AIAA-93-3021

Numerical Wind Tunnel — Vision or Reality

A. Jameson
Princeton University
Princeton, NJ

AIAA 24th
Fluid Dynamics Conference
July 6–9, 1993 / Orlando, FL

Numerical Wind Tunnel

Vision or Reality

Antony Jameson

Department of Mechanical and Aerospace Engineering

Princeton University

Princeton, New Jersey 08544 USA

1 Abstract

In 1975 Chapman, Mark and Pirtle suggested that computers should begin to supplant wind tunnels. This paper examines the extent to which this vision has been sustained, and what remains to be done to achieve the goal of a numerical wind tunnel. Five principal steps in the development of software for aerodynamic simulation are identified, with emphasis on the issues of mathematical modeling of fluid flow and computational algorithms. After a survey of the current state of the art, future prospects are examined. In light of progress in computer processing speeds and memory, it is concluded that the goal will be realized, though much remains to be done, and to be an effective tool, aerodynamic simulation needs to be embedded in a complete numerical design environment.

2 Introduction

In a landmark paper of 1975 [1], Chapman, Mark and Pirtle anticipated that "computers should begin to supplant wind tunnels in the aerodynamic design and testing process". In effect, computers would ultimately provide a numerical wind tunnel. They listed three main objectives of computational aerodynamics:

1. To provide flow simulations that are either impractical or impossible to obtain in wind tunnels or other ground based experimental test facilities.

2. To lower the time and cost required to obtain aerodynamic flow simulations necessary for the design of new aerospace vehicles.

3. Eventually, to provide more accurate simulations of flight aerodynamics than wind tunnels can.

Chapman, Mark, and Pirtle also noted that the inherent limitations of computational and wind tunnel simulations are complementary. Wind tunnels are limited by the size of the models that can be placed in them, and by the density, temperature and velocity of the flow that they can sustain, with the consequence that flight-Reynolds numbers cannot be realized with complete models. Their accuracy is also limited by wall and support interference, and by aeroelastic distortion. Computers are not limited in any of these ways, but they are limited in speed and memory, which in turn limit the attainable complexity and resolution of the simulations.

Now that computational fluid dynamics (CFD) has come of age, we can survey the extent to which their vision has been sustained, and examine what remains to be done to achieve a more complete realization of the goal of a numerical wind tunnel. Prior to 1965 computational methods were hardly used in aerodynamic analysis, although they were widely used for structural analysis. There had been serious studies of CFD at Los Alamos and a basic theoretical groundwork and fundamental algorithms had been developed, especially at the Courant Institute. The introduction of panel methods by Hess and Smith at Douglas Aircraft in 1962, and their subsequent development [2, 3, 4], marked the introduction of CFD as a practical tool for aerodynamic analysis.

The decade 1970–1980 was a period in which CFD became useful for the analysis of previously intractable transonic flow problems. Garabedian and his associates at the Courant Institute devel-

oped design and analysis methods for supercritical airfoils from 1970 onwards [5], and these were used by Whitcomb to improve his airfoils [6]. Murman and Cole revolutionized the field by their introduction of a scheme using mixed upwind and central differences to solve the transonic small disturbance equation [7, 8]. This was extended to treat general transonic potential flows in the author's rotated difference scheme [9], and fully conservative and finite volume implementations were soon developed [10, 11, 12]. Intensive research led to dramatic reductions in computational costs through the use of fast algorithms, notably the introduction of approximate factorization methods and multigrid schemes [13, 14, 15, 16, 17, 18, 19, 20]. There were also notable developments in algorithms for solving the Euler and Navier-Stokes equations, including the MacCormack and the Beam-Warming schemes [21, 22, 23]. An important theoretical development was the Kreiss-Gustafsson theory for initial boundary value problems [24, 25, 26]. This decade also saw the emergence of practical computer programs for three-dimensional transonic flow, some of which have remained in use to the present time.[1]

The decade 1980–1990 saw the rapid expansion of CFD to more complex applications, which included more complete geometric configurations and higher level models of the flow. A variety of methods were developed for the solution of the Euler and Navier-Stokes equations on arbitrary meshes, including NASA Langley's CFL3D and NASA Ames' ARC3D and INS3D codes. The author's FLO57 code to solve the three-dimensional Euler equations was also widely distributed, and provided the basis for Euler codes used at Boeing, for the Lockheed TEAM code, and for NASA Langley's TLNS3D code to solve the thin layer Navier-Stokes equations. Multigrid acceleration methods were extended to the Euler equations [27, 28, 29]. These methods achieve very rapid convergence to a steady state by advancing the solution in a cycle through a sequence of successively coarser grids, on each of which the correction is determined by the error on the next finer grid. This essentially allows global equilibrium to be achieved simultaneously with local equilibrium. Intensive research efforts were concentrated

on the development of non-oscillatory shock capturing schemes, with the emergence of key new concepts such as flux splitting [30], and total variation diminishing (TVD) and essentially non-oscillatory (ENO) schemes [31, 32].

An obstacle to the treatment of configurations with complex geometry has been the problem of mesh generation. Several general techniques were developed, including algebraic transformations and methods based on the solution of elliptic and hyperbolic equations. In the last few years methods using unstructured meshes have also begun to gain more general acceptance. The Dassault-INRIA group led the way in developing a finite element method for transonic potential flow. They obtained a solution for a complete Falcon 50 as early as 1982 [33]. Euler methods for unstructured meshes have been the subject of intensive development by several groups since 1985 [34, 35, 36, 37, 38], and Navier-Stokes methods on unstructured meshes have also been demonstrated [39, 40].

In the next section, this paper discusses steps needed for the implementation of a numerical wind tunnel which could meet industrial needs. A critical issue examined in Section 4 is the choice of mathematical models: what level of complexity is needed to provide sufficient accuracy for aerodynamic design, and what is the impact on cost and turn-around. Section 5 addresses issues in the formulation of numerical algorithms which provide the fundamental building blocks for a numerical wind tunnel. Issues of software design, verification, and maintenance are briefly reviewed in Section 6, which also addresses software evaluation and validation. Section 7 addresses the current state of the art, and presents representative calculations from a number of contributors who have kindly supplied their results. The final section examines some of the remaining issues, including optimization and design, and offers an outlook for the future.

3 Disciplines and steps needed for the implementation of a numerical wind tunnel

CFD, as it has now developed, lies at the intersection of a number of distinct disciplines, as illustrated

---

[1]FLO22, for example, developed by Caughey and the author was modified by Henne for inverse wing design, and is still used at Douglas Aircraft.

in the Figure 1. These include Mathematics, Fluid Dynamics, Computer Science, and (for useful aeronautical applications) Aerospace Engineering. The development of a numerical wind tunnel wil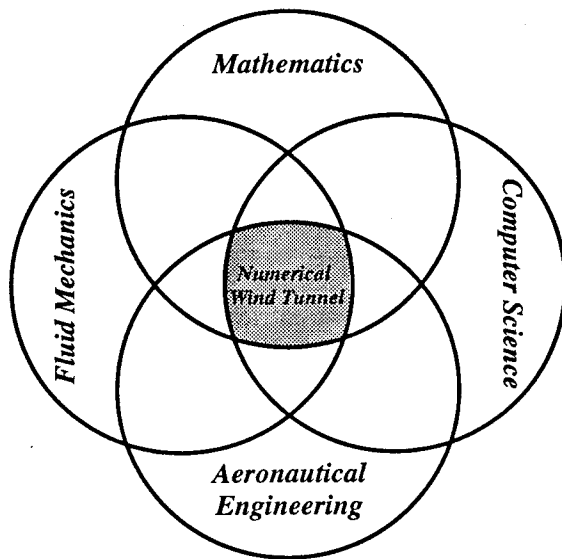l require all of these disciplines. The theoretical underpinnings rest in mathematics and numerical analysis. These include such areas as theory of partial differential equations (including mixed elliptic, parabolic and hyperbolic systems of equations), formulation of proper boundary conditions, and error estimates and convergence proofs for discrete approximation schemes.

