. One reason for preferring a cell-vertex scheme over a cell-centered one, is that the number of cells in a tetrahedral mesh is about six times larger than the number of vertices, and in the past, the available computer memory was a significant constraint on the size of the calculations which could be attempted. On modern machines this is no longer a dominating consideration. In this paper we propose a vertex-centroid (V-C) scheme which combines some of the features of cell-vertex and cell-centered schemes. This leads to a neat reconstruction procedure. It also simplifies the inter-mesh transfers of an unstructured-mesh multigrid scheme, in both the fine-to-coarse and coarse-to-fine directions.

[*] T. V. Jones Professor of Engineering
[†] Associate Technical Fellow

2 Description of the Algorithms

2.1 Finite-Volume Scheme

We consider the two dimensional Euler equations for inviscid, compressible flow. Let p, ρ, u, v, E and H denote the pressure, density, Cartesian velocity components, total energy and total enthalphy. For a perfect gas,

$$\begin{aligned} p &= (\gamma-1)\rho(E-\frac{1}{2}(u^2+v^2)), \\ H &= \frac{p}{\rho}, \\ c^2 &= \gamma\frac{p}{\rho} \end{aligned}$$

where c is the speed of sound, and γ is the ratio of specific heats. The Euler equations can be written in integral form

$$\frac{\partial}{\partial t}\int_D w ds + \int_B f dy - g dx = 0 \tag{1}$$

for a domain D with boundary B, where the state and flux vectors are

$$\begin{aligned} w &= (\rho, \rho u, \rho v, \rho E)^T, \\ f &= (\rho u, \rho u^2+p, \rho uv, \rho uH)^T, \\ g &= (\rho v, \rho uv, \rho v^2+p, \rho vH)^T. \end{aligned} \tag{2}$$

We assume the domain to be subdivided into small triangles. Equation (1) is applied directly to each triangle to follow the evolution of the cell-averaged values of the state vector w. This leads to the semi-discrete form

$$\frac{dw_j}{dt}+R_j=0$$

where R_j is the residual for cell j. This is evaluated as

$$R_j=\sum_{sides} h_{jk}$$

where h_{jk} is a numerical estimate of the flux $(f\Delta y - g\Delta x)$ across the edge separating cell j and a neighboring cell k. This is illustrated in Figure 1, which shows a cell numbered 0, and its three neighbors.

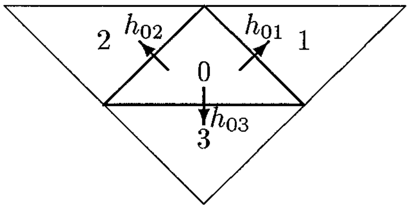


Figure 1: Residual Summation.

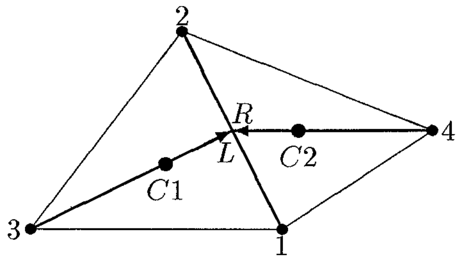


Figure 2: Left-Right Extrapolation.

2.2 Reconstruction

The accuracy of the scheme depends on the accuracy of the estimate of the fluxes h_{jk}. Current upwind schemes [3, 12] reconstruct values of the state vector w at the center of each edge from the cell-averaged values and the gradients in the cells on the left and right side of each edge. These values, which may be denoted as w_L and w_R, are then used to construct the numerical flux taking account of the convection and wave propogation across the edge. In order to estimate the gradients in each cell, a least squares fit is constructed from the values in a set of neighboring cells.

In the present work, we adopt an alternative reconstruction procedure, in which the left and right values are constructed in two steps. First we derive estimates of the state vectors at each vertex, by considering the aggregate conserved values of mass, momentum and energy in the polygon surrounding each vertex.

$$w_{vertex} = \frac{\sum_{cells} w_k A_k}{\sum_{cells} A_k}$$

where A_k is the area of cell k.

This recovery formula does not allow for the fact that the centroid of the polygon does not coincide with the vertex. We have found it necessary to use an alternative procedure to estimate the values at boundary verticies, where the discrepancy is particularly large. An improved recovery procedure which is exact for linearly varying fields will be presented in a forthcoming paper [13]. We anticipate that this will significantly improve the accuracy on irregular meshes.

In the second step, we simply use linear extrapolation along the lines connecting the verticies through the centroid of each cell to the center of the opposing edge, to form the left and right states across each edge. This is illustrated in Figure 2, where the left state across edge 12 is extrapolated from vertex 3 through centroid $C1$, and the right state is extrapolated from vertex 4 through centroid $C2$.

2.3 Limiters

In order to preserve monotonicity in the solution, the left and right values w_L and w_R must be limited so that they do not overshoot the values in the neighboring cells [3]. The results presented in this paper were obtained using a symmetric limiter similar to that proposed by Jameson [10]. Following the notation of Figure 2, define the left and right differences

$$\begin{aligned} \Delta w_L &= w_{C1} - w_3, \\ \Delta w_R &= w_4 - w_{C2}. \end{aligned}$$

Then set

$$\begin{aligned} w_L &= w_{C1} + \frac{1}{2}\mathcal{L}(\Delta w_L, \Delta w_R), \\ w_R &= w_{C2} - \frac{1}{2}\mathcal{L}(\Delta w_L, \Delta w_R) \end{aligned}$$

where the factor $\frac{1}{2}$ accounts for the relative distances along the directions of extrapolation, and $\mathcal{L}$ is a limited average satisfying the following four properties:

$$\begin{array}{rcll} \mathcal{L}(u,v) & = & \mathcal{L}(v,u) & (P1) \\ \mathcal{L}(\alpha u,\alpha v) & = & \alpha\mathcal{L}(u,v) & (P2) \\ \mathcal{L}(u,u) & = & u & (P3) \\ \mathcal{L}(u,v) & = & 0 \ if \ uv < 0. & (P4) \end{array}$$

The limited average used in this work is

$$\mathcal{L}(u,v) = \frac{1}{2}(1 - \mathcal{R}(u,v))(u + v)$$

where

$$\mathcal{R}(u,v) = \left| \frac{u - v}{max((|u| + |v|), tol)} \right|^q$$

and tol is a threshold proportional to the mesh width to the $\frac{3}{2}$ power. Increasing the power q reduces the severity of the limiter. The value $q = 1$ corresponds to the minmod limiter. The calculations presented here were performed with $q = 3$. Alternative one-sided limiters have also been investigated.

