and $\mathbf{G}$ the viscous flux vector.

$$\mathbf{G} = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ u\tau_{xx} + v\tau_{xy} \\ -q_x \end{pmatrix} \hat{\mathbf{i}} + \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{yx} + v\tau_{yy} \\ -q_y \end{pmatrix} \hat{\mathbf{j}}$$

$$\begin{aligned} \nabla q &= -\kappa \nabla T \\ \boldsymbol{\tau} &= \lambda(u_x + v_y)I + \mu \begin{bmatrix} 2u_x & u_y + v_x \\ v_x + u_y & 2v_y \end{bmatrix} \\ p &= (\gamma - 1)\left(E - \frac{1}{2}\rho(u^2 + v^2)\right) = \rho RT \end{aligned}$$

In these equations λ and μ are diffusion coefficients, κ the coefficient of thermal conductivity, γ the ratio of specific heats, and R the ideal gas law constant.

At solid walls with no slip boundary conditions the flux reduces to the following form for steady flow:

$$\mathbf{F} \cdot \mathbf{n} = \begin{pmatrix} 0 \\ n_x p \\ n_y p \\ 0 \end{pmatrix}, \quad \mathbf{G} \cdot \mathbf{n} = \begin{pmatrix} 0 \\ \mu \nabla u \cdot \mathbf{n} \\ \mu \nabla v \cdot \mathbf{n} \\ -q \cdot \mathbf{n} \end{pmatrix}.$$

The last entry in the viscous flux vanishes for adiabatic flow. The conditions can be enforced weakly or alternatively the second and third equations can be replaced by the strong no slip wall condition $u = v = 0$.

6.3d High Reynolds Number Flow

Simulating the effects of turbulence on unstructured meshes via the Reynolds-averaged Navier-Stokes equations and turbulence modeling is a relatively unexplored topic. In early work by Rostand [Rost88], an algebraic turbulence model was incorporated into an unstructured mesh flow solver. This basic methodology was later refined by Mavriplis [Mav88] for the flow about multi-element airfoil configurations. Both of these implementations utilize locally structured meshes to produce one-dimensional-like boundary-layer profiles from which algebraic models can determine an eddy viscosity coefficient for use in the Reynolds-averaged Navier-Stokes equations. The use of local structured meshes limits the general applicability of the method.

The next level of turbulence modeling involves the solution of one or more auxiliary differential equations. Ideally these differential equations would only require initial data and boundary conditions in the same fashion as the Reynolds-averaged mean equations. The use of turbulence models based on differential equations greatly increases the class of geometries that can be treated "automatically." Unfortunately this does not make the issue of turbulence modeling a "solved" problem since most turbulence models do not perform well across the broad range of flow regimes usually generated by complex geometries. Also keep in mind that most turbulence models for wall-bounded flow require knowledge of

distance to the wall for use in "damping functions" which simulate the damping effect of solid walls on turbulence. The distance required in these models is measured in "wall units" which means that physical distance from the wall y is scaled by the local wall shear, density, and viscosity.

$$y^+ = \sqrt{\frac{\tau_{wall}}{\rho_{wall}}}\frac{y}{\nu} \tag{6.3.2}$$

Scaling by wall quantities is yet another complication but does not create serious implementation difficulties for unstructured meshes as we will demonstrate shortly.

In a recent report with Baldwin [BalB90] we proposed a single equation turbulence transport model. In this report, the model was tested on various subsonic and transonic flow problems: flat plates, airfoils, wakes, etc. The model consists of a single advection-diffusion equation with source term for a field variable which is the product of turbulence Reynolds number and kinematic viscosity, $\nu\widetilde{R}_T$. This variable is proportional to the eddy viscosity except very near a solid wall. The model equation is of the form:

$$\begin{aligned}\frac{D(\nu\widetilde{R}_T)}{Dt} =& (c_{\epsilon_2}f_2(y^+) - c_{\epsilon_1})\sqrt{\nu\widetilde{R}_T P} \\ &+(\nu + \frac{\nu_t}{\sigma_R})\nabla^2(\nu\widetilde{R}_T) - \frac{1}{\sigma_\epsilon}(\nabla\nu_t)\cdot\nabla(\nu\widetilde{R}_T).\end{aligned} \tag{6.3.3}$$

In this equation P is the production of turbulent kinetic energy and is related to the mean flow velocity rate-of-strain and the kinematic eddy viscosity ν_t. Equation (6.3.3) depends on distance to solid walls in two ways. First, the damping function f_2 appearing in equation (6.3.3) depends directly on distance to the wall (in wall units). Secondly, ν_t depends on $\nu\widetilde{R}_t$ and damping functions which require distance to the wall.

$$\nu_t = c_\mu D_1(y^+)D_2(y^+)\nu\widetilde{R}_t$$

It is important to realize that the damping functions f_2, D_1, and D_2 deviate from unity only when very near a solid wall. For a typical turbulent boundary-layer (see fig. 6.3.4) accurate distance to the wall is only required for mesh points which fall below the logarithmic region of the boundary-layer.

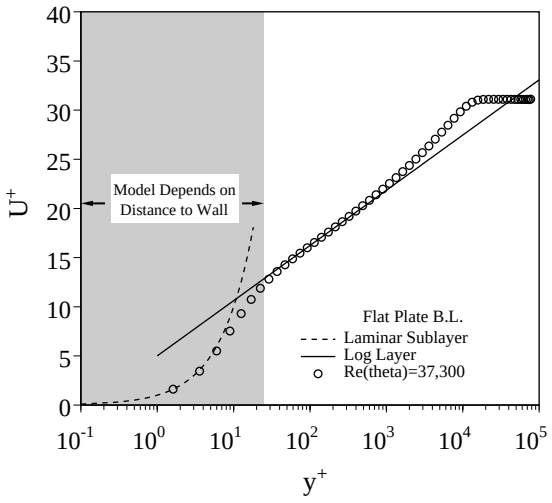

Figure 6.3.4 Typical flat plate boundary-layer showing dependence of turbulence model on distance to wall.

The relative insensitivity of distance to the wall means that accurate estimation of distance to the wall is only required for a small number of points that are extremely close to a boundary surface. The remaining points can be assigned any approximate value of physical distance since the damping functions are essentially at their asymptotic values. A general procedure for calculation of distance to the wall in wall units is to precompute and store, for each vertex of the mesh, the minimum distance from the vertex to the closest solid wall (see fig. 6.3.8 below). This strategy can only fail if two bodies are in such close proximity that the near wall damping functions never reach their asymptotic values. Realistically speaking, the validity of most turbulence models would be in serious question in this situation anyway. In general, the minimum distance from vertex to boundary edge does not occur at the end points of a boundary edge but rather interior to a boundary edge. For each vertex, information concerning which boundary edge achieves the minimum distance and the weight factor for linear interpolation along the boundary edge must be stored. Data can then be interpolated to the point of minimum distance whenever needed. In the course of solving (6.3.3), distance to a solid wall in wall units is calculated by retrieving physical distance to the wall and the local wall quantities needed for (6.3.2) as interpolated along the closest boundary edge.

In the paper with Baldwin [BalB90] we construct a numerical scheme and show precise conditions for which a maximum principle is guaranteed. This aspect of the work was critical in producing a robust numerical model for complex fluid flows. In the following paragraphs we discuss another turbulence model which shares this feature.

One noticeable weakness of the Baldwin-Barth model is the poor accuracy in computing free shear-layer flows. This motivated Spalart and Allaras [SpalA92] to devised a variant of our model. Using their notation the model is written

$$\begin{aligned}\frac{D\widetilde{\nu}}{Dt} =& \frac{1}{\sigma}\left[\nabla\cdot((\nu+\widetilde{\nu})\nabla\widetilde{\nu}) + c_{b2}(\nabla\widetilde{\nu})^2\right] \\ &- \left[c_{w1}f_w - \frac{c_{b1}}{\kappa^2}f_{t2}\right]\left[\frac{\widetilde{\nu}}{d}\right]^2 + c_{b1}\widetilde{S}\widetilde{\nu}\end{aligned} \tag{6.3.4}$$

where d is the *physical* distance to solid walls, c_b1, c_b2, c_w1, and σ are constants, and $\widetilde{S}$ is related to the strain tensor. The complete model also includes terms for simulating the transition of turbulence. Numerical schemes can be constructed for solving this equation which *naturally* exhibit a maximum principle. When combined with the improved modeling of shear-layer flows, the model is very attractive.

6.3e Examples of Navier-Stokes Flow

The numerical calculations presented in this section represent a successful application of the ideas discussed in previous sections.

Blasius Boundary-Layer

Self-similar laminar boundary-layer is simulated on a 10x20 subdivided quadrilateral mesh, see fig. 6.3.5. An analytical Blasius profile is specified at inflow and computations at low Mach number (M=.08) were performed using the various reconstruction schemes.