Issues in computer science are becoming increasingly important in CFD. With the advent, first of vector, and then of massively parallel computers, algorithms can no longer be developed without considering the architecture on which they are to be implemented. In algorithms using unstructured meshes, for example, loops which exchange data along edges between neighboring mesh points introduce vector dependencies. Graph coloring algorithms can be used to separate the edges into groups such that no two edges of the same color meet at the same mesh point. The dependency is then eliminated by using a separate loop for each group. Another important contribution of Computer Science to CFD is the development of computer graphics. These techniques are vital for effective postprocessing and interpretation of the results.

The complexity of fluid flow is such that useful CFD methods can hardly be developed without a good understanding of fluid mechanics, and uncritical use of inappropriate CFD methods by users lacking an understanding of fluid mechanics can lead to unfortunate results. Numerical dissipation, for example, can lead to spurious flow separation at a high angle of attack in what is supposed to be an inviscid solution. Finally, it is difficult to achieve effective use of CFD methods in the design process without a broad knowledge of aerospace engineering. This background is needed both to determine which calculations are likely to be useful and cost effective, and to know how to interpret the results to improve a proposed design.

The core requirement for the successful implementation of a numerical wind tunnel is the development of the basic software for the computational simulation of complex fluid flows. A level of accuracy sufficient to assure confidence in the aerodynamic design must be achieved with acceptable computational costs and rapid turn-around. Five principal steps can be identified in the development of software for aerodynamic simulation. These are:

Figure 1: Disciplines of Computational Aerodynamics.

1. Choice of a mathematical model appropriate to the requirements.

2. Mathematical analysis of the model to ensure that the problem is properly formulated (for example, definition of far field boundary conditions, conditions for uniqueness).

3. Formulation of a numerical approximation scheme.

4. Implementation as a computer program.

5. Validation.

All of these steps need to be carefully carried out to produce the reliable, robust, and accurate software that is needed.

Software which meets the basic requirements for the computational simulation of aerodynamic flows is the first step towards a numerical wind tunnel, but not by itself sufficient. A satisfactory numerical wind tunnel will need the simulation software to be embedded in a more comprehensive environment to provide user-friendly interfaces and efficient data

3

management. The transfer of large volumes of data representing the geometries and computational results can become a major bottleneck. It is essential to develop procedures allowing the easy transfer of geometric data from computer aided design (CAD) systems. A numerical wind tunnel should also contain automatic measurement systems capable of determining the geometry of real objects. These interfaces should be general enough to allow the substitution of alternative CAD systems, and also of alternative aerodynamic simulation modules. Efficient methods for handling the output data are equally important. These must provide both for visualization of the results, and for their quantitative evaluation. Visualization is important because it can provide the designer with the insights needed to make an improvement, while quantitative evaluation is needed both for verification of the performance, and to allow optimization of the design.

Figure 2: Equations of Fluid Dynamics for Mathematical Models of Varying Complexity (Supplied by Luis Miranda, Lockheed Corporation).

4 Mathematical Models of Fluid Flow and Computational Costs

The choice of a model depends on the complexity of the flow, the level of accuracy required, and the computational cost. The Reynolds number of aerodynamic flows associated with the flight envelopes of full scale aircraft are very large (of the order of 10 million and up). Correspondingly the flows that need to be predicted are generally turbulent. The size of the smallest eddies in a turbulent flow may be estimated by dimensional analysis to be of the order of $Re^{\frac{3}{4}}$, where Re is the Reynolds number based on a representative global length scale. In a three-dimensional simulation, allowing for the need to resolve time scales which correspond to the smallest length scales, the computational complexity of a full simulation of a turbulent flow, down to the smallest scales in space and time, can therefore be estimated as proportional to the cube of the Reynolds number. Consequently a direct simulation of the full Navier-Stokes equations is not feasible, forcing the use of mathematical models with some level of simplification. Figure 2 indicates a hierarchy of models at different levels of simplification which have proved useful in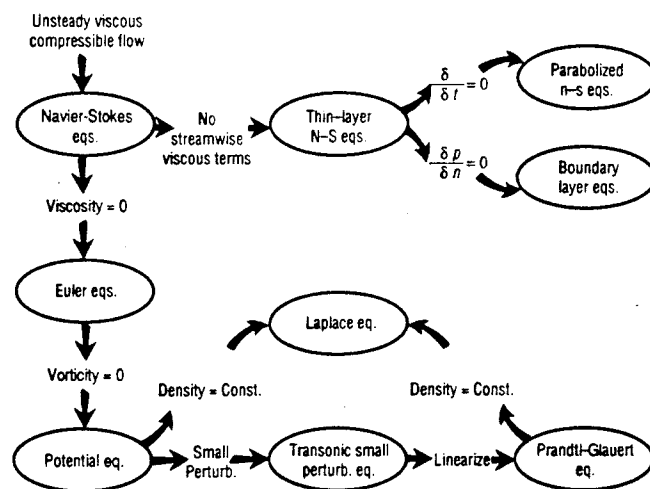 practice. Efficient flight is generally achieved by the use of smooth and streamlined shapes which avoid flow separation and minimize viscous effects, with the consequence that useful predictions can be made using inviscid models. Inviscid calculations with boundary layer corrections can provide quite accurate predictions of lift and drag when the flow remains attached, but iteration between the inviscid outer solution and the inner boundary layer solution becomes increasingly difficult with the onset of separation. Procedures for solving the full viscous equations are likely to be needed for the simulation of arbitrary complex separated flows, which may occur at high angles of attack or with bluff bodies. In order to treat flows at high Reynolds numbers, one is generally forced to estimate turbulent effects by Reynolds averaging of the fluctuating components. This requires the introduction of a turbulence model. As the available computing power increases one may also aspire to large eddy simulation (LES) in which the larger scale eddies are directly calculated, while the influence of turbulence at scales smaller than the mesh interval is represented by a subgrid scale model.

Computational costs vary drastically with the choice of mathematical model. Panel methods can be effectively implemented with higher-end personal computers (with an Intel 80486 microprocessor, for

4

example). Studies of the dependency of the result on mesh refinement, performed by this author and others, have demonstrated that inviscid transonic potential flow or Euler solutions for an airfoil can be accurately calculated on a mesh with 160 cells around the section, and 32 cells normal to the section. Using multigrid techniques 10 to 25 cycles are enough to obtain a converged result. Consequently airfoil calculations can be performed in seconds on a Cray YMP, and can also be performed on 486-class personal computers. Correspondingly accurate three-dimensional inviscid calculations can be performed for a wing on a mesh, say with $192\times32\times48 = 294,912$ cells, in about 5 minutes on a single processor Cray YMP, or less than a minute with eight processors, or in 1 or 2 hours on a work station such as a Hewlett Packard 735 or an IBM 560 model.