2.4 Numerical Flux

The numerical flux is constructed with upwind biasing to take account of wave propogation. The flux across an edge has the form

$$h = f\Delta y - g\Delta x.$$

The Jacobian $\frac{\partial h}{\partial w}$ has the eigenvalues $Q, Q, Q \pm cs$, where s is the edge length, c is the speed of sound, and Q is the rate of convection across the edge,

$$Q = u\Delta y - v\Delta x.$$

Following Roe [14], we can introduce a matrix A_{LR} which exactly satisfies the relation

$$h_R - h_L = A_{LR}(w_R - w_L)$$

where

$$h_R = h(w_R), \quad h_L = h(w_L).$$

Then one can take the eigenvalues of A_{LR} to represent the eigenvalues of $\frac{\partial h}{\partial w}$ across the interface. Following Jameson [11] three families of numerical flux of increasing complexity can be identified:

(1) Scalar diffusion:

$$h_{LR} = \frac{1}{2}(h_R + h_L) - \frac{1}{2}\alpha(w_R - w_L) \tag{3}$$

where the diffusion coefficient should be no smaller that the largest wave speed

$$\alpha \geq Q + cs.$$

This is similar to the diffusion used in the Jameson-Schmidt-Turkel scheme [15] and has also been advocated by Tadmor [16].

(2) Schemes forming diffusion from differences of both flux and state vectors:

$$h_{LR} = \frac{1}{2}(h_R + h_L) - \frac{1}{2}\alpha(w_R - w_L) - \frac{1}{2}\beta(h_R - h_L). \tag{4}$$

In order to produce full upwinding in supersonic flow, we have: $\alpha = 0$ and $\beta = sign(M)$. In a subsonic flow, a blend is required to contain the appropriate domain of dependence.

(3) Characteristic diffusion:

$$h_{LR} = \frac{1}{2}(h_R + h_L) - \frac{1}{2}|A_{RL}|(w_R - w_L) \tag{5}$$

where the diffusion matrix $|A_{RL}|$ is formed by decomposing A_{RL} in terms of its eigenvectors and eigenvalues, and replacing the eigenvalues with their absolute values.

In this work, we have tested both scalar diffusion and the H-CUSP scheme, which defines coefficients in Equation 4 such that in one-dimensional flow the numerical shock structure contains a single interior point [11]. Results are presented for the H-CUSP scheme.

2.5 Discretization of Navier-Stokes Viscous Terms

Following the same philosophy of an interplay between nodal and cell-centered quantities, the V-C scheme can be readily extended to treat the Navier-Stokes equations for viscous flow calculations. First the velocity derivatives, $\frac{\partial u_l}{\partial x_m}$, are evaluated at the grid vertices by applying the Gauss theorem

$$\int \frac{\partial u}{\partial x} dA = \oint u dy$$

to a control volume around each interior vertex, as illustrated in Figure 3. The control volume is defined by lines through the centroids of the cells, drawn parallel to the outer edge of each cell, with a length equal to $\frac{2}{3}$ the length of the outer edge, and containing $\frac{4}{9}$ the area of the cell. Using a rectangle rule of integration, the derivative is estimated as

$$(D_x u)_V = \frac{\frac{2}{3}\sum u_k \Delta y_k}{\frac{4}{9}\sum A_k} = \frac{3}{2}\frac{\sum u_k \Delta y_k}{\sum A_k}$$

where u_k is the value of the velocity at the centroid, A_k is the triangle area, Δy_k is the outer edge length, and the sum is over the triangles surrounding the vertex.

Modified control volumes are used at boundary vertices, closed by edges along the boundary. The rate of strain tensor and energy flux are then calculated at each vertex. Finally the viscous fluxes for each triangle are evaluated by a trapezoidal rule of integration using the vertex values.

$$\sum_{edges} (\sigma_{xx}\Delta y - \sigma_{xy}\Delta x) \,.$$

2.6 Time Stepping Scheme

Following earlier work by the present authors, a hybrid multi-stage time-stepping scheme is used in which the convective and diffusive terms are treated differently in order to expand the stability region. The residual is split as

$$R(w) = Q(w) + D(w)$$

where $Q(w)$ is the convective part and $D(w)$ is the diffusive part. Then the hybrid multi-stage time-stepping scheme is formulated as

$$\begin{aligned}
w^{(1)} &= w^{(0)} - \alpha_1 \Delta t (Q^{(0)} + D^{(0)}), \\
w^{(2)} &= w^{(0)} - \alpha_2 \Delta t (Q^{(1)} + D^{(1)}), \\
&\dots \\
w^{(m)} &= w^{(0)} - \alpha_m \Delta t (Q^{(m-1)} + D^{(m-1)})
\end{aligned}$$