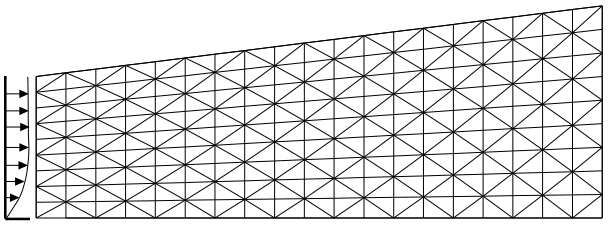

Figure 6.3.5 Coarse Flat Plate Grid.

Viscous terms have been discretized in finite-volume form using the continuous triangle interpolants (linear or quadratic) and integration quadrature rules consistent with these interpolants. Figure 6.3.6 graphs the velocity profile which has been vertically sampled at x/L = .88. Data is plotted at locations corresponding to the intersection of this vertical line and the edges of the mesh. This results in a somewhat irregular spacing of data points.

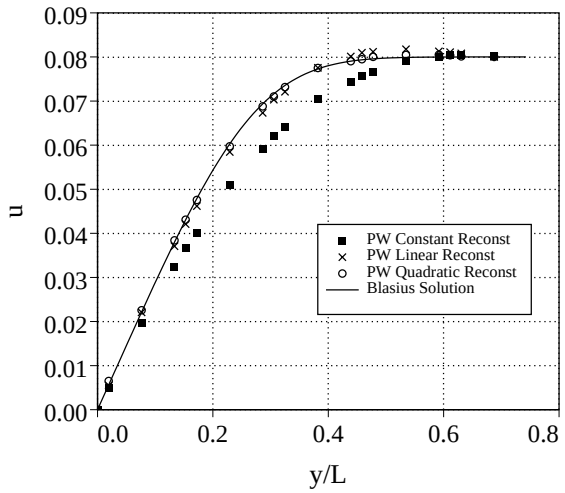

Figure 6.3.6 Velocity profiles at x/L=.88.

The piecewise constant reconstruction performs rather poorly owing to the excessive diffusion in the Euler scheme. The linear reconstruction scheme (without limiting) produces a reasonably accurate solution with a slight overshoot in the outer portion of the boundary-layer. The quadratic scheme performs very well with a slight error in the region of maximum curvature. Both higher order methods appear to have acceptable accuracy for this problem.

RAE 2822 Airfoil

The geometry consists of a single RAE 2822 airfoil. Navier-Stokes flow is computed about this geometry assuming turbulent flow with the following free-stream conditions: $M_\infty = .725, \alpha = 2.31°, Re = 6.5$ million. Wind tunnel experiments for the RAE 2822 geometry have been performed by Cook, McDonald, and Firmin [CookMF79]. The RAE 2822 airfoil mesh shown in fig. 6.3.7 contains approximately 14000 vertices and 41000 edges.

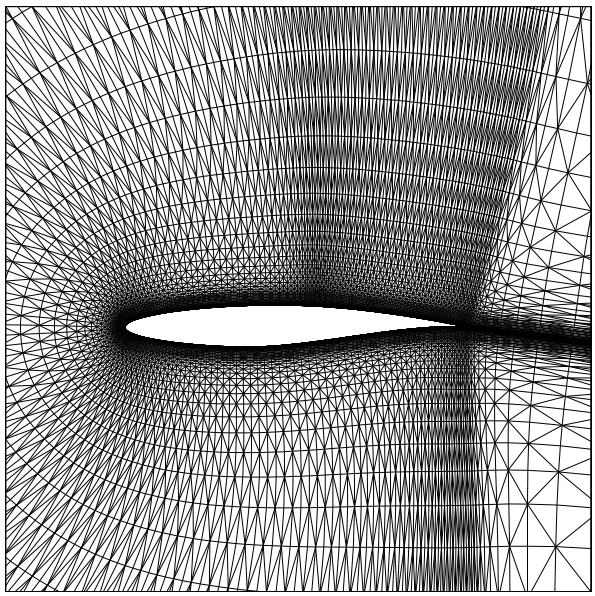

Figure 6.3.7 Mesh near RAE 2822 airfoil.

Navier-Stokes calculations on the RAE 2822 airfoil were performed using the implicit scheme described in the next section. The Baldwin-Barth turbulence model was used to simulate the effects of turbulence. For purposes of calculating distance to solid walls in the turbulence model, a distance function is first computed for all mesh vertices. Contours of this function are shown in fig. 6.3.8.

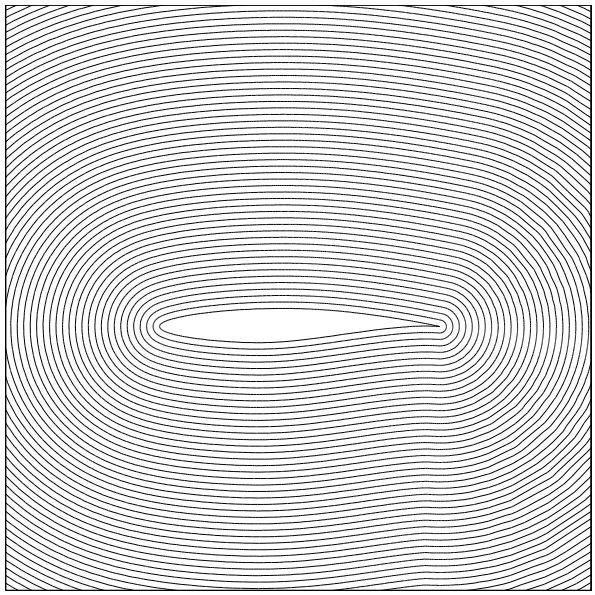

Figure 6.3.8 Contours of distance function for turbulence model.

Figures 6.3.9-10 plot Mach number contours and pressure coefficient distributions for the RAE 2822 airfoil. The pressure coefficient distribution compares favorably with the experiment.

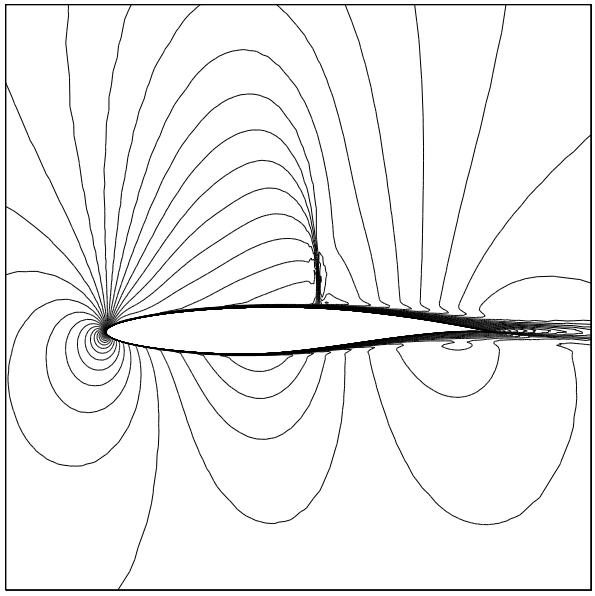

Figure 6.3.9. Closeup of Mach number contours near airfoil.

Leading edge trip strips were used on the experimental model but not simulated in the computations. This may explain the minor differences in the leading edge pressure distribution.

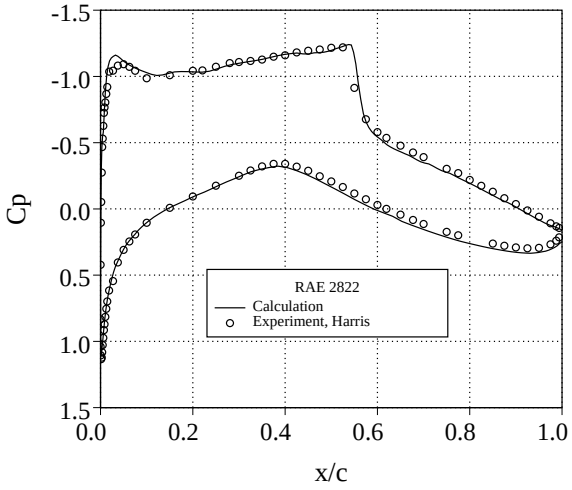

Figure 6.3.10 Pressure coefficient distribution on airfoil.

Multi-Element Airfoil Flow

Turbulent high Reynolds number flow ($M = .2$, $\alpha = 8.2°$, Re = 9 million) is computed about a 3-element airfoil configuration. Experimental data is available at these conditions, see Valarezo *et al.* [ValDM91].