Viscous simulations at high Reynolds numbers require vastly greater resources. Careful two-dimensional studies of mesh requirements have been carried out at Princeton by Martinelli. He found that on the order of 32 mesh intervals were needed to resolve a turbulent boundary layer, in addition to 32 intervals between the boundary layer and the far field, leading to a total of 64 intervals. In order to prevent degradations in accuracy and convergence due to excessively large aspect ratios (in excess of 1,000) in the surface mesh cells, the chordwise resolution must also be increased to 512 intervals. Reasonably accurate solutions can be obtained in a 512×64 mesh in 100 multigrid cycles. Figure 3 shows a comparison of experimental data with a calculated result for the RAE 2822 airfoil at a Mach number of 0.729, an angle of attack of 2.31° and a Reynolds number of 6 million. Translated to three dimensions, this would imply the need for meshes with 5–10 million cells (for example, $512\times64\times256$ = 8,388,608 cells). When simulations are performed on less fine meshes with, say, 500,000 to 1 million cells, it is very hard to avoid mesh dependency in the solutions as well as sensitivity to the turbulence model.

A typical algorithm requires of the order of 5,000 floating point operations per mesh point in one multigrid iteration. With 10 million mesh points, the operation count is of the order of 0.5×10^{11} per cycle. Given a computer capable of sustaining 10^{11}
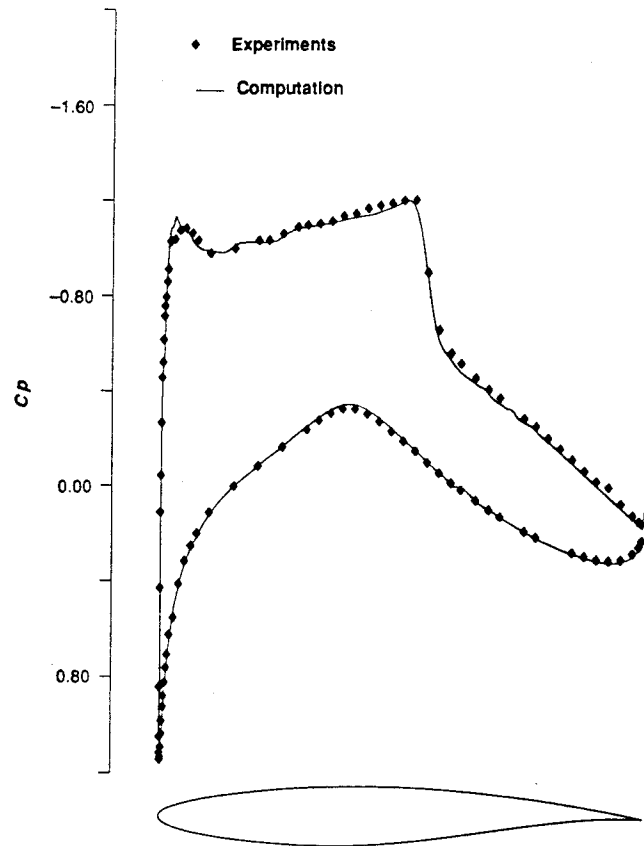

Figure 3: Comparison of the Calculated Result and Experimental Data for the RAE 2822 Airfoil at Mach 0.729 and 2.31° Angle of Attack (Supplied by Luigi Martinelli, Princeton University).

operations per second (100 gigaflops), 200 cycles could then be performed in 100 seconds. Simulations of unsteady viscous flows (flutter, buffet) would be likely to require 1,000–10,000 time steps. A further progression to large eddy simulation of complex configurations would require even greater resources. The following estimate is due to W.H. Jou [41]. Suppose that a conservative estimate of the size of eddies in a boundary layer that ought to be resolved is 1/5 of the boundary layer thickness. Assuming that 10 points are needed to resolve a single eddy, the mesh interval should then be 1/50 of the boundary layer thickness. Moreover, since the eddies are three-dimensional, the same mesh interval should be used in all three directions. Now, if the boundary layer thickness is of the order of 0.01 of the chord length, 5,000 intervals will be needed in the chordwise direction, and for a wing with an aspect ratio of 10, 50,000 intervals will be needed in the spanwise

direction. Thus, of the order of $50 \times 5,000 \times 50,000$ or 12.5 billion mesh points would be needed in the boundary layer. If the time dependent behavior of the eddies is to be fully resolved using time steps on the order of the time for a wave to pass through a mesh interval, and one allows for a total time equal to the time required for waves to travel three times the length of the chord, of the order of 15,000 time steps would be needed. Performance beyond the teraflop (10^{12} operations per second) will be needed to attempt calculations of this nature, which also have an information content far beyond what is needed for enginering analysis and design. The designer does not need to know the details of the eddies in the boundary layer. The primary purpose of such calculations is to improve the calculation of averaged quantities such as skin friction, and the prediction of global behavior such as the onset of separation. The current use of Navier-Stokes and large eddy simulations is to try to gain an improved insight into the physics of turbulent flow, which may in turn lead to the development of more comprehensive and reliable turbulence models.

It is doubtful whether a universally valid turbulence model, capable of describing all complex flows, could be devised [42]. Algebraic models [43, 44] have proved fairly satisfactory for the calculation of attached and slightly separated wing flows. These models rely on the boundary layer concept, usually incorporating separate formulas for the inner and outer layers, and they require an estimate of a length scale which depends on the thickness of the boundary layer. The estimation of this quantity by a search for a maximum of the vorticity times a distance to the wall, as in the Baldwin-Lomax model, can lead to ambiguities in internal flows, and also in complex vortical flows over slender bodies and highly swept or delta wings [45, 46]. The Johnson-King model [47], which allows for non-equilibrium effects through the introduction of an ordinary differential equation for the maximum shear stress, has improved the prediction of flows with shock induced separation [48, 49].

One-point closure models depending on the solution of transport equations are widely accepted for industrial applications. These models eliminate the need to estimate a length scale by detecting the edge of the boundary layer. Eddy viscosity models typically use two equations for the turbulent kinetic energy k and the dissipation rate ϵ, or a pair of equivalent quantities [50, 51, 52, 53, 54, 55]. Models of this type generally tend to present difficulties in the region very close to the wall. They also tend to be badly conditioned for numerical solution. In an attempt to alleviate this problem, new models requiring the solution of one transport equation have recently been introduced [56, 57]. The performance of the algebraic models remains competitive for wing flows, but the one- and two-equation models show promise for broader classes of flows. In order to achieve greater universality, research is also being pursued on more complex Reynolds stress transport models, which require the solution of a larger number of transport equations.

Another direction of research is the attempt to devise more rational models via renormalization group (RNG) theory [58, 59]. Both algebraic and two-equation $k - \epsilon$ models devised by this approach have shown promising results [60].