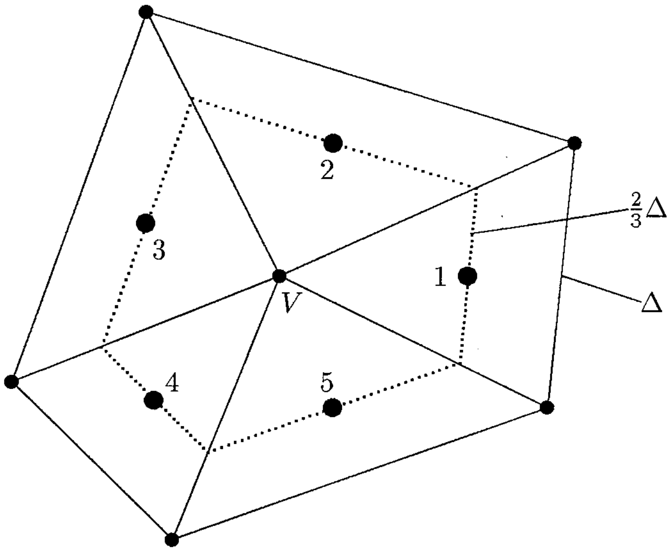


Figure 3: Gradient Polygon about Vertex V.

where $w^{(0)}$ is the value of the state vector at the beginning of the time step, and for an m-stage scheme, $\alpha_m = 1$. Here

$$Q^{(0)} = Q(w^{(0)}), \quad D^{(0)} = D(w^{(0)})$$

and

$$Q^{(k)} = Q(w^{(k)}), \quad D^{(k)} = \beta_k D(w^{(k)}) + (1 - \beta_k) D(w^{(k-1)}).$$

The results presented herein have been calculated with a 5-stage scheme with the following coefficients:

$$\begin{array}{lllll} \alpha_1 = \frac{1}{4}, & \alpha_2 = \frac{1}{6}, & \alpha_3 = \frac{3}{8}, & \alpha_4 = \frac{1}{2}, & \alpha_5 = 1 \\ \beta_1 = 1, & \beta_2 = 0, & \beta_3 = 0.56, & \beta_4 = 0, & \beta_5 = 0.44. \end{array}$$

2.7 Multigrid Scheme

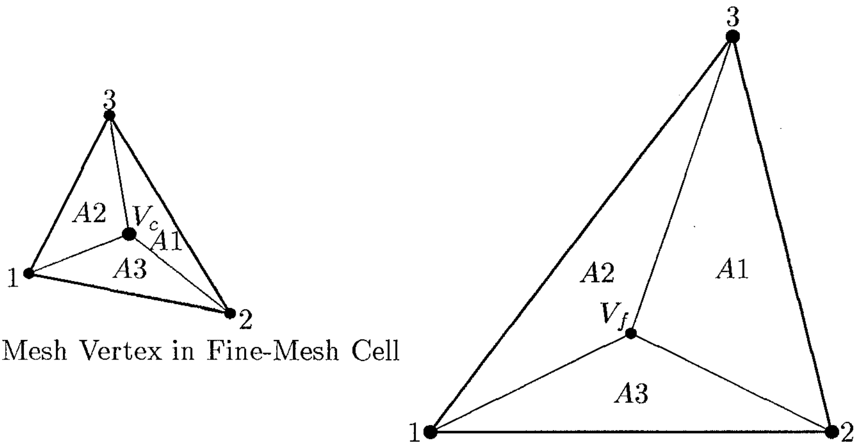
It is well established that multigrid acceleration can drastically reduce the computational costs of flow simulations. With a structured mesh, a sequence of successively coarser grids can easily be generated by eliminating alternate points in each grid direction. With unstructured meshes it is not so easy to generate a suitable sequence of meshes, and in general, the cells of any pair of meshes in the sequence may overlap each other in an arbitrary manner. This complicates the construction of appropriate mesh transfer operators, and is one motivation for the introduction of automated grid-coarsening schemes such as that proposed by Vassberg [17], or the popular agglomeration techniques [18, 19]. Here we simplify the construction of mesh transfer operators by transferring the flow variables from the cell centers to the vertices in an intermediate step.

The multigrid time stepping scheme is similar to that we have previously used [20]. Suppose the grids are numbered consecutively from the finest to the coarsest grid. Then on any grid the following procedure is used. First the solution vector on grid k must be initialized as

$$w_k^{(0)} = T_{k,k-1} w_{k-1},$$

where w_{k-1} is the current state vector on grid $k-1$, and $T_{k,k-1}$ is a transfer operator. Next it is necessary to transfer a residual forcing function such that the solution on grid k is driven by the residuals of grid $k-1$. This can be accomplished by setting

$$P_k = Q_{k,k-1} R_{k-1}(w_{k-1}) - R_k\left(w_k^{(0)}\right),$$



(a) Coarse-Mesh Vertex in Fine-Mesh Cell

(b) Fine-Mesh Vertex in Coarse-Mesh Cell

Figure 4: Area Stencils for Transfer Operators.

where $Q_{k,k-1}$ is another transfer operator. Then $R_k(w_k)$ is replaced with $R_k(w_k) + P_k$ in the time-stepping scheme. Thus the multi-stage scheme is reformulated as

$$w_k^{(q)} = w_k^{(0)} - \alpha_q \Delta t_k \left(R_k^{(q-1)} + P_k \right).$$

The result $w_k^{(m)}$ then provides initial data for grid $k+1$. Finally, the accumulated correction on grid k has to be transferred back to grid $k-1$ with the aid of an interpolation operator $I_{k-1,k}$.