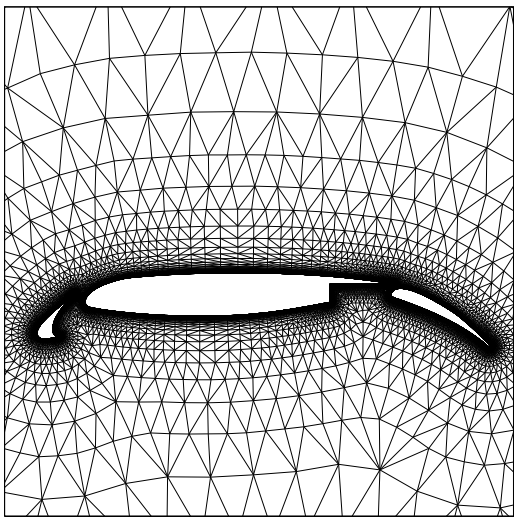

Figure 6.3.11 Grid about multi-element airfoil configuration.

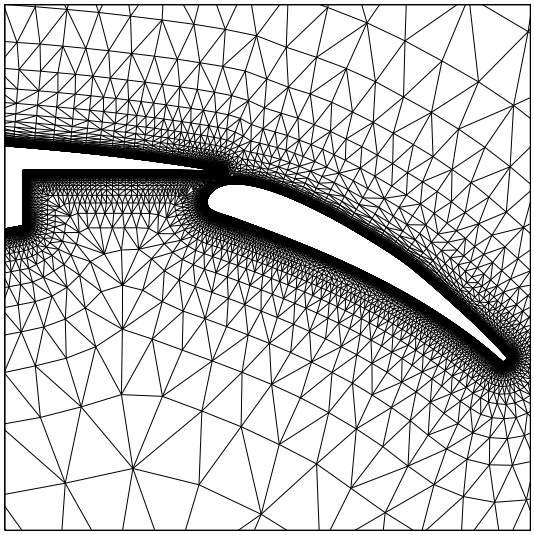

Figure 6.3.12 Closeup of grid near flap region.

Computations were performed using the linear least-squares reconstruction without limiting. The Baldwin-Barth one-equation turbulence model was used to simulate the effects of turbulence. This model has not been implemented in the other schemes at this date so the computations have been restricted to the linear reconstruction scheme only. The mesh containing approximately 28,000 vertices is shown in figs. 6.3.11-12.

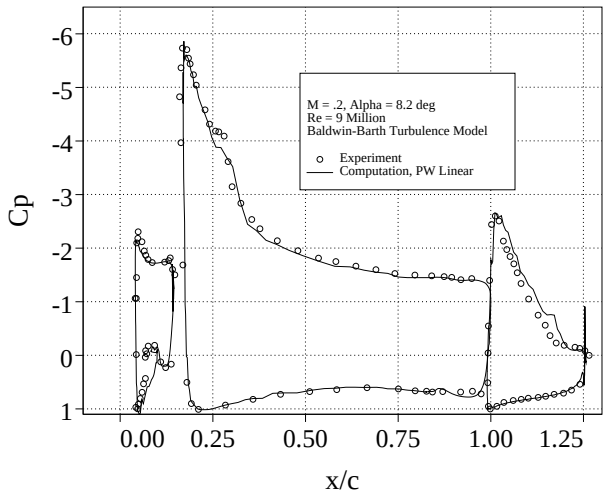

Figure 6.3.13 Cp distribution on element surfaces.

The pressure coefficient distribution is graphed in fig. 6.3.13 for the computation and experiment. The agreement with experiment is reasonably good although the mesh resolution is inadequate in the wake regions of the flow. This can be seen in the Mach contour plots shown in figs. 6.3.14-15. Proper mesh resolution of the wake flow is an excellent opportunity for mesh adaptation although we have not developed an adaptation procedure

for this purpose.

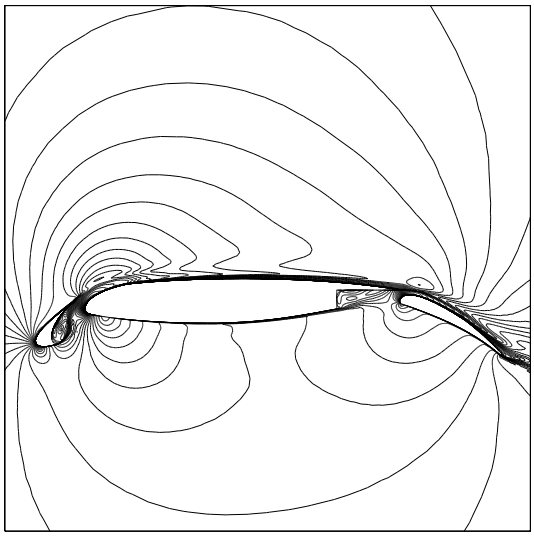

Figure 6.3.14 Mach contours about multi-element airfoil.

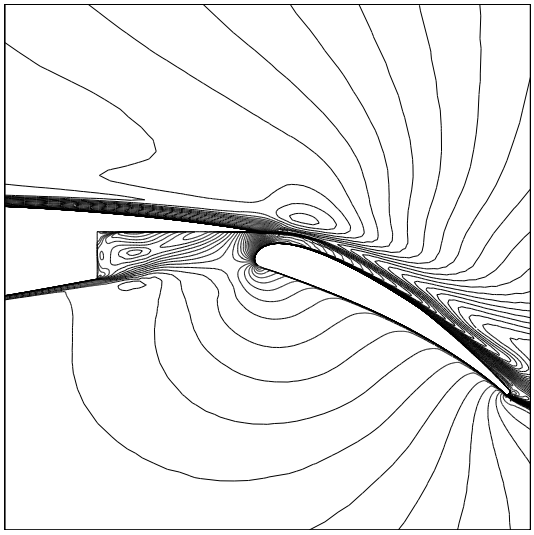

Figure 6.3.15 Closeup of Mach contours in trailing edge region.

7.0 Convergence Acceleration For Steady-State Calculations

In this section we examine two techniques for accelerating the convergence of numerical schemes to steady-state. Section 7.1 will discuss a randomized multigrid procedure. The central idea is to exploit the randomized theory for Delaunay and related triangulations. Then in Section 7.2 we consider implicit solution strategies based on a preconditioned minimum residual method.

7.1 Randomized Multigrid

Recall from Section 3.3c the theory of randomized incremental triangulation. If the *order* in which sites are introduced for insertion into the triangulation is randomized then it is possible to store a history of the *entire* triangulation process as a data tree (see fig. 7.1.0) with the following characteristics:

(1) Expected size $O(m \log n + n)$

(2) Average traversal depth $O(\log n)$

where m is the size of the original triangulation T_0 and n is the number sites to be inserted into T_0. Usually m is a small number so that the size is approximately $O(n)$. Thus we construct a data structure which effectively encodes a composition of structural changes transforming the mesh from its coarsest representation, T_0, to its finest representation, T_n. Let, S_i denote the i-th structure change, the encoding process is represented by the following composition:

$$T_n = (S_l(S_{l-1}(S_{l-2}...(S_3(S_2(S_1T_0)))...))). \tag{7.1.0}$$

This structure permits us to stop at any point in the composition and extract a valid intermediate triangulation. This is the central idea that we exploit in a multigrid setting. In addition, this composition is used for global restriction and prologation given local restriction and prologation operators on convex sets of $d+1$ points in R^d, see discussion below.

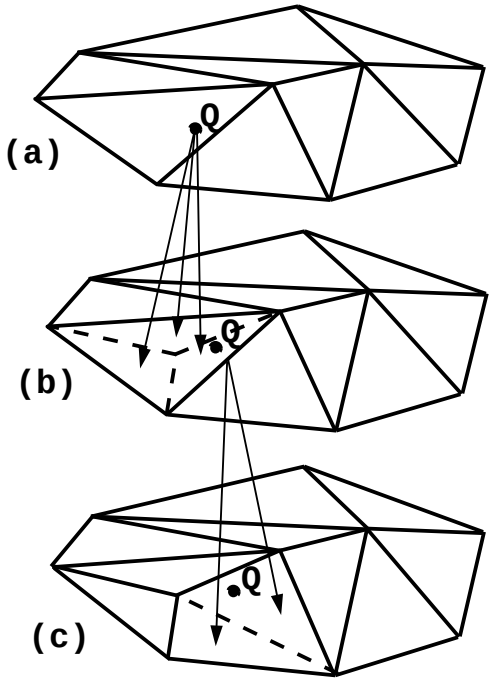

Figure 7.1.0 Data tree for site location. (a) Original triangulation containing Q. (b) Triangle is split into 3 from site insertion. (c) Triangle is split into 2 from edge swap.

Note that the intermediate meshes produced by this procedure are node nested subsets. This is a departure from the multigrid algorithms of Jameson and Mavriplis [JamM85],

[MavJ86]. The actual multigrid algorithm follows a standard procedure for nonlinear problems, see Jespersen [Jesp83] or Spekreijse [Spek87].