The selection of sufficiently accurate mathematical models and a judgment of their cost effectiveness ultimately rests with industry. Aircraft and spacecraft designs normally pass through the three phases of conceptual design, preliminary design, and detailed design. Correspondingly, the appropriate CFD models will vary in complexity. In the conceptual and preliminary design phases, the emphasis will be on relatively simple models which can give results with very rapid turn-around and low computer costs, in order to evaluate alternative configurations and perform quick parametric studies. The detailed design stage requires the most complete simulation that can be achieved with acceptable cost. In the past, the low level of confidence that could be placed on numerical predictions has forced the extensive use of wind tunnel testing at an early stage of the design. This practice was very expensive. The limited number of models that could be fabricated also limited the range of design variations that could be evaluated. It can be anticipated that in the future, the role of wind tunnel testing in the design process will be more one of verification. Experimental research to improve our understanding of the physics of complex flows will continue, however, to play a vital role.

5 Algorithms for flow simulation

The computational simulation of fluid flow presents a number of severe challenges for algorithm design. At the level of inviscid modeling, the inherent nonlinearity of the fluid flow equations leads to the formation of singularities such as shock waves and contact discontinuities. Moreover, the geometric configurations of interest are extremely complex, and generally contain sharp edges which lead to the shedding of vortex sheets. Extreme gradients near stagnation points or wing tips may also lead to numerical errors that can have global influence. Numerically generated entropy may be convected from the leading edge for example, causing the formation of a numerically induced boundary layer which can lead to separation. The need to treat exterior domains of infinite extent is also a source of difficulty. Boundary conditions imposed at artificial outer boundaries may cause reflected waves which significantly interfere with the flow. When viscous effects are also included in the simulation, the extreme difference of the scales in the viscous boundary layer and the outer flow, which is essentially inviscid, is another source of difficulty, forcing the use of meshes with extreme variations in mesh interval. For these reasons CFD, has been a driving force for the development of numerical algorithms.

The algorithm designer faces a number of critical decisions. The first choice that must be made is the nature of the mesh used to divide the flow field into discrete subdomains. The discretization procedure must allow for the treatment of complex configurations. The principal alternatives are Cartesian meshes, body-fitted curvilinear meshes, and unstructured tetrahedral meshes. Each of these approaches has advantages which have led to their use. The Cartesian mesh minimizes the complexity of the algorithm at interior points and facilitates the use of high order discretization procedures, at the expense of greater complexity, and possibly a loss of accuracy, in the treatment of boundary conditions at curved surfaces. This difficulty may be alleviated by using mesh refinement procedures near the surface. With their aid, schemes which use Cartesian meshes have recently been developed to treat very complex configurations [61, 62, 63, 64].

Body-fitted meshes have been widely used and are particularly well suited to the treatment of viscous flow because they readily allow the mesh to be compressed near the body surface. With this approach, the problem of mesh generation itself has proved to be a major pacing item. The most commonly used procedures are algebraic transformations [65, 66, 67, 68], methods based on the solution of elliptic equations, pioneered by Thompson [69, 70, 71, 72], and methods based on the solution of hyperbolic equations marching out from the body [73]. In order to treat very complex configurations it generally proves expedient to use a multiblock [74, 75] procedure, with separately generated meshes in each block, which may then be patched at block faces, or allowed to overlap, as in the Chimera scheme [76, 77]. While a number of interactive software systems for grid generation have been developed, such as EAGLE, GRIDGEN, GRAPE, and ICEM, the generation of a satisfactory grid for a very complex configuration may require months of effort.

The alternative is to use an unstructured mesh in which the domain is subdivided into tetrahedrons. This in turn requires the development of solution algorithms capable of yielding the required accuracy on unstructured meshes. This approach has been gaining acceptance, as it is becoming apparent that it can lead to a speed-up and reduction in the cost of mesh generation that more than offsets the increased complexity and cost of the flow simulations. Two competing procedures for generating triangulations which have both proved successful are Delaunay triangulation [78], based on concepts introduced at the beginning of the century by Voronoi [79], and the moving front method [80].

Associated with choice of mesh type is the formulation of the discretization procedure for the equations of fluid flow, which can be expressed as differential conservation laws. In the Cartesian tensor notation, let x_i be the coordinates, p, ρ, T, and E the pressure, density, temperature, and total energy, and u_i the velocity components. Each conservation equation has the form

$$\frac{\partial w}{\partial t} + \frac{\partial F_j}{\partial x_j} = 0. \tag{1}$$

For the mass equation

$$w = \rho, \quad F_j = \rho u_j.$$

For the i momentum equation

$$w_i = \rho u_i, \quad F_{ij} = \rho u_i u_j + p\delta_{ij} - \sigma_{ij},$$

where σ_{ij} is the viscous stress tensor. For the energy equation

$$w = \rho E, \quad F_j = (\rho E + p)u_j - \sigma_{jk}u_k - \kappa\frac{\partial T}{\partial x_j},$$

where κ is the coefficient of heat conduction. The pressure is related to the density and energy by the equation of state

$$p = (\gamma - 1)\rho\left(E - \frac{1}{2}u_i u_i\right)$$

in which γ is the ratio of specific heats. In the Navier-Stokes equations the viscous stresses are assumed to be linearly proportional to the rate of strain, or

$$\sigma_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \lambda\delta_{ij}\left(\frac{\partial u_k}{\partial x_k}\right),$$

where μ and λ are the coefficients of viscosity and bulk viscosity, and usually $\lambda = -2\mu/3$.

The finite difference method, which requires the use of a Cartesian or a structured curvilinear mesh, directly approximates the differential operators appearing in these equations. In the finite volume method [81], the discretization is accomplished by dividing the domain of the flow into a large number of small subdomains, and applying the conservation laws in the integral form

$$\frac{\partial}{\partial t}\int_{\Omega} w\,dV + \int_{\partial\Omega} \mathbf{F}\cdot d\mathbf{S} = 0.$$

Here $\mathbf{F}$ is the flux appearing in equation (1) and $d\mathbf{S}$ is the directed surface element of the boundary $\partial\Omega$ of the domain Ω. The use of the integral form has the advantage that no assumption of the differentiability of the solutions is implied, with the result that it remains a valid statement for a subdomain containing a shock wave. In general the subdomains could be arbitrary, but it is convenient to use either hexahedral cells in a body conforming curvilinear mesh or tetrahedrons in an unstructured mesh.

Alternative discretization schemes may be obtained by storing flow variables at either the cell centers or the vertices. These variations are illustrated in Figure 4 for the two-dimensional case. With a cell-centered scheme the discrete conservation law takes the form
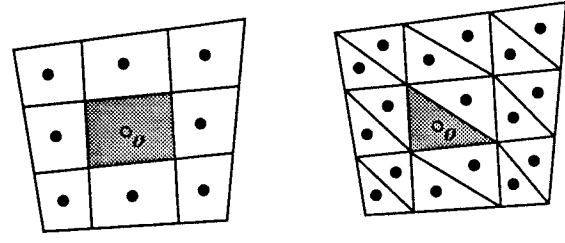
4a: Cell Centered Scheme.
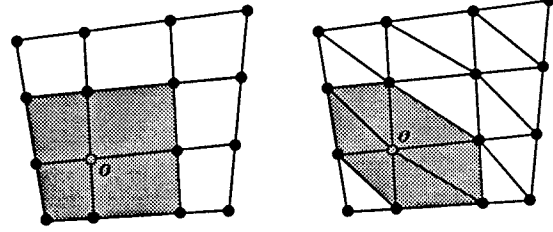
4b: Vertex Scheme.