The fine-to-coarse transfer operators are defined next. The flow variables are first distributed to the vertices with area weighting, as in the reconstruction procedure, and the sum of the residuals in the surrounding triangles is accumulated at each vertex. Next the flow variables are transferred from the vertices of the fine mesh to the vertices of the coarse mesh, by finding the fine-mesh triangle which contains the coarse-mesh vertex, and interpolating the fine-mesh values with area weighting, as shown in Figure 4(a).

$$w_{V_c} = \frac{A_1 w_1 + A_2 w_2 + A_3 w_3}{A_1 + A_2 + A_3}.$$

Also the residuals are transferred by finding the coarse-mesh triangle which contains a given fine-mesh vertex. The residual at the fine-mesh vertex is distributed to the three coarse-mesh vertices with the area weighting of Figure 4(b).

$$R_k = R_k + \frac{A_k R_{V_f}}{A_1 + A_2 + A_3}.$$

Finally the flow variables at the coarse-mesh centroids are formed as the arithematic average of the values at the vertices of each cell

$$w_c = \frac{1}{3}(w_1 + w_2 + w_3),$$

while the residuals are distributed from a vertex to the surrounding cells with the area weighting illustrated in Figure 5. Triangle 1 receives $(A_1 R_{V_c})/(\sum_k A_k)$ from V_c, with similar contributions from its other two vertices. This procedure conserves the sum of the residuals between the coarse and fine meshes.

The coarse-to-fine mesh interpolation operator reverses the process. The coarse mesh corrections are first transferred to the coarse-mesh vertices with area weighting, then the correction at each fine-mesh vertex is interpolated between the three vertices of the coarse-mesh triangle in which it is contained, with area weighting as in Figure 4(b). Finally, the correction in each fine-mesh cell is taken as the arithematic average of the corrections at the three vertices of that cell.

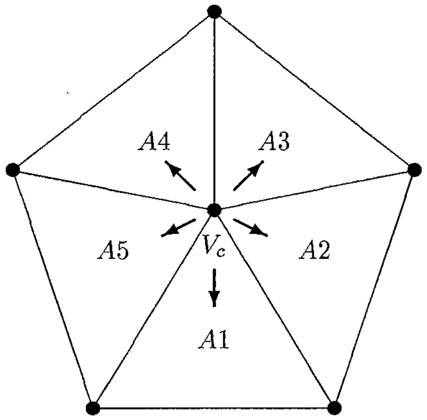


Figure 5: Area Stencil for Residual Re-Distribution.

3 Results

In order to evaluate the accuracy and efficiency of the vertex-centroid (V-C) scheme, we have performed calculations for a variety of airfoils in subsonic and transonic flow. In this section we present results for the standard test case of a RAE-2822 airfoil at a Mach number of $M = 0.75$, and an angle of attack $\alpha = 3.0°$.

In order to assess the dependence of the accuracy on the mesh size, each result is provided for a series of increasingly fine meshes, with respectively (40×9), (80×17), (160×33) and (320×65) nodes; the first number is in the chordwise direction, and the second in the normal direction. In addition, the multigrid scheme with a W-cycle was used to accelerate the calculation on each of these meshes, so a total of 3 grids were used to obtain the result on the (40×9) mesh, while 4, 5 and 6 grids were used, respectively, for the calculations on the successively finer meshes. In every case, the coarsest grid used in the multigrid sequence contained (8×3) nodes.

For all the grids, the mesh nodes were generated via a conformal mapping of the airfoil section to a near circle. Nodes are evenly distributed in the mapped plane, so that they are bunched near the leading and trailing edges in the physical plane. Alternate rows of points were shifted a half-mesh interval in the mapped plane to produce a distribution of points which permits a very regular triangulation. Finally, a constrained Delaunay triangulation algorithm [21, 22] was used to connect the mesh points while preserving the integrity of the profile. Figure 6 shows a close up of the trailing edge region for the four solution grids.

The results of the calculations are presented in the following figures. For each calculation, we show the solution on each of the four meshes. The (40×9) mesh in the top left, the (80×17) mesh in the top right, the (160×33) mesh in the bottom left, and the (320×65) mesh in the bottom right. To represent the solution, we display a plot of the pressure coefficient, $C_p = (p - p_\infty)/(\frac{1}{2}p_\infty q_\infty^2)$, with the axis upward in the negative direction. We also show a porcupine plot of the computed value of a measure of the entropy, $\frac{p}{p_\infty} / \left(\frac{\rho}{\rho_\infty}\right)^\gamma$, along the surface of the profile, with positive values in the outward normal direction. This is a useful measure of the numerical error, as entropy should be generated only through the occurrence of shock waves in the inviscid flow.

As a basis for comparison, we first show in Figure 7 results for the RAE-2822 airfoil computed with the cell-vertex scheme which we have previously used to perform calculations for complete aircraft [7, 8, 9]. This scheme may be viewed as a modified Galerkin method, stabilized by artificial diffusion adapted for unstructured meshes from the Jameson-Schmidt-Turkel (JST) scheme [15]. It is very robust, but exhibits a high level of numerical diffusion, as can be seen from the plots of the entropy along the surface. The multigrid acceleration scheme produces quite a high rate of convergence, dropping the residuals 6 orders of magnitude in 100 cycles.

Next in Figure 8 we show the results of introducing the CUSP numerical flux into the cell-vertex scheme. It can be seen both that the shock waves are captured more crisply, and that the levels of spurious entropy generated ahead of the shock waves are greatly reduced. The convergence rates are slightly slower, but still quite fast, dropping 4 orders in 100 cycles. Since 1993 it has been our normal practice to use the CUSP scheme in preference to the JST scheme for calculations on both structured and unstructured meshes.

In Figure 9 we present results computed with the new vertex-centroid (V-C) scheme using the CUSP numerical flux. It can be seen that the shocks are captured extremely crisply with only one interior point and without any overshoot in pressures. The production of spurious numerical entropy is significantly less than that of the cell-vertex scheme. On the finest mesh, there is no visible signs of entropy production ahead of the shock. However, the convergence rate is somewhat slower, taking almost 500 cycles to drop the residuals by 4 orders of magnitude.