Stage I: Nested iteration

Let

$$R_k(u_k) = r_k \tag{7.1.1}$$

be a discretization of the conservation law with source term r on $T_k, k = 0, ...l$. Denote the solution of (7.1.1) by $u^*, k = 0, ..., l$. The nested iteration scheme starts with some initial guess, u_0^* and procedes recursively. Given an approximation u_k^*, an approximation of u_{k+1}^* is obtained as follows. The approximation of u_k^* is improved by a single NMG iteration and this improved solution is projected onto a finer mesh, T_{k+1}. These steps are repeated until u_l^* is obtained.

Stage II: nonlinear multi-grid (NMG)

One iteration of NMG on a general grid T_k is defined recursively by the following steps:

(0) Start with an approximation u_k of u_k^*

(1) Improve u_k by application of p relaxation iterations to $R_k(u_k) = r_k$.

(2) Compute the defect $d_k := r_k - R_k(u_k)$.

(3) Find an approximation of u_{k-1} of u_{k-1}^* of the coarser grid T_{k-1} via restriction operator, I_k^{k-1}.

(4) Compute $r_{k-1} := R_{k-1}(u_{k-1}) + I_k^{k-1} d_k$.

(5) Approximate the solution of $R_{k-1}(u_{k-1}) = r_{k-1}$ by NMG on T_{k-1}. The result is $\widetilde{u}_{k-1}$.

(6) Correct the current approximation by $u_k := u_k + I_{k-1}^k(\widetilde{u}_{k-1} - u_{k-1})$.

(7) Improve u_k by application of q relaxation iterations to $R_k(u_k) = r_k$.

To demonstrate the method, consider solving the Burger's equation problem introduced in Section 5.3 using the upwind scheme (4.2.18) based on piecewise constant reconstruction. A 4-stage Runge-Kutta scheme is used for pre and post relaxation iterations with $p = q = 2$. For ease of exposition, assume that the control volumes are the actual triangles. In this case the restriction and prolongation operators are particularly simple. We also introduce a new local operator called the "recombination" operator. This operator accounts for the edge swapping transformation during the triangulation process.

The restriction and prolongation operators can now be represented as the composition of local operators, T_i, for example:

$$R_k = I_{k-1}^k R_{k-1} = (T_j(T_{j-1}(...(T_{i+1}(T_i R_{k-1})))))$$

where i and j are the locations of the levels $k-1$ and k and $j > i$. The three local transfer operators are depicted in the fig. 7.1.1.

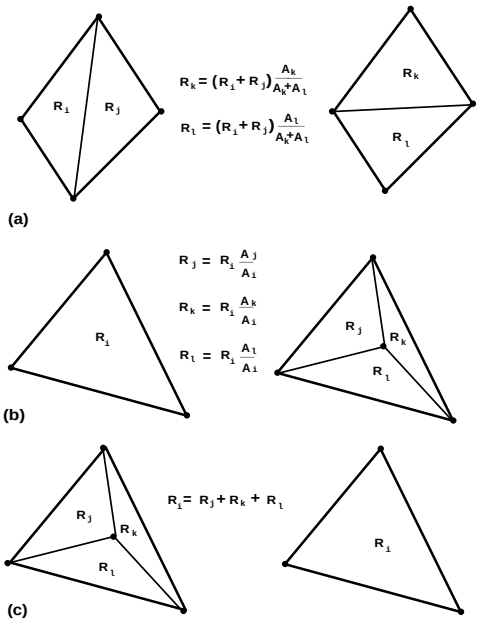

Figure 7.1.1 Local Transfer Operators (a) Recombination (b) Prolongation (c) Restriction

The algorithm is tested using the Burger's equation problem of Section 5.3 on meshes containing 3200 and 12800 cells. Intermediate meshes are extracted from (7.1.0) so that the number of solution unknowns is successively reduced by a factor of 4. Figure 7.1.2 plots the coarsest mesh and fig. 7.1.3 plots the mesh containing 3200 cells.

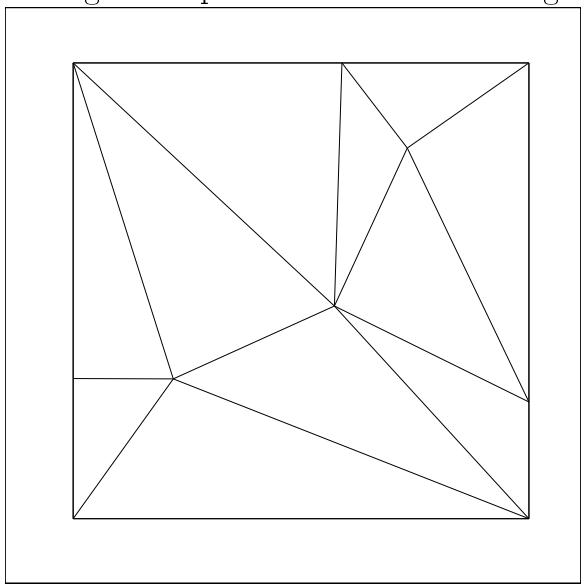

Figure 7.1.2 Coarsest mesh for multigrid calculation.

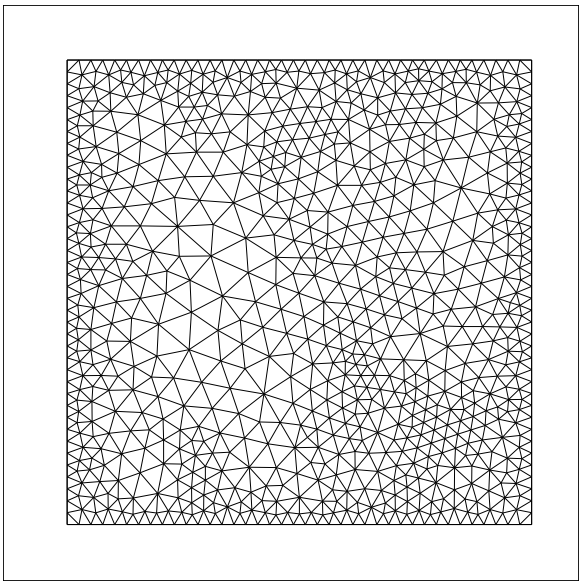

Figure 7.1.3 Finest mesh (3200 cells) for multigrid calculation.

Figure 7.1.4 graphs the convergence history for the multigrid scheme using 5 level V-cycles on the mesh containing 3200 cells and 6 level V-cycles on the mesh containing 12800 cells.

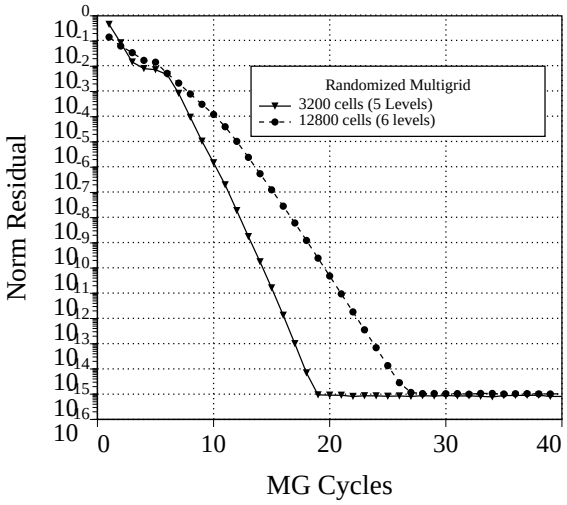

Figure 7.1.4 Convergence history for randomized multigrid scheme.

Ideally we would like to observe grid size independent rates of convergence. The results do not indicate true grid independent convergence but come fairly close. The results may be improved somewhat by alternate mesh level choices. One alternative might be a nonstationary randomized algorithm for level selection.