Figure 4: Structured and Unstructured Discretizations.

$$\frac{d}{dt}wV + \sum_{\text{faces}} \mathbf{F}\cdot\mathbf{S} = 0, \tag{2}$$

where V is the cell volume, and $\mathbf{F}$ is now a numerical estimate of the flux vector through each face. $\mathbf{F}$ may be evaluated from values of the flow variables in the cells separated by each face, using upwind biasing to allow for the directions of wave propagation. With hexahedral cells, equation (2) is very similar to a finite difference scheme in curvilinear coordinates. Under a transformation to curvilinear coordinates X_j, equation (1) becomes

$$\frac{\partial}{\partial t}(Jw) + \frac{\partial}{\partial X_i}\left(J\frac{\partial X_i}{\partial x_j}F_j\right) = 0,$$

where J is the Jacobian determinant of the transformation matrix $\left[\frac{\partial x_i}{\partial X_j}\right]$. The transformed flux $J\frac{\partial X_i}{\partial x_j}F_j$ corresponds to the dot product of the flux $\mathbf{F}$ with a vector face area $J\frac{\partial X_i}{\partial x_j}$, while J represents the transformation of the cell volume. The finite volume form (2) has the advantages that it is valid for both structured and unstructured meshes, and that it assures that a uniform flow exactly satisfies the equations, because $\sum_{\text{faces}}\mathbf{S} = 0$ for a closed control volume. Finite difference schemes do not necessarily satisfy this constraint because of the discretization errors in evaluating $\frac{\partial X_i}{\partial x_j}$ and the inversion of the transformation matrix. A cell-vertex finite volume scheme can be derived by taking the union of the cells surrounding a given vertex as the control volume for that ver-

tex [82, 83, 84]. In equation (2), V is now the sum of the volumes of the surrounding cells, while the flux balance is evaluated over the outer faces of the polyhedral control volume. In the absence of upwind biasing the flux vector is evaluated by averaging over the corners of each face. This has the advantage of remaining accurate on an irregular or unstructured mesh.

An alternative route to the discrete equations is provided by the finite element method. Whereas the finite difference and finite volume methods approximate the differential and integral operators, the finite element method proceeds by inserting an approximate solution into the exact equations. On multiplying by a test function ϕ and integrating by parts over space, one obtains the weak form

$$\frac{\partial}{\partial t} \iiint_\Omega \phi w d\Omega = \iiint_\Omega \mathbf{F} \cdot \nabla \phi d\Omega - \iint_{\partial\Omega} \phi \mathbf{F} \cdot d\mathbf{S} \quad (3)$$

which is also valid in the presence of discontinuities in the flow. In the Galerkin method the approximate solution is expanded in terms of the same family of functions as those from which the test functions are drawn. By choosing test functions with local support, separate equations are obtained for each node. For example, if a tetrahedral mesh is used, and ϕ is piecewise linear, with a nonzero value only at a single node, the equations at each node have a stencil which contains only the nearest neighbors. In this case the finite element approximation corresponds closely to a finite volume scheme. If a piecewise linear approximation to the flux $\mathbf{F}$ is used in the evaluation of the integrals on the right hand side of equation (3), these integrals reduce to formulas which are identical to the flux balance of the finite volume scheme.

Thus the finite difference and finite volume methods lead to essentially similar schemes on structured meshes, while the finite volume method is essentially equivalent to a finite element method with linear elements when a tetrahedral mesh is used. Provided that the flow equations are expressed in the conservation law form (1), all three methods lead to an exact cancelation of the fluxes through interior cell boundaries, so that the conservative property of the equations is preserved. The importance of this property ensuring correct shock jump conditions was pointed out by Lax and Wendroff [85].

These procedures lead to nondissipative approximations to the Euler equations. Dissipative terms may be needed for two reasons. The first is the possibility of undamped oscillatory modes. The second reason is the need for the clean capture of shock waves and contact discontinuities without undesirable oscillations. An extreme overshoot could result in a negative value of an inherently positive quantity such as the pressure or density. Consider a general semi-discrete scheme of the form

$$\frac{d}{dt} v_j = \sum_k c_{jk}(v_k - v_j). \quad (4)$$

A maximum cannot increase and minimum cannot decrease if the coefficients c_{jk} are non-negative, since at a maximum $v_k - v_j \leq 0$, and at a minimum $v_k - v_j \geq 0$. Thus the condition

$$c_{jk} \geq 0 \quad (5)$$

is sufficient to ensure stability in the maximum norm. Moreover, if the scheme has a compact stencil, so that $c_{jk} = 0$ when j and k are not nearest neighbors, a local maximum cannot increase and local minimum cannot decrease. This local extremum diminishing (LED) property prevents the birth and growth of oscillations. The one-dimensional conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} f(u) = 0$$

provides a useful model for analysis. In this case waves are propagated with a speed $a(u) = \frac{\partial f}{\partial u}$, and the solution is constant along the characteristics $\frac{dx}{dt} = a(u)$. Thus the LED property is satisfied. In fact the total variation

$$TV(u) = \int_{-\infty}^{\infty} \left| \frac{\partial u}{\partial x} \right| dx$$

of a solution of this equation does not increase, provided that any discontinuity appearing in the solution satisfies an entropy condition [86]. Harten proposed that difference schemes ought to be designed so that the discrete total variation cannot increase [31]. If the end values are fixed, the total variation can be expressed as

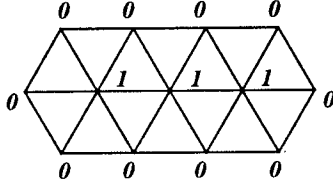$$TV(u) = 2 \left(\sum \text{maxima} - \sum \text{minima} \right).$$

Thus an LED scheme is also total variation diminishing (TVD). On a triangular mesh, a definition of total variation such as

$$TV(u) = \int ||\nabla u|| dS$$

is not an entirely satisfactory measure of oscillation. This is illustrated in Figure 5, where the total variation of two peaks is found to be less than that of a
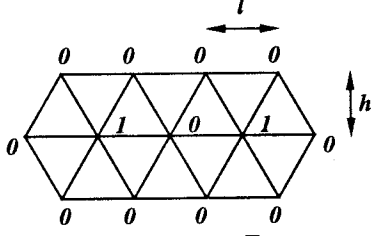
5a: Two Peaks: $TV = 4 + 2\sqrt{3}$ (L_1), 6 (L_2), or $2 + 2\sqrt{3}$ (L_∞).

5b: One Ridge: $TV = 6 + \sqrt{3}$ (L_1), 7 (L_2), or $5 + 3\sqrt{3}$ (L_∞).

Figure 5: Breakdown of TVD: The One Ridge Case is Less Oscillatory than the Two Peaks Case.

single ridge. The LED principle, however, continues to be useful for multi-dimensional problems on both structured and unstructured meshes.