4 Conclusion

We consider the results we have obtained with the V-C scheme to be extremely promising. In fact, the level of spurious numerical entropy production is lower than we have ever previously observed with any scheme on either a structured or an unstructured mesh. Our tests have also shown that the V-C scheme exhibits low numerical entropy production and remarkably crisp shocks when it is stabilized with a simple scalar artificial diffusion.

These results have benefited from the use of very regular triangulations: we believe it is essential to first validate the accuracy of the method with meshes of high quality. Our efforts are now focused on improving the reconstruction method to maintain the highest possible accuracy on less regular meshes [13]. We are also testing the scheme for viscous flows, and for three-dimensional calculations.

5 Acknowledgements

This work has benefited from the generous support of the Air Force Office of Scientific Research under Grant No. AF F49620-98-1-002, as well as from Internal Research and Development support from The Boeing Company.

References

[1] T.J.R. Hughes, L.P. Franca, and M. Mallet. A new finite element formulation for computational fluid dynamics,I, Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Comp. Meth. Appl. Mech. and Eng.*, 59:223–231, 1986.

[2] R. Lohner, K. Morgan, and J. Peraire. Improved adaptive refinement strategies for the finite element aerodynamic configurations. AIAA Paper 86-0499, AIAA 24th Aerospace Sciences Meeting, Reno, NV, January 1986.

[3] T.J. Barth and D.C. Jespersen. The design and application of upwind schemes on unstructured meshes. *AIAA paper 89-0366*, AIAA 27th Aerospace Sciences Meeting, Reno, NV, January 1989.

[4] W.K. Anderson and D.L. Bonhaus. An implicit upwind algorithm for computing turbulent flows on unstructured grids. *Computers and Fluids*, 23:1–21, 1994.

[5] N.T. Frink. Upwind scheme for solving the Euler equations on unstructured tetrahedral meshes. *AIAA Journal*, 30(1):70–77, 1992.

[6] W.A. Smith. Multigrid solution of transonic flow on unstructured grids. Technical report, ASME WAM, Dallas, TX, November 1990.

[7] A. Jameson, T.J. Baker, and N.P. Weatherill. Calculation of inviscid transonic flow over a complete aircraft. *AIAA paper 86-0103*, AIAA 24th Aerospace Sciences Meeting, Reno, Nevada, January 1986.

[8] J.C. Vassberg and K.B. Dailey. AIRPLANE: Experiences, Benchmarks & Improvements. *AIAA paper 90-2998*, AIAA Applied Aerodynamics Conference, Portland, OR, August 1990.

[9] J.C. Vassberg, K.B. Dailey, and D.M. Friedman. AIRPLANE: Unstructured-Mesh Applications. *SAE paper 901857*, SAE International, Long Beach, CA, October 1990.

[10] A. Jameson. Analysis and design of numerical schemes for gas dynamics 1, artificial diffusion, upwind biasing, limiters and their effect on multigrid convergence. *Int. J. of Comp. Fluid Dyn.*, 4:171–218, 1995.

[11] A. Jameson. Analysis and design of numerical schemes for gas dynamics 2, artificial diffusion and discrete shock structure. *Int. J. of Comp. Fluid Dyn.*, 5:1–38, 1995.

[12] V. Venkatakrishnan. A perspective on unstructured grid flow solvers. *Report* 95-3, ICASE, Hampton, VA, 1995.

[13] J.C. Vassberg and A. Jameson. Accurate reconstruction procedure for upwind algorithms on unstructured meshes. *In Preparation*, The Boeing Company, Long Beach, CA, August 2000.

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

[15] 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*, AIAA 24th Aerospace Sciences Meeting, Reno, NV, January 1981.

[16] A. Kurganov and E. Tadmor. New high-resolution central schemes for nonlinear conservation laws and convective-diffusion equations. *J. Comp. Phys.*, 160:241–282, 2000.

[17] J.C. Vassberg. A fast, implicit unstructured-mesh Euler method. *AIAA paper 92-2693*, 10th AIAA Applied Aerodynamics Conference, Palo Alto, CA, June 1992.

[18] D.J. Mavriplis and V. Venkatakrishnan. A 3D agglomeration multigrid solver for the Reynolds-averaged Navier-Stokes equations on unstructured meshes. *Int. J. Numerical Methods in Fluids*, 23:1–18, 1996.

[19] M.H. Lallemand, H. Steve, and A. Dervieux. Unstructured multigridding by volume aggregation: Current status. *Computers and Fluids*, 21:397–433, 1992.

[20] A. Jameson. Solution of the Euler equations for two dimensional transonic flow by a multigrid method. *Applied Mathematics and Computations*, 13:327–356, 1983.

[21] B. Delaunay. Sur la sphere vide. *Bull. Acad. Science USSR VII: Class Scil, Mat. Nat.*, pages 793–800, 1934.

[22] A. Jameson. Notes on constrained Delaunay triangulation for two-dimensional unstructured meshes. *Unpublished Notes*, Princeton University, Princeton, NJ, 1985.