7.2 Implicit Solution Algorithms

In this section we consider implicit solution strategies for the upwind schemes described

in Section 4-6. Defining the solution vector

$$\mathbf{u} = [\overline{u}_1, \overline{u}_2, \overline{u}_3, ..., \overline{u}_N]^T$$

the basic scheme is written as

$$D\mathbf{u}_t = \mathbf{r}(\mathbf{u}) \tag{7.2.0}$$

where D is a positive diagonal matrix. Next perform a backward Euler time integration, equation (7.2.0) is rewritten as

$$D(\mathbf{u}^{n+1} - \mathbf{u}^n) = \Delta t\ \mathbf{r}(\mathbf{u}^{n+1}) \tag{7.2.1}$$

where n denotes the iteration (time step) level. Linearizing the right-hand-side of (7.2.1) in time produces the following form:

$$D(\mathbf{u}^{n+1} - \mathbf{u}^n) = \Delta t \left[\mathbf{r}(\mathbf{u}^n) + \frac{d\mathbf{r}^n}{d\mathbf{u}}(\mathbf{u}^{n+1} - \mathbf{u}^n)\right] \tag{7.2.2}$$

By rearrangement of terms, we arrive at the delta form of the backward Euler scheme

$$\left[D - \Delta t\ \frac{d\mathbf{r}^n}{d\mathbf{u}}\right](\mathbf{u}^{n+1} - \mathbf{u}^n) = -\Delta t\ \mathbf{r}(\mathbf{u}^n) \tag{7.2.2}$$

Note that for large time steps, the scheme becomes equivalent to Newton's method for finding roots of a nonlinear system of equations. In practice the time step is usually scaled as a exponential function of the norm of the residual

$$\Delta t = f(\|\mathbf{r}(\mathbf{u})\|)$$

so that when $\|\mathbf{r}(\mathbf{u})\|$ approaches zero the time step becomes large. Each iteration of the scheme requires the solution of a linear system of equations of the form $Mx = b$. The two most difficult tasks in this procedure are:

(1) Calculation of the Jacobian matrix elements

(2) Solution of the sparse linear system, $Mx = b$

Before discussing these steps, it is worthwhile to assess the storage requirements for this strategy. Assume that the solution unknowns are associated with vertices of the mesh. All of the schemes discussed in previous sections require knowledge of distance-one neighbors in the graph (mesh). Furthermore, the higher order accurate schemes require more distant neighbors in building the scheme. If we consider only distance-one neighbors then the number of nonzero entries in each row of the matrix is related to the number of edges incident to the vertex associated with that row. Or equivalently, each edge $e(v_i, v_j)$ will guarantee nonzero entries in the i-th column and j-th row and similarly the j-th column and i-th row. In addition nonzero entries will be placed on the diagonal of the matrix. From this counting argument we see that the number of nonzero entries in the matrix is exactly twice the number of edges plus the number of vertices, $2E + N$. This

argument holds in three dimensions as well. Although we do not prove it here, the storage requirements become prohibitively large when more distant neighbors are included into the matrix pattern.

The strategy taken here is to store only the matrix associated with distance-one neighbors. We will require two copies of this matrix which are used in two different ways: (1) in the preconditioning of the linear system, (2) matrix-vector products in the iterative solver.

7.2a Calculation of Matrix Elements

The main burden in this calculation is the linearization of the numerical flux vector with respect to the two solution states.

For example, given the flux vector

$$\begin{aligned}\mathbf{h}(\mathbf{u}^R, \mathbf{u}^L; \mathbf{n}) =& \frac{1}{2}\left(\mathbf{f}(\mathbf{u}^R, \mathbf{n}) + \mathbf{f}(\mathbf{u}^L, \mathbf{n})\right) \\ & -\frac{1}{2}|A(\mathbf{u}^R, \mathbf{u}^L; \mathbf{n})|\left(\mathbf{u}^R - \mathbf{u}^L\right)\end{aligned} \tag{7.2.3}$$

we require the Jacobian terms $\frac{d\mathbf{h}}{d\mathbf{u}^R}$ and $\frac{d\mathbf{h}}{d\mathbf{u}^L}$. Exact analytical expressions for these terms are available [Bar87]. As mentioned above, in practice we only compute the matrix elements associated with distance-one neighbors in the mesh. For the edge $e(v_i, v_j)$ this requires the Jacobian terms $\frac{d\mathbf{h}_{ij}}{d\overline{u}_i}$ and $\frac{d\mathbf{h}_{ij}}{d\overline{u}_j}$.

Using the edge data structure mentioned in Section 1 the actual assembly of the sparse matrix from components becomes very simple. Again the idea is that we can determine the flux Jacobian terms edge-wise. The edge $e(v_i, v_j)$ then contributes entries in the i-th row and j-th column as well as the j-th row and i-th column.

7.2b Solution of the Linear System

The task is to solve the large sparse linear system

$$Mx - b = 0$$

or in left preconditioned form

$$P(Mx - b) = 0.$$

The approach take here is to use the GMRES algorithm of Saad and Schultz [SaadS86] with incomplete L-U factorization preconditioning. Consider for simplicity the nonpreconditioned form of the linear system. Let x_0 be an approximate solution of

$$Mx - b = 0$$

where M is assumed invertible. The solution is advanced from x_0 to $x_k = x_0 + y_k$. Let $v_1 = Mx_0 - b$, $r_k = Mx_k - b$. GMRES(k) finds the best possible folution for the y_k over the Krylov subspace $[v_1, Mv_1, M^2 v_1, ..., M^{k-1} v_1]$ by solving the minimization problem

$$\|r_k\| = \min_y \|v_1 + My\|.$$

GMRES forms an orthogonal basis $v_1, v_2, ..., v_k$ spanning the Krylov subspace. As k increases, the storage increases linearly and the computation quadratically. Thus Saad and Schultz devised GMRES(k, m) where the GMRES algorithm is restarted every m steps. A complete discussion of the GMRES solver and Krylov subspace methods will be deferred to the lectures of Prof. Saad.

If we consider the backward Euler time stepping scheme with large Δt, we see that the matrix-vector products are actually Fréchet directional derivatives of the residual vector in the direction v, i.e.

$$Mv = \frac{\partial R(u)}{\partial u} v.$$

This has prompted researchers [Shak88] to consider "matrix-free" versions of GMRES in which the right-hand-side term is calculated numerically

$$\frac{\partial R(u)}{\partial u} v \approx \frac{R(u + \epsilon v) - R(u)}{\epsilon}$$

We do not adopt this strategy because we actually desire a matrix (presumably better than a diagonal approximation) for the preconditioning step . Let M_0 denote the Jacobian matrix associated with the distance-one neighbors in the mesh. We compute the matrix-vector product terms in GMRES in the following fashion:

$$Mv = M_0 v + \frac{\Delta R(u + \epsilon v) - \Delta R(u)}{\epsilon}$$

where $\Delta R(u)$ represents the remaining terms in the discretization. We find that this choice makes the scheme less dependent on the choice of ϵ. This decomposition strategy is a departure from previous work by Venkatakrishnan who examined iterative ILU-GMRES algorithms [VenkM93] together with fluid flow algorithms but used the distance-one neighbor matrix for matrix-vector products in GMRES.

The effectiveness of the GMRES hinges heavily on the preconditioning of the linear system. In the present applications, we precondition using incomplete L-U factorization with zero additional fill, ILU(0). The basic idea in ILU(0) is to perform a lower-upper factorization of the matrix ignoring any new fill that might occur. Again we defer a discussion of ILU to the lectures of Prof. Saad.

7.3c Numerical Examples

Subsonic Airfoil Flow

As a first example inviscid subsonic flow $M_\infty = .63, \alpha = 2.0°$ is computed about the NACA 0012 airfoil using the first order spatially accurate scheme. In this case the Jacobian linearization M_0 is exact and Newton's method is obtained for large Δt. The grid, solution, and convergence history are shown in fig. 7.3.0a-c.

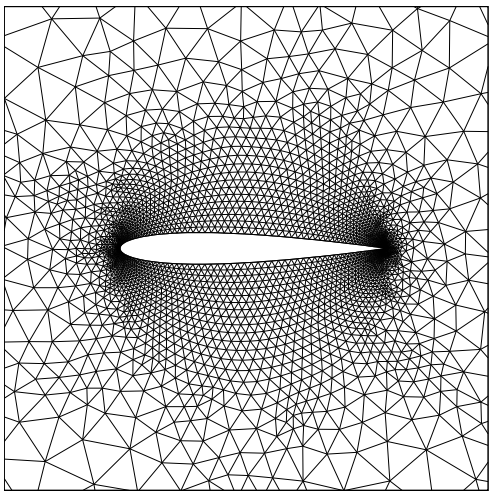

Figure 7.3.0a Grid about NACA 0012 airfoil.

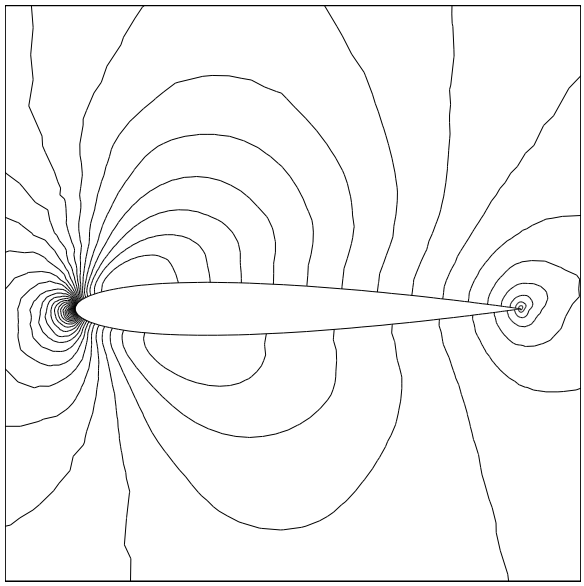

Figure 7.3.0b Mach contours about NACA 0012 airfoil using first order accurate scheme.