Positivity conditions of the type expressed in equations (4) and (5) lead to diagonally dominant schemes, and are the key to the elimination of improper oscillations. The positivity conditions may be realized by the introduction of diffusive terms or by the use of upwind biasing in the discrete scheme. Unfortunately, they may also lead to severe restrictions on accuracy unless the coefficients have a complex nonlinear dependence on the solution. Following the pioneering work of Godunov [87], a variety of dissipative and upwind schemes designed to have good shock capturing properties have been developed during the past two decades [30, 88, 89, 90, 91, 92, 31, 93, 94, 95, 96, 97, 98].

A conservative semidiscrete approximation to the one-dimensional conservation law can be derived by subdividing the line into cells. Then the evolution of the value v_j in the jth cell is given by

$$\Delta x \frac{d}{dt} v_j + \left(h_{j+\frac{1}{2}} - h_{j-\frac{1}{2}} \right) = 0,$$

where $h_{j+\frac{1}{2}}$ is an estimate of the flux between cells

j and $j + 1$. The simplest estimate is the arithmetic average $(f_{j+1} + f_j)/2$ but this leads to a scheme that does not satisfy the positivity conditions. To correct this, one may add a dissipative term and set

$$h_{j+\frac{1}{2}} = \frac{1}{2} \left(f_{j+1} + f_j \right) - \alpha_{j+\frac{1}{2}} \left(v_{j+1} - v_j \right).$$

In order to estimate the required value of the coefficient $\alpha_{j+\frac{1}{2}}$, let $a_{j+\frac{1}{2}}$ be a numerical estimate of the wave speed $\frac{\partial f}{\partial u}$,

$$a_{j+\frac{1}{2}} = \begin{cases} \frac{f_{j+1}-f_j}{v_{j+1}-v_j} & \text{if } v_{j+1} \neq v_j \\ \left. \frac{\partial f}{\partial v} \right|_{v=v_j} & \text{if } v_{j+1} = v_j \end{cases}.$$

Now

$$\begin{aligned} h_{j+\frac{1}{2}} &= f_j + \frac{1}{2} \left(f_{j+1} - f_j \right) - \alpha_{j+\frac{1}{2}} \left(v_{j+1} - v_j \right) \\ &= f_j - \left(\alpha_{j+\frac{1}{2}} - \frac{1}{2} a_{j+\frac{1}{2}} \right) \left(v_{j+1} - v_j \right) \end{aligned}$$

and similarly

$$h_{j-\frac{1}{2}} = f_j - \left(\alpha_{j-\frac{1}{2}} + \frac{1}{2} a_{j-\frac{1}{2}} \right) \left(v_j - v_{j-1} \right).$$

Thus the positivity conditions are satisfied if

$$\alpha_{j+\frac{1}{2}} \geq \frac{1}{2} \left| a_{j+\frac{1}{2}} \right| \tag{6}$$

for all j. The minimum sufficient value of one half the wave speed produces the upwind scheme

$$h_{j+\frac{1}{2}} = \begin{cases} f_j & \text{if } a_{j+\frac{1}{2}} \geq 0 \\ f_{j+1} & \text{if } a_{j+\frac{1}{2}} < 0 \end{cases}.$$

It may be noted that the successful treatment of transonic potential flow also involved the use of upwind biasing. This was first introduced by Murman and Cole to treat the transonic small disturbance equation [7]. The author's rotated difference scheme [9], which extended their technique to treat the general transonic potential flow equation, proved to be very robust. TVD schemes can yield sharp discrete shock waves without oscillations, but in this simple form they are at best first- order accurate.

Higher order non-oscillatory schemes can be derived by introducing anti-diffusive terms in a controlled manner. If the first order diffusive flux $\alpha_{j+\frac{1}{2}} (v_{j+1} - v_j)$ is replaced by the third order diffusive flux

$$d_{j+\frac{1}{2}} = \alpha_{j+\frac{1}{2}} \left\{ \Delta v_{j+\frac{1}{2}} - \frac{1}{2} \left(\Delta v_{j+\frac{3}{2}} + \Delta v_{j-\frac{1}{2}} \right) \right\}$$

where $\Delta v_{j+\frac{1}{2}}$ denotes $v_{j+1} - v_j$, the positivity condition (5) is violated because of the presence of the anti-diffusive terms $\frac{1}{2}\left(\Delta v_{j+\frac{3}{2}} + \Delta v_{j-\frac{1}{2}}\right)$. The Jameson-Schmidt-Turkel (JST) scheme [99], addressed this problem by blending first and third order diffusive fluxes. The ratio of the blend is controlled by a sensor of flow gradients, usually the second difference of the pressure. This is quite effective in steady state calculations. An alternative route to high resolution is to limit the anti-diffusion through the introduction of flux limiters dependent on the ratios of slopes $\Delta v_{j+\frac{1}{2}}$ and $\Delta v_{j-\frac{1}{2}}$ in adjacent cells, which guarantee the satisfaction of the positivity condition. The use of limiters can be traced to the work of Boris and Book with their flux corrected transport (FCT) method [88], which was generalized and improved by Zalesak [100]. Flux limiters were independently introduced by Van Leer [89].

A particularly simple way to introduce limiters, proposed by the author [96], is to use flux limited dissipation. This method produces a symmetric TVD scheme similar to that subsequently proposed by Yee [97]. It may be implemented as follows. Let $L(u,v)$ be a limited average of u and v with the following properties:

P1. $L(u,v) = L(v,u)$

P2. $L(\alpha u, \alpha v) = \alpha L(u,v)$

P3. $L(u,u) = u$

P4. $L(u,v) = 0$ if u and v have opposite signs

Properties (P1–P3) are natural properties of an average. Property P4 is needed for the construction of an LED or TVD scheme. Consistent with these properties

$$L(u,v) = \phi\left(\frac{v}{u}\right)u = \phi\left(\frac{u}{v}\right)v.$$

where $\phi(r) \geq 0$ when $r \geq 0$ and $\phi(r) = 0$ when $r < 0$. Now replace the arithmetic average $\frac{1}{2}\left(\Delta v_{j+\frac{3}{2}} + \Delta v_{j-\frac{1}{2}}\right)$ by the limited average so that the diffusive flux becomes

$$d_{j+\frac{1}{2}} = \alpha_{j+\frac{1}{2}}\left\{\Delta v_{j+\frac{1}{2}} - L\left(\Delta v_{j+\frac{3}{2}}, \Delta v_{j-\frac{1}{2}}\right)\right\}.$$

It may then be verified that the positivity condition (5) is satisfied if (6) holds. This symmetric limited positive (SLIP) scheme can also be implemented for multi-dimensional problems on unstructured grids [101].