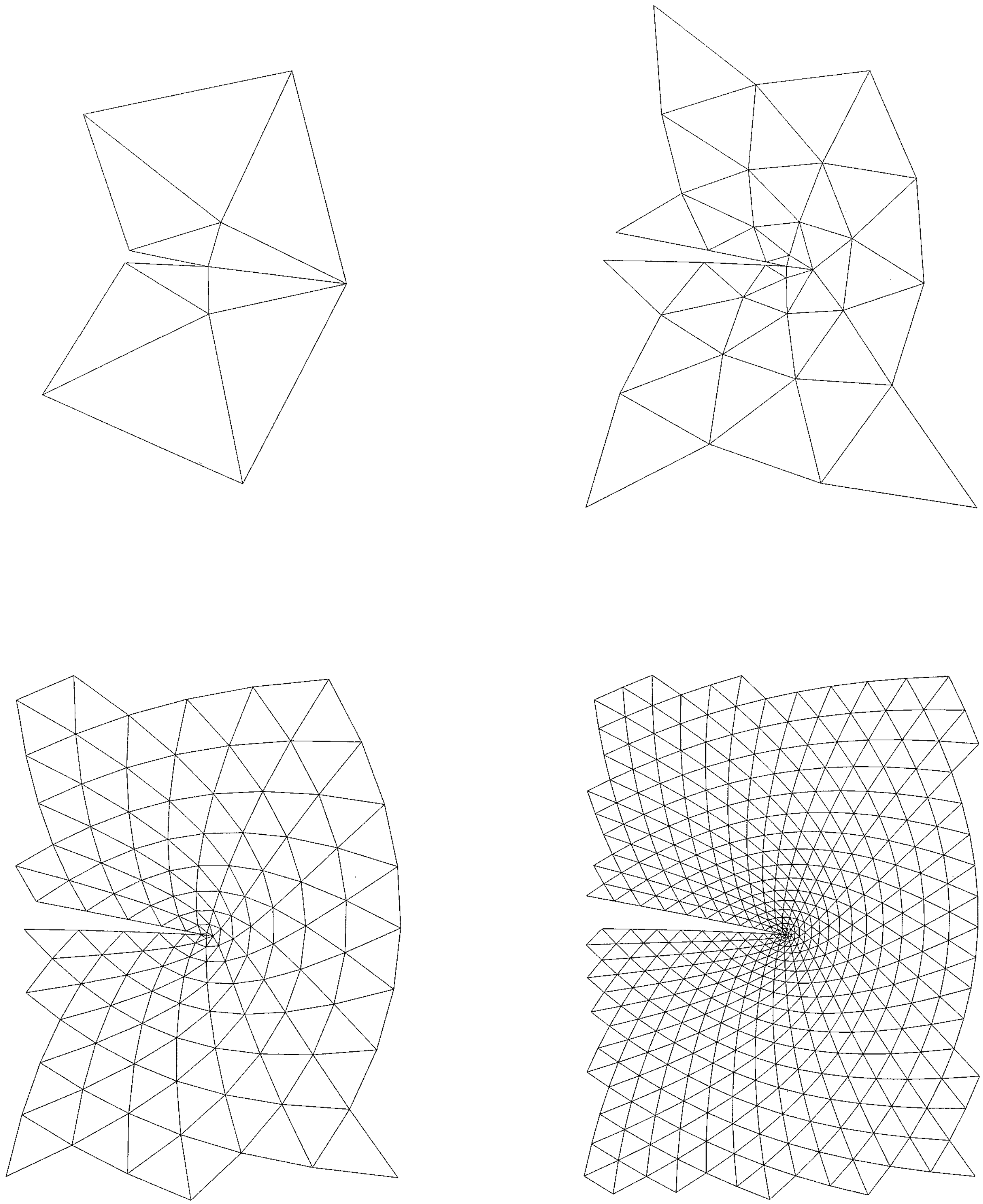
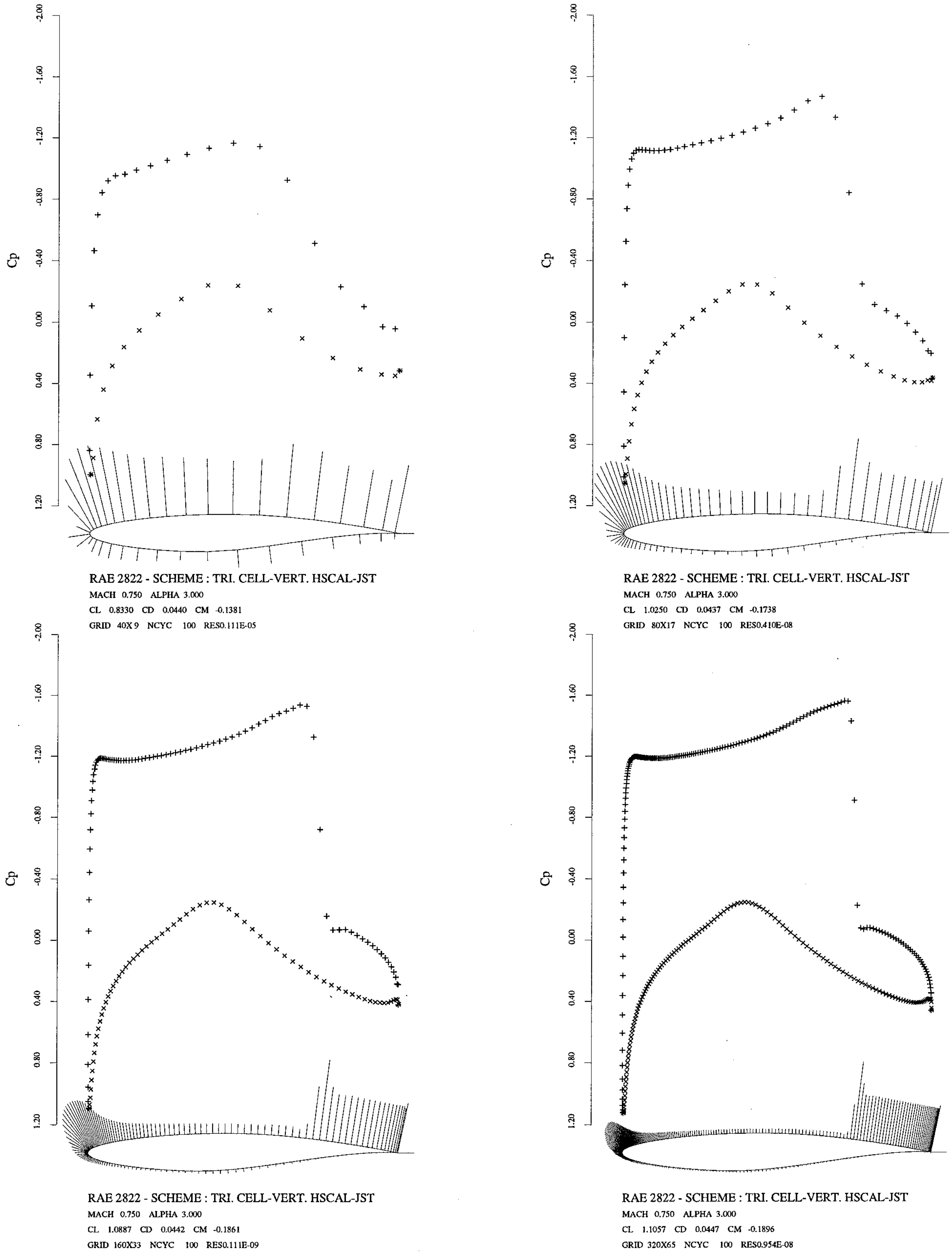