In the convergence history plot we also show the convergence behavior of one GMRES cycle with Δt corresponding to a CFL number of about 10000. For comparison we show the behavior with ILU preconditioning and simple block-diagonal preconditioning. The improvement with ILU preconditioning is dramatic at this large time step.

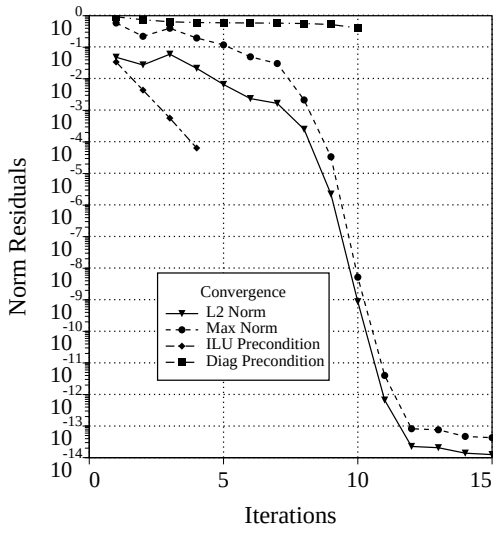

Figure 7.3.0c Convergence history showing global convergence of the scheme and convergence of 1-cyle of GMRES(k,m) $k = 12, m = 5$ using block diagonal and ILU preconditioning.

As a second example inviscid subsonic flow is computed about the NACA 0012 airfoil using the second order spatially accurate scheme with linear reconstruction and no limiting. In this case Fréchet numerical derivatives are also included in the GMRES computation. The numerical solution and convergence history are shown in fig. 7.3.1a-b.

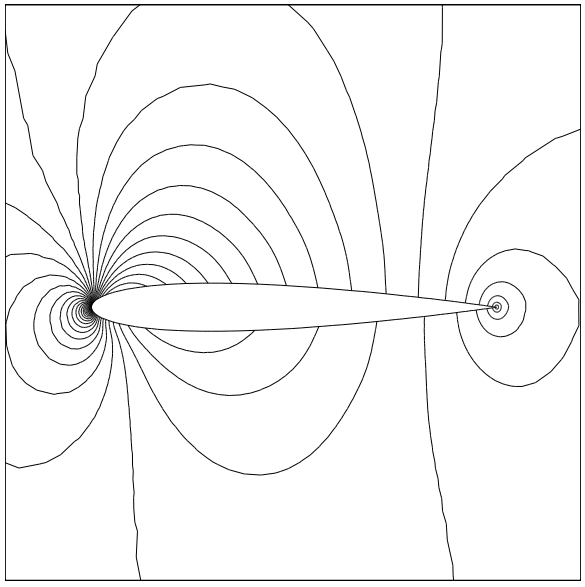

Figure 7.3.1a Mach contours about NACA 0012 airfoil using second order accurate scheme.

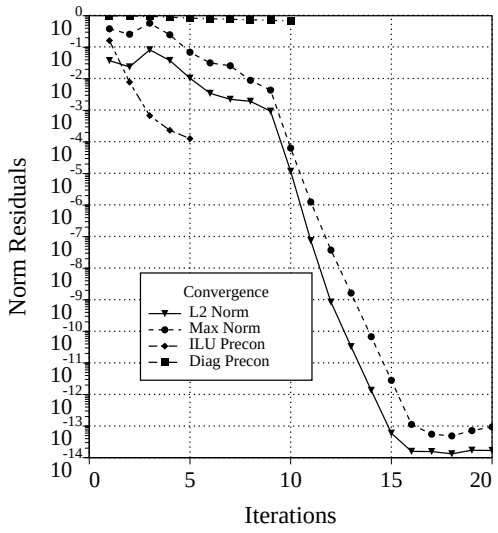

Figure 7.3.1b Convergence history showing global convergence of the scheme and convergence of 1-cyle of GMRES(k,m) $k = 12, m = 5$ using ILU preconditioning.

The residual history shows a rapid convergence to steady-state but never achieves Newton convergence. This could be due to the choice of $\epsilon = 1.e-6$. We also show the convergence behavior of one GMRES cycle with Δt corresponding to a CFL number of about 10000. For comparison we show the behavior with ILU preconditioning and simple block-diagonal preconditioning. The ILU preconditioning is degraded somewhat from the previous example but is still very effective. The block diagonal preconditioning is largely ineffective at this large time step.

Transonic Airfoil Flow

As a third example inviscid transonic flow $M_\infty = .8, \alpha = 1.25°$ is computed about the NACA 0012 airfoil using the first and second order spatially accurate schemes. To avoid convergence problems owing to the nondifferentiability of the limiter function in the second order accurate scheme, the smooth limiter function proposed by Venkatakrishnan [Venkat93] is used (although the limiter allows small overshoots in the solution). The numerical solution and convergence histories are shown in fig. 7.3.2a-b.

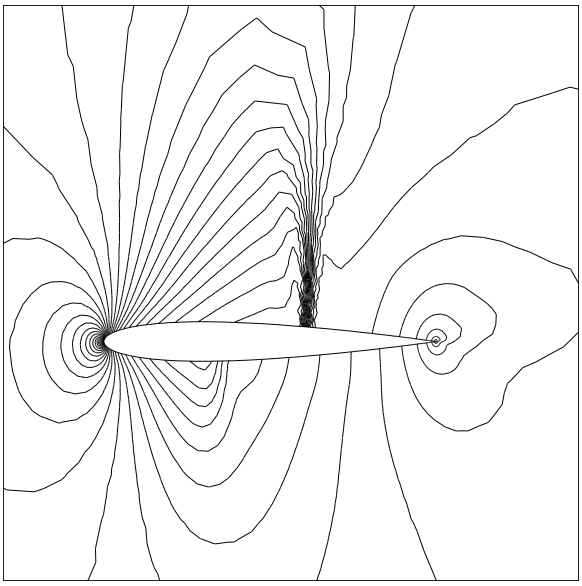

Figure 7.3.2a Mach contours about NACA 0012 airfoil using second order accurate scheme.

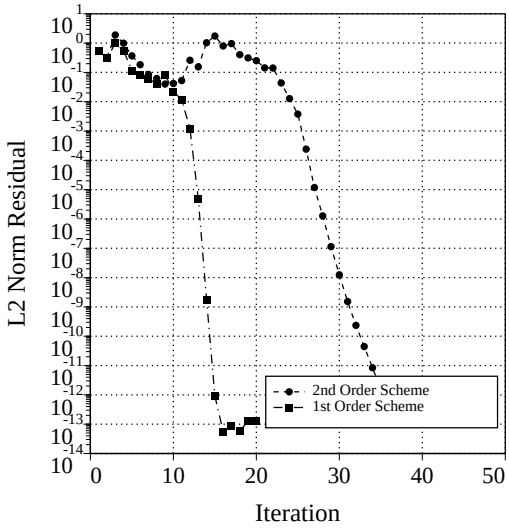

Figure 7.3.2b Convergence history showing global convergence of the scheme and convergence of 1-cyle of GMRES(k,m) $k = 12, m = 5$ using ILU preconditioning.

Although both the first and second order accurate schemes converge rapidly near steady-state, the second order scheme requires almost twice as many iterations before the solution is "sufficiently close" so that Newton's method may yield rapid convergence. This is largely due to the strong shock which develops early in the transient and must propagate upstream to its final position.

Viscous Flow

As a final example, viscous turbulent flow ($M_\infty = .3, \alpha = 2.31°$, Re = 6 million) is computed about the RAE 2822 airfoil using the first and second order spatially accurate

schemes with linear reconstruction. The mesh has been previously shown in fig. 6.3.7. The numerical solution, and convergence history are shown in fig. 7.3.3b-c. In these calculations, the turbulence model equations are solved independently of the mean flow equations. This creates a serious departure from Newton's method.

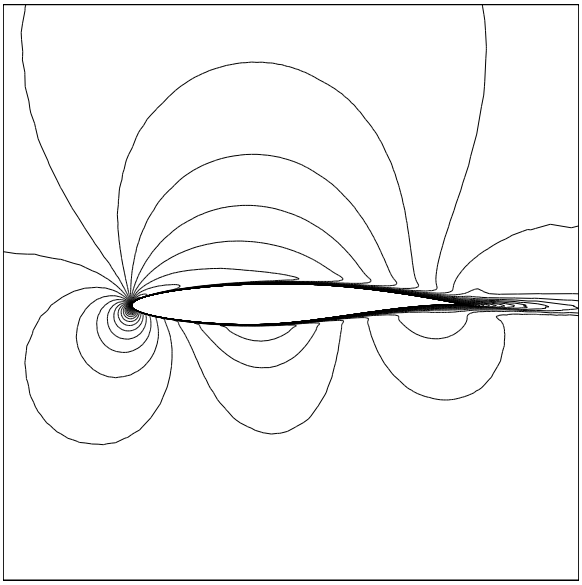

Figure 7.3.3a Mach contours about RAE 2822 airfoil using second order accurate scheme.