In order to apply these ideas to a system of equations one may split the flux into components corresponding to the different wave speeds. This concept was first proposed by Steger and Warming [30]. Roe's flux difference splitting method [91] has proved particularly effective. Consider the one-dimensional case, which may be written in vector notation as

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x} = \mathbf{0},$$

where $\mathbf{w}$ is the vector of dependent variables, and $\mathbf{f}(\mathbf{w})$ is the flux vector. The wave speeds of the system (1) are the eigenvalues of the Jacobian matrix $\frac{\partial \mathbf{f}}{\partial \mathbf{w}}$. Let $A_{j+\frac{1}{2}}$ be a matrix depending on the values $\mathbf{w}_{j+1}$ and $\mathbf{w}_j$ such that

$$\mathbf{f}_{j+1} - \mathbf{f}_j = A_{j+\frac{1}{2}}\left(\mathbf{w}_{j+1} - \mathbf{w}_j\right).$$

Now decompose this matrix as $A_{j+\frac{1}{2}} = T\Lambda T^{-1}$, where the columns of T are the eigenvectors of $A_{j+\frac{1}{2}}$, and Λ is a diagonal matrix containing the eigenvalues. Then the low order diffusive flux may be calculated as $\frac{1}{2}T\left|\Lambda\right|T^{-1}\left(\mathbf{w}_{j+1} - \mathbf{w}_j\right)$, which corresponds to upwinding the characteristic variables. A simple non-oscillatory scheme is obtained by the use of a scalar diffusive coefficient $\alpha_{j+\frac{1}{2}} \geq \frac{1}{2}\max\left|\lambda\left(A_{j+\frac{1}{2}}\right)\right|$. Blended first and third order scalar diffusion is quite effective for steady flow calculations [99], and it can be improved by the addition of pressure difference to the momentum diffusion [101].

Characteristic splitting has the advantage that it allows a discrete shock structure with a single interior point. In multi-dimensional problems it may be implemented separately in each mesh direction. A substantial body of current research is directed toward the implementation of truly multi-dimensional upwind schemes [102, 103, 104]. Some of the most impressive simulations of time dependent flows with strong shock waves have been achieved with higher order Godunov schemes [105]. In these schemes the average value in each cell is updated by applying the integral conservation law using interface fluxes predicted from the exact or approximate solution of a Riemann problem between adjacent cells. A higher order estimate of the solution is then reconstructed from the cell averages, and slope limiters are applied to the reconstruction. An example is the class of essentially non-oscillatory (ENO) schemes, which can attain a very high order of accuracy at

11

the cost of a substantial increase in computational complexity [32, 106, 107, 108]. Methods based on reconstruction can also be implemented on unstructured meshes [109, 110]. Recently there has been an increasing interest in kinetic flux splitting schemes, which use solutions of the Boltzmann equation to predict the interface fluxes [111, 112, 113, 114]

The discretization of the viscous terms of the Navier-Stokes equations requires an approximation to the velocity derivatives $\frac{\partial u_i}{\partial x_j}$ in order to calculate the stress tensor. Usually these are calculated by central difference formulas. A desirable property is that a linearly varying velocity distribution, as in a Couette flow, should produce a constant viscous stress and hence an exact stress balance. This property is not necessarily satisfied by finite difference schemes or finite volume schemes on curvilinear meshes. The characterization k-exact has been proposed for schemes that are exact for polynomials of degree k [115]. Finite volume schemes that are linearly exact can be constructed by using the same type of formula to calculate the velocity and metric derivatives $\frac{\partial u_i}{\partial X_j}$, and $\frac{\partial x_i}{\partial X_j}$ and setting $\left[\frac{\partial u}{\partial x}\right] = \left[\frac{\partial x}{\partial X}\right]^{-1}\left[\frac{\partial u}{\partial X}\right]$. This formulation yields a constant stress when u is linear, and in turn a viscous stress balance. In the case of an unstructured mesh the weak form (3) leads to a natural discretization with linear elements, in which the piecewise linear approximation yields a constant stress in each cell. This method produces a representation that is globally correct when averaged over the cells, as is proved by energy estimates for elliptic problems [116]. It should be noted however, that it yields formulas that are not necessarily locally consistent with the differential equations, if Taylor series expansions are substituted for the solution at the vertices appearing in the local stencil. Thus the use of an irregular triangulation in the boundary layer may significantly degrade the accuracy.

If the space discretization procedure is implemented separately, it leads to a set of coupled ordinary differential equations, which can be written in the form

$$\frac{d\mathbf{w}}{dt} + \mathbf{R}(\mathbf{w}) = 0,$$

where $\mathbf{w}$ is the vector of the flow variables at the mesh points, and $\mathbf{R}(\mathbf{w})$ is the vector of the residuals, consisting of the flux balances defined by the space discretization scheme, together with the added

dissipative terms. If the objective is simply to reach the steady state and details of the transient solution are immaterial, the time-stepping scheme may be designed solely to maximize the rate of convergence. The first decision that must be made is whether to use an explicit scheme, in which the space derivatives are calculated from known values of the flow variables at the beginning of the time step, or an implicit scheme, in which the formulas for the space derivatives include as yet unknown values of the flow variables at the end of the time step, leading to the need to solve coupled equations for the new values. The permissible time step for an explicit scheme is limited by the Courant-Friedrichs-Lewy (CFL) condition, which states that a difference scheme cannot be a convergent and stable approximation unless its domain of dependence contains the domain of dependence of the corresponding differential equation. One can anticipate that implicit schemes will yield convergence in a smaller number of time steps, because the time step is no longer constrained by the CFL condition. Implicit schemes will be efficient, however, only if the decrease in the number of time steps outweighs the increase in the computational effort per time step consequent upon the need to solve coupled equations. The prototype implicit scheme can be formulated by estimating $\frac{\partial \mathbf{w}}{\partial t}$ at $t + \mu \Delta t$ as a linear combination of $\mathbf{R}(\mathbf{w}^n)$ and $\mathbf{R}(\mathbf{w}^{n+1})$. The resulting equation

$$\mathbf{w}^{n+1} = \mathbf{w}^n - \Delta t \left\{ (1-\mu)\,\mathbf{R}\left(\mathbf{w}^n\right) + \mu\mathbf{R}\left(\mathbf{w}^{n+1}\right)\right\}$$

can be linearized as

$$\left(\mathbf{I} + \mu\Delta t \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right)\delta\mathbf{w} + \Delta t\mathbf{R}\left(\mathbf{w}^n\right) = 0.$$

If one sets $\mu = 1$ and lets $\Delta t \to \infty$ this reduces to the Newton iteration , which has been successfully used in two-dimensional calculations [117, 118]. In the three-dimensional case with, say, an $N \times N \times N$ mesh, the bandwidth of the matrix that must be inverted is of order N^2. Direct inversion requires a number of operations proportional to the number of unknowns multiplied by the square of the bandwidth of the order of N^7. This is prohibitive, and forces recourse to either an approximate factorization method or an iterative solution method.

Alternating direction methods, which introduce factors corresponding to each coordinate, are widely used for structured meshes [22, 23]. They cannot be implemented on unstructured tetrahedral

meshes that do not contain identifiable mesh directions, although other decompositions are possible [119]. If one chooses to adopt the iterative solution technique, the principal alternatives are variants of the Gauss-Seidel and Jacobi methods. A symmetric Gauss-Seidel method with one iteration per time step is essentially equivalent to an approximate lower-upper (LU) factorization of the implicit scheme [120, 121, 122, 123]. On the other hand, the Jacobi method with a fixed number of iterations per time step reduces to a multistage explicit scheme, belonging to the general class of Runge-Kutta schemes [124]. Schemes of this type have proved very effective for wide variety of problems, and they have the advantage that they can be applied equally easily on both structured and unstructured meshes [99, 29, 125, 126].

Let $\mathbf{w}^n$ be the result after n steps. The general form of an m-stage scheme is

$$
\begin{aligned}
\mathbf{w}^{(0)} &= \mathbf{w}^n \\
\mathbf{w}^{(1)} &= \mathbf{w}^{(0)} - \alpha_1 \Delta t \mathbf{R}^{(0)} \\
\cdots & \qquad \cdots \\
\mathbf{w}^{(m-1)} &= \mathbf{w}^{(0)} - \alpha_{m-1} \Delta t \mathbf{R}^{(m-2)} \\
\mathbf{w}^{(m)} &= \mathbf{w}^{(0)} - \Delta t \mathbf{R}^{(m-1)} \\
\mathbf{w}^{(n+1)} &= \mathbf{w}^{(m)}.
\end{aligned}
$$

In cases where only the steady state solution is needed, it is helpful to separate the residual $\mathbf{R}(\mathbf{w})$ into its convective and dissipative parts $\mathbf{Q}(\mathbf{w})$ and $\mathbf{D}(\mathbf{w})$. Then the residual in the $(q+1)$ st stage is evaluated as

$$
\mathbf{R}^{(q)} = \sum_{r=0}^{q} \left\{ \beta_{qr} \mathbf{Q}\left(\mathbf{w}^{(r)}\right) + \gamma_{qr} \mathbf{D}\left(\mathbf{w}^{(r)}\right) \right\},
$$

where

$$
\sum_{r=0}^{q} \beta_{qr} = 1, \sum_{r=0}^{q} \gamma_{qr} = 1.
$$

Blended multistage schemes of this type [125], can be tailored to give large stability intervals along both the imaginary and negative real axes.

Radical improvements in the rate of convergence to a steady state can be realized by the multigrid time-stepping technique. The concept of acceleration by the introduction of multiple grids was first proposed by Fedorenko [13]. There is by now a fairly well-developed theory of multigrid methods for elliptic equations based on the concept that the updating scheme acting as a smoothing operator on each

grid [14, 19]. This theory does not hold for hyperbolic systems. Nevertheless, it seems that it ought to be possible to accelerate the evolution of a hyperbolic system to a steady state by using large time steps on coarse grids so that disturbances will be more rapidly expelled through the outer boundary. Various multigrid time-stepping schemes designed to take advantage of this effect have been proposed [27, 28, 82, 83, 127, 128, 129, 130, 131].

One can devise a multigrid scheme using a sequence of independently generated coarser meshes by eliminating alternate points in each coordinate direction. In order to give a precise description of the multigrid scheme, subscripts may be used to indicate the grid. Several transfer operations need to be defined. 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 value 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 grid k is driven by the residuals calculated on grid $k - 1$. This can be accomplished by setting

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

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

$$
\begin{aligned}
w_k^{(1)} &= w_k^{(0)} - \alpha_1 \Delta t_k \left[R_k^{(0)} + P_k\right] \\
\cdots & \qquad \cdots \\
w_k^{(q+1)} &= w_k^{(0)} - \alpha_{q+1} \Delta t_k \left[R_k^{(q)} + P_k\right].
\end{aligned}
$$

The result $w_k^{(m)}$ then provides the 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 $I_{k-1,k}$. With properly optimized coefficients multistage time-stepping schemes can be very efficient drivers of the multigrid process. A W-cycle of the type illustrated in Figure 6 proves to be a particularly effective strategy for managing the work split between the meshes. In a three-dimensional case the number of cells is reduced by a factor of eight on each coarser grid. On examination of the figure, it can therefore be seen that the work measured in
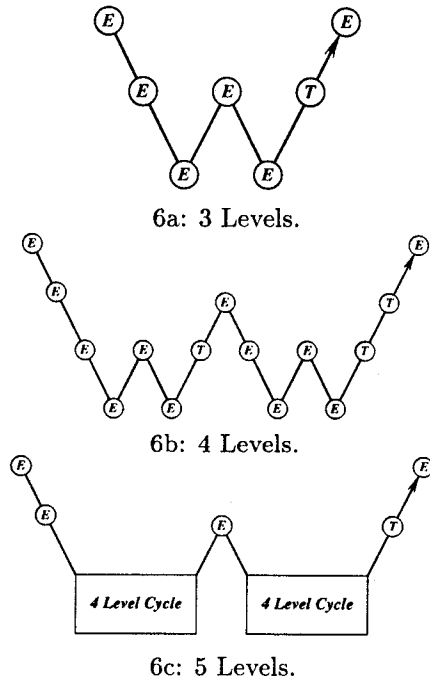
6a: 3 Levels.

6b: 4 Levels.

6c: 5 Levels.

Figure 6: Multigrid W-cycle for managing the grid calculation. E, evaluate the change in the flow for one step; T, transfer the data without updating the solution.

units corresponding to a step on the fine grid is of the order of

$$1 + 2/8 + 4/64 + \ldots < 4/3,$$

and consequently the very large effective time step of the complete cycle costs only slightly more than a single time step in the fine grid.

The need both to improve the accuracy of computational simulations, and to assure known levels of accuracy is the focus of ongoing research. The main routes to improving the accuracy are to increase the order of the discrete scheme, and reduce the mesh interval. High order difference methods are most easily implemented on Cartesian, or at least extremely smooth grids. The expansion of the stencil as the order is increased leads to the need for complex boundary conditions. Compact schemes keep the stencil as small as possible [132, 133, 134]. On simple domains, spectral methods are particularly effective, especially in the case of periodic boundary conditions, and can be used to produce exponentially fast convergence of the error as the mesh interval is decreased [135, 136]. A compromise is to divide the field into subdomains and introduce high order elements. This approach is used in the spectral element method [137].

High order difference schemes and spectral methods have proven particularly useful in direct Navier-Stokes simulations of transient and turbulent flows. High order methods are also beneficial in computational aero-acoustics, where it is desired to track waves over long distances with minimum error. If the flow contains shock waves or contact discontinuities, the ENO method may be used to construct high order non-oscillatory schemes.

In multi-dimensional flow simulations, global reduction of the mesh interval can be prohibitively expensive, motivating the use of adaptive mesh refinement procedures which reduce the local mesh width h if there is an indication that the error is too large [138, 139, 140, 141, 142, 143]. In such h-refinement methods, simple error indicators such as local solution gradients may be used. Alternatively, the discretization error may be estimated by comparing quantities calculated with two mesh widths, say on the current mesh and a coarser mesh with double the mesh interval. Procedures of this kind may also be used to provide *a posteriori* estimates of the error once the calculation is completed.

This kind of local adaptive control can also be applied to the local order of a finite element method to produce a p-refinement method, where p represents the order of the polynomial basis functions. Finally, both h- and p- refinement can be combined to produce an h-p method in which h and p are locally optimized to yield a solution with minimum error [144]. Such methods can achieve exponentially fast convergence, and are well established in computational solid mechanics.

6 Software Design, Development, Maintenance and Validation

Given a well formulated mathematical model and a stable and accurate numerical approximation scheme, the difficulties of implementation in computer programs are often underestimated. With the use of CFD for