Figure 6: RAE-2822 Grid Series, Close-Up of Trailing Edge Region.



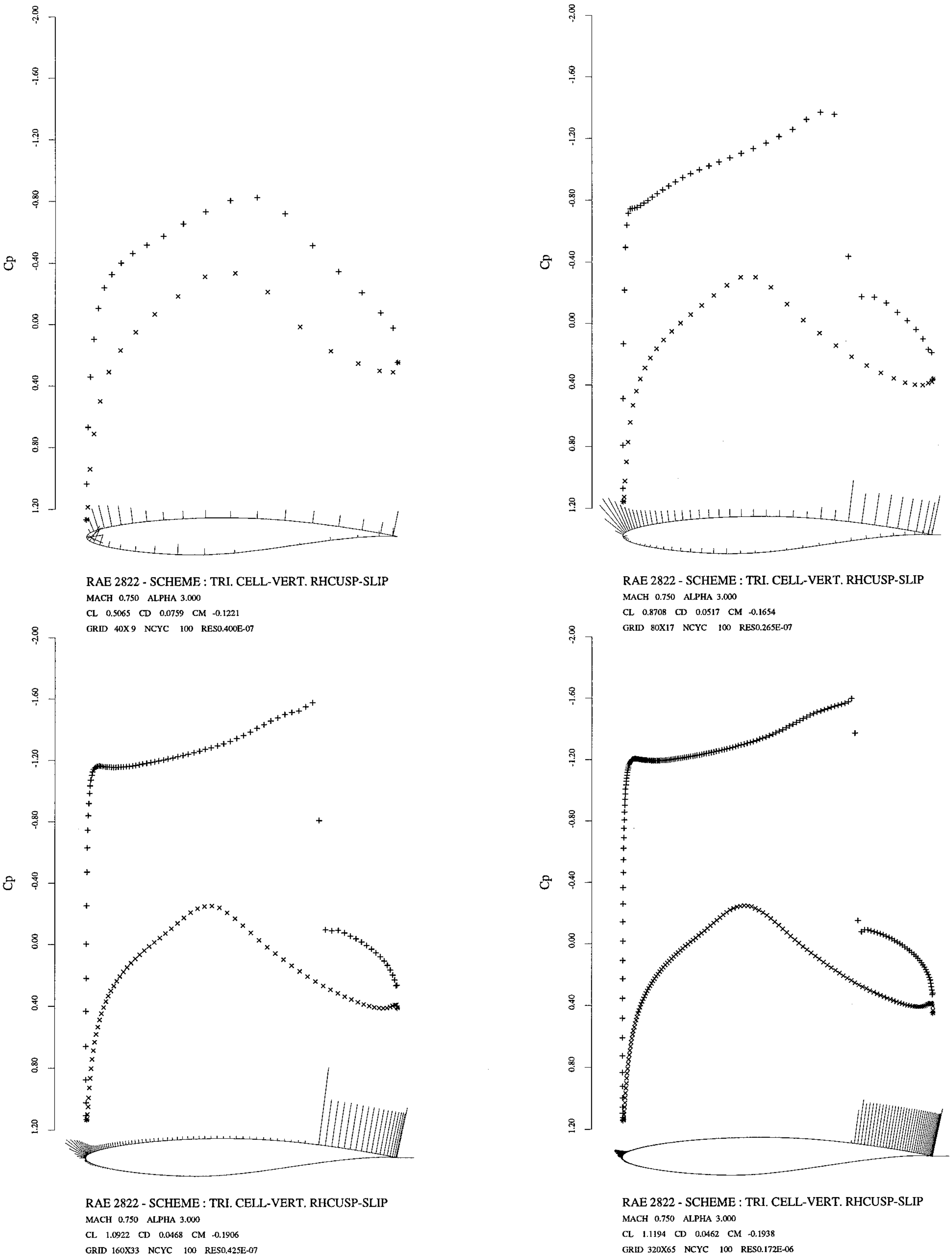
RAE 2822 - SCHEME : TRI. CELL-VERT. HSCAL-JST
MACH 0.750 ALPHA 3.000
CL 0.8330 CD 0.0440 CM -0.1381
GRID 40X 9 NCYC 100 RES0.111E-05

RAE 2822 - SCHEME : TRI. CELL-VERT. HSCAL-JST
MACH 0.750 ALPHA 3.000
CL 1.0250 CD 0.0437 CM -0.1738
GRID 80X17 NCYC 100 RES0.410E-08

RAE 2822 - SCHEME : TRI. CELL-VERT. HSCAL-JST
MACH 0.750 ALPHA 3.000
CL 1.0887 CD 0.0442 CM -0.1861
GRID 160X33 NCYC 100 RES0.111E-09

RAE 2822 - SCHEME : TRI. CELL-VERT. HSCAL-JST
MACH 0.750 ALPHA 3.000
CL 1.1057 CD 0.0447 CM -0.1896
GRID 320X65 NCYC 100 RES0.954E-08

Figure 7: RAE-2822 Solution Series, Cell-Vertex HSCAL-JST Scheme.



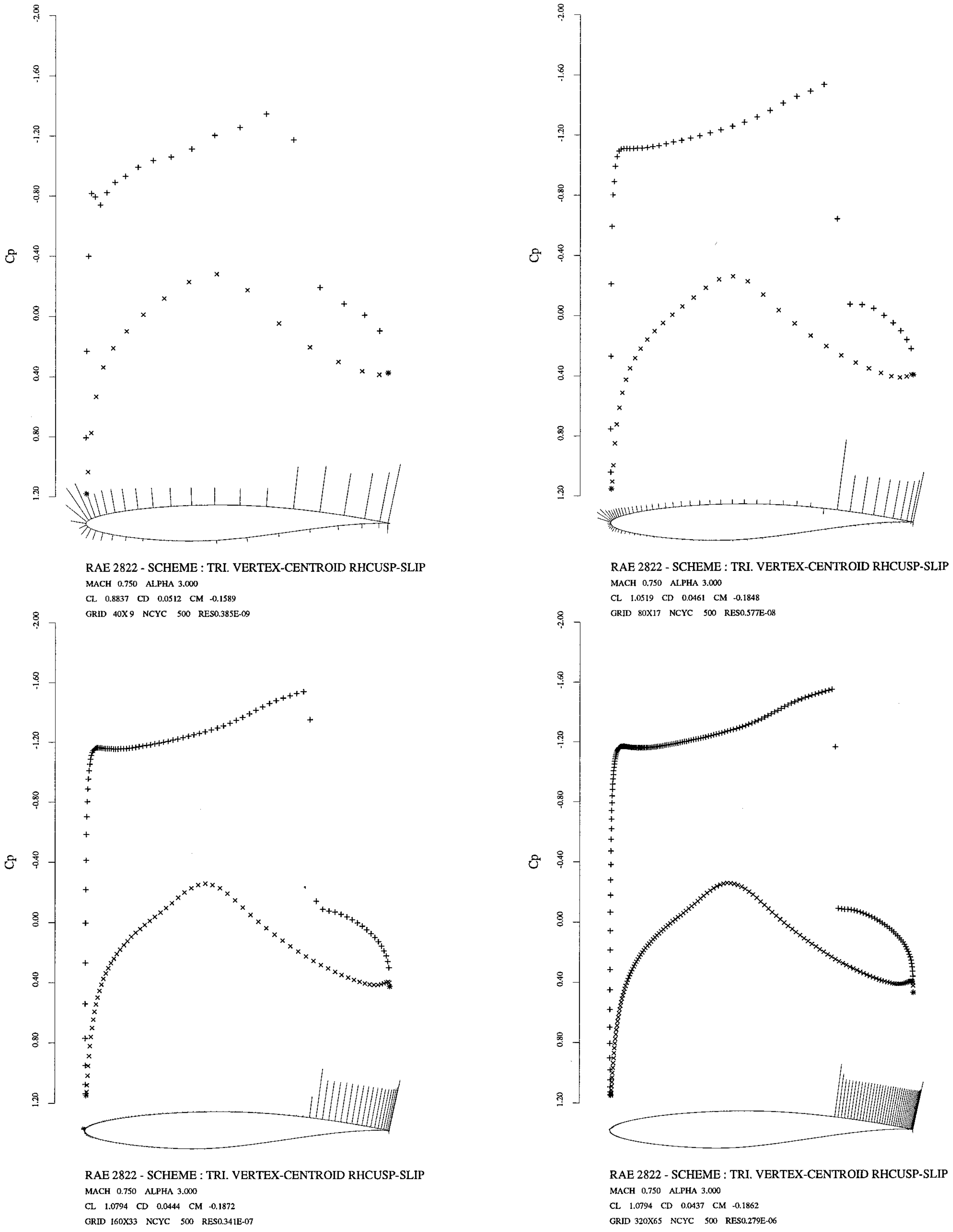
RAE 2822 - SCHEME : TRI. CELL-VERT. RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 0.5065 CD 0.0759 CM -0.1221
GRID 40X9 NCYC 100 RES0.400E-07

RAE 2822 - SCHEME : TRI. CELL-VERT. RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 0.8708 CD 0.0517 CM -0.1654
GRID 80X17 NCYC 100 RES0.265E-07

RAE 2822 - SCHEME : TRI. CELL-VERT. RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 1.0922 CD 0.0468 CM -0.1906
GRID 160X33 NCYC 100 RES0.425E-07

RAE 2822 - SCHEME : TRI. CELL-VERT. RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 1.1194 CD 0.0462 CM -0.1938
GRID 320X65 NCYC 100 RES0.172E-06

Figure 8: RAE-2822 Solution Series, Cell-Vertex RHCUSP-SLIP Scheme.



RAE 2822 - SCHEME : TRI. VERTEX-CENTROID RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 0.8837 CD 0.0512 CM -0.1589
GRID 40X 9 NCYC 500 RES0.385E-09

RAE 2822 - SCHEME : TRI. VERTEX-CENTROID RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 1.0519 CD 0.0461 CM -0.1848
GRID 80X17 NCYC 500 RES0.577E-08

RAE 2822 - SCHEME : TRI. VERTEX-CENTROID RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 1.0794 CD 0.0444 CM -0.1872
GRID 160X33 NCYC 500 RES0.341E-07

RAE 2822 - SCHEME : TRI. VERTEX-CENTROID RHCUSP-SLIP
MACH 0.750 ALPHA 3.000
CL 1.0794 CD 0.0437 CM -0.1862
GRID 320X65 NCYC 500 RES0.279E-06

Figure 9: RAE-2822 Solution Series, Vertex-Centroid RHCUSP-SLIP Scheme.