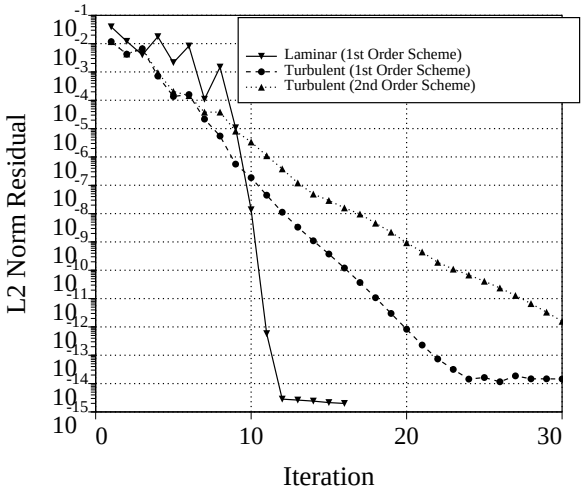

Figure 7.3.3b Convergence history of the scheme for laminar flow (Re= .5 Million), turbulent flow (Re=6 Million) using first and second order schemes.

The effect is quite noticeable for the first order scheme. In this case the number of iterations required has doubled. The effect is very damaging for the second order scheme. In this case the only reasonable solution would be to fully couple the turbulence equations to the mean flow equations. This is an avenue for future work.

References

[AftGT94] "On the Accuracy, Stability and Monotonicity of Various Reconstruction Algorithms for Unstructured Meshes," AIAA paper 94-0415, 1994.

[And92] Anderson, W.K.,"A Grid Generation and Flow Solution Method for the Euler Equations on Unstructured grids," NASA Langley Research Center, USA, unpublished manuscript, 1992.

[ArmD89] Construction of TVD-like Artificial Viscosities on 2-Dimensional Arbitrary FEM Grids," INRIA Report 1111, October 1989.

[BabuA76] Babuška, I., and Aziz, A. K., "On the Angle Condition in the Finite Element Method", SIAM J. Numer. Anal., Vol. 13, No. 2, 1976.

[Bak89] Baker, T. J. ,"Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation", Engineering with Computers, Vol. 5, 1989, pp. 161-175.

[BaldB90] Baldwin, B.S., and Barth, T.J.,"A One-Equation Turbulence Transport Model for High Reynolds Number Wall-Bounded Flows," NASA TM-102847, August 1990.

[Barn93] Barnard, S.T., and Simon, H.D., "A Fast Multilevel Implementation pf recursive Spectral Bisection," In Proc. 6h SIAM Conf. Parallel Proc. for Sci. Comp., 1993, pp. 711-718.

[BarL87] Barth, T.J., and Lomax, H.L., "Algorithm Development," NASA CP 2454, 1987, pp. 191-200.

[Bar87] Barth, T.J.,"Analysis of Implicit Local Linearization Techniques for Upwind and TVD Schemes," AIAA Paper 87-0595, 1987.

[BarJ89] Barth, T. J., and Jespersen, D. C., "The Design and Application of Upwind Scehemes on Unstructured Meshes", AIAA-89-0366, Jan. 9-12, 1989.

[BarF90] Barth, T. J., and Frederickson, P. O., "Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction", AIAA-90-0013, Jan. 8-11, 1990.

[Bar91a] Barth, T.J.,"Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes", AIAA paper 91-0721, January, 1991.

[Bar91b] Barth, T. J.,"A Three-Dimensional Upwind Euler Solver of Unstructured Meshes," AIAA Paper 91-1548, Honolulu, Hawaii, 1991.

[Bar93] Barth, T.J.,"Recent Developments in High Order K-Exact Reconstuction on Unstructured Meshes", AIAA paper 93-0668, January, 1993.

[BernE90] "Mesh Generation and Optimal Triangulation," Tech Report CSL-92-1, Xerox PARC, March 1992.

[Bow81] Bowyer, A., "Computing Dirichlet Tessellations", The Computer Journal, Vol. 24, No. 2, 1981, pp. 162—166.

[Bri89] Brisson, E.,"Representing Geometric Structures in d Dimensions: Topology and Order", In Proceedings of the 5th ACM Symposium on Computational Geometry., 1989, pp. 218-227.

[Chio85] Chiocchia, G., " Exact Solutions to Transonic and Supersonic Flows", AGARD Advisory Report AR-211,1985.

[Chew89] Chew, L.P., " Guaranteed-Quality Triangular Meshes," Department of Computer Science Tech Report TR 89-983, Cornell University, 1989.

[Chew93] Chew, L.P., "Guaranteed-Quality Mesh Generation for Curved Surfaces," ACM Press, Proceedings of the 9th Annual Symposium on Computational Geometry, San Diego, CA, 1993.

[ChroE91] Chrobak, M. and Eppstein, D.,"Planar Orientations with Low Out-Degree and Compaction of Adjacency Matrices", Theo. Comp. Sci., Vol. 86, No. 2, 1991, pp.243-266.

[Ciarlet73] Ciarlet, P.G., Raviart, P.-A., "Maximum Principle and Uniform Convergence for the Finite Element Method", Comp. Meth. in Appl. Mech. and Eng., Vol. 2., 1973, pp. 17-31.

[ColW84] Colella, P., Woodward, P., "The Piece- wise Parabolic Method for Gas-Dynamical Simulations", J. Comp. Phys., Vol. 54, 1984.

[CookMF79] Cook, P.H., McDonald M.A., Firmin, M.C.P., "AEROFOIL RAE 2822 Pressure Distributions, and Boundary Layer and Wake Measurements," AGARD Advisory Report No. 139, 1979.

[CutM69] Cuthill, E., and McKee, J.,"Reducing the Band Width of Sparse Symmetric Matrices", Proc. ACM Nat. Conf., 1969, pp. 157-172.

[Del34] Delaunay, B., "Sur la Sphére Vide", Izvestia Akademii Nauk SSSR, Vol. 7, No. 6, Oct. 1934, pp. 793-800.

[DesD88] Desideri, J. A., and Dervieux, A., "Compressible Flow Solvers on Unstructured Grids," VKI Lecture Series 1988-05, March 7-11, 1988, pp. 1–115.

[DobL89] Dobkin, D.P., and Laszlo, M.J.,"Primitives for the Manipulation of Three- dimensional Subdivisions", Algorithmica, Vol. 4, 1989, pp. 3-32.

[Edel90] Edelsbrunner, H., Tan, T.S, and Waupotitsch, R.,"An $O(n^2 \log n)$ Time Algorithm for the MinMax Angle Triangulation," Proceedings of the 6th ACM Symposium on Computational Geometry, 1990, pp. 44-52.

[Fort93] Fortune, S.J., and Van Wyk, C.J., "Efficient Exact Arithmetic for Computational Geometry," ACM Press, Proceedings of the 9th Annual Symposium on Computational Geometry, San Diego, CA, 1993.

[GanB92] Gandhi, A. S., and Barth T. J.,"3-D Unstructured Grid Generation and Refinement Using 'Edge-Swapping' ", NASA TM-103966, 1993.

[Geo71] George, J.A,"Computer Implementation of the Finite Element Method", Techical Report No. STAN-CS-71-208, Computer Science Dept., Stanford University, 1971.

[Gil79] Gilbert, P.N., "New Results on Planar Triangulations", Tech. Rep. ACT-15, Coord. Sci. Lab., University of Illinois at Urbana, July 1979.

[God59] Godunov, S. K., "A Finite Difference Method for the Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics", Mat. Sb., Vol. 47, 1959.

[GoodLV85] Goodman, J.D., Le Veque, R.J., "On the Accuracy of Stable Schemes for 2D Conservation Laws," Math. Comp., Vol. 45.

[GreS77] Green, P. J. and Sibson, R., "Computing the Dirichlet Tesselation in the Plane", The Computer Journal, Vol. 21, No. 2, 1977, pp. 168—173.

[GuiS85] Guibas, L.J., and Stolfi, J.,"Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams", ACM Trans. Graph., Vol. 4, 1985, pp. 74-123.

[GuiKS92] Guibas, L.J., Knuth, D.E., and Sharir, M., "Randomizing Incremental Construction of Delaunay and Voronoi Diagrams", Algorithmica, Vol. 7, 1992, pp. 381-413.

[Guill1993] Guillard, H., "Node-Nested Multi-grid Method with Delaunay Coarsening," INRIA report 1898, 1993.

[HamB91] Hammond, S., and Barth, T.J., "An Efficient Massively Parallel Euler Solver for Unstructured Grids", AIAA paper 91-0441, Reno, 1991.

[Hart83] Harten, A., "High Resolution Schemes for Hyperbolic Conservation Laws," J. Comp. Phys., Vol. 49, pp. 357-393.

[HartEOC] Harten, A., Engquist, B., Osher, S., Chak- ravarthy, "Uniformly High Order Accurate Essentially Non - Oscillatory Schemes III, ICASE report 86-22, 1986.

[HartHL76] Harten, A., Hyman, J.M., and Lax, P.D., "On Finite-Difference Approximations and Entropy Conditions for Shocks," Comm. Pure Appl. Math., Vol. 29, 1976, pp. 297-322.

[HartO85] Harten, A. , Osher, S., "Uniformly High-Order Accurate Non-oscillatory Schemes, I.," MRC Technical Summary Report 2823, 1985.

[HolS88] Holmes, G. and Snyder, D., "The Generation of Unstructured Triangular Meshes using Delaunay Triangulation," in *Numerical Grid Generation in CFD*, pp. 643-652, Pineridge Press, 1988.

[Jam86] Jameson, A., and Lax, P., "Conditions for the Construction of Multi-Point Total Variation Diminishing Difference Schemes," ICASE Report 178076, March 1986.

[Jam93] Jameson, A., "Artificial Diffusion, Upwind Biasing, Limiters and their Effect on Accuracy and Multigrid Convergence in Transonic and Hypersonic Flows," AIAA paper 93-3359, 1993.

[JamM85] Jameson, A. and Mavriplis, D., "Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh", AIAA paper 85-0435, January 1985.

[Jesp83] Jespersen, D.C., "Design and Implementation of a Multigrid Code for the Euler Equations," Appl. Math. and Computat., Vol. 13, 1983, pp. 357–374.

[Joe89] Joe, B.,"Three-Dimensional Delaunay Triangulations From Local Transformations", SIAM J. Sci. Stat. Comput., Vol. 10, 1989, pp. 718-741.

[Joe91] Joe, B.,"Construction of Three-Dimensional Delaunay Triangulations From Local Transformations", CAGD, Vol. 8, 1991, pp. 123-142.

[Klee80] Klee, V., "On the Complexity of d-dimensional Voronoi diagrams", Archiv der Mathematik, Vol. 34, 1980.

[Knuth92] Knuth, D.E., **Axioms and Hulls**, Lecture Notes in Computer Science, Springer-Verlag, Heidelberg, 1992.

[Knuth93] Knuth, D.E., **Stanford Graph Base: A Platform for Combinatorial Computing**, Addison- Wesley Publishing, New York, 1993.

[Law77] Lawson, C. L., "Software for C^1 Surface Interpolation", Mathematical Software III, (Ed., John R. Rice), Academic Press, New York, 1977.

[Law86] Lawson, C. L., "Properties of n-dimensional Triangulations" CAGD, Vol. 3, April 1986, pp. 231–246.

[Mav88] Mavriplis, D., "Adaptive Mesh Generation for Viscous Flows Using Delaunay Triangulation," ICASE Report No. 88-47,1988.

[MavJ87] Mavriplis, D. and Jameson, A., "Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes", AIAA paper 87-0353, January 1987.

[Mich94] Michell, C., "Improved Reconstruction Schemes for the Navier-Stokes Equations on Structured Meshes," AIAA paper 94-0642, January 1994.

[Nira90a] Nira, D., Levin, D., Rippa, S.,"Data Dependent Triangulations for Piecewise Linear Interpolation", J. Numer. Anal., Vol. 10, No. 1, 1990, pp. 137-154.

[Nira90b] Nira, D., Levin, D., Rippa, S.,"Algorithms for the Construction of Data Dependent Triangulations", **Algorithms for Approximation, II**, Chapman, and Hall, London, 1990, pp. 192-198.

[PothSL90] Pothen, A., Simon, H.D., and Liou, K.-P., "Partitioning Sparse Matrices with Eigenvectors of Graphs," SIAM J. Matrix Anal. Appl., Vol. 11, No. 3, July 1990, pp. 430-452.

[Raj91] Rajan, V. T., "Optimality of the Delaunay Triangulation in $\mathbf{R}^d$", Proceedings of the 7th ACM Symposium on Computational Geometry, 1991, pp. 357-372.

[Rippa90] Rippa, S., "Minimal Roughness Property of the Delaunay Triangulation", CAGD, Vol. 7, No. 6., 1990, pp-489–497.

[Roe81] Roe, P.L.,"Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes", J. Comput. Phys., Vol 43, 1981.

[Ros68] Rosen, R., "Matrix Band Width Minimization", Proc. ACM Nat. Conf., 1968, pp. 585-595.

[Rost88] Rostand, P., "Algebraic Turbulence Models for the Computation of Two- dimensional High Speed Flows Using Unstructured Grids", ICASE Report 88-63, 1988.

[Rupp92] Ruppert, J., "A New and Simple Algorithm for Quality 2-Dimensional Mesh Generation," Report UCB/CSD 92/694, University of California, Berkeley, 1992.

[SaadS86] Saad, Y., Schultz, M.H. "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," SIAM J. Sci. Stat. Comp., Vol. 7, No. 3, 1986, pp. 856-869.

[Schmitt79] Schmitt, V., and Charpin, F., "Pressure Distributions on the ONERA M6-Wing at Transonic Mach Numbers," in "Experimental Data Base for Computer Program Assessment," AGARD AR-138, 1979.

[Shak88] Shakib, F., "Finite Element Analysis of the Compressible Euler and Navier-Stokes Equations", PhD Thesis, Department of Mechanical Engineering, Stanford University, 1988.

[Simon91] Simon, H.D., "Partitioning of Unstructured Problems for Parallel Processing," NASA Ames R.C., Tech. Report RNR-91-008, 1991.

[SpalA92] Spalart, P., and Allmaras, S., "A One-Equation Turbulence Model for Aerodynamic Flows," AIAA Paper 92-0439, 1992.

[Spek87] Spekreijse, S.P., "Multigrid Solution of Monotone Second-Order Discretizations of Hyperbolic Conservation Laws," Math. Comp. Vol. 49, pp. 135-155.

[Spek88] Spekreijse, S.P., "Multigrid Solution of the Steady Euler Equations," CWI Tract, Centre for Mathematics and Computer Science, Amsterdam, The Netherlands, 1988.

[StruVD89] Struijs, R., Vankeirsblick, and Deconinck, H., "An Adaptive Grid Polygonal Finite Volume Method for the Compressible Flow Equations," AIAA-89-1959-CP, 1989.

[ThomLW85] Thomas, J.L., van Leer, B., and Walters, R.W., "Implicit Flux-Split Schemes for the Euler Equations," AIAA paper 85-1680, 1985.

[VanL79] Van Leer, B., "Towards the Ultimate Conservative Difference Schemes V. A Second Order Sequel to Godunov's Method", J. Comp. Phys., Vol. 32, 1979.

[ValDM91] Valarezo, W.O., Dominik, C.J., and McGhee, R.J.,"Multi-Element Airfoil Optimization for Maximum Lift at High Reynolds Numbers," AIAA Paper 91-3332, September, 1991.

[VankD92] Vankeirsbilck, P., and Deconinck, H., "Higher Order Upwind Finite Volume Schemes with ENO Properties for General Unstructured Meshes," AGARD Report R-787, May 1992.

[VenSB91] Venktakrishnan, V., Simon, H.D., Barth, T.J., "A MIMD Implementation of a Parallel Euler Solver for Unstructured Grids", NASA Ames R.C., Tech. Report RNR-91-024, 1991.

[Ven93] Venktakrishnan, V., "On the Accuracy of Limiters and Convergence to Steady State Solutions," AIAA paper 93-0880, 1993.

[Vor07] Voronoi, G., "Nouvelles Applications des Paramétres Continus á la Théorie des Formes Quadratiques," Premier Mémoire: Sur Quelques Proprieteés de Formes Quadratiques Positives Parfaites, J. Reine Angew. Math., Vol. 133, 1907, pp. 97-178.

[Wat81] Watson, D. F, "Computing the n-dimensional Delaunay Tessellation with Application to Voronoi Polytopes", The Computer Journal, Vol. 24, No. 2, 1981, pp. 167—171.

[WoodC84] Woodward, P., Colella, P., "The Numerical Simulation of Two-Dimensioal Fluid Flow with Strong Shocks" J. Comp. Phys., Vol. 54, 1984.

[XiaN92] Xia, and Nicolaides, "Covolume Techniques for Anisotropic Media," Numer. Math., Vol. 61, 1992, pp. 215-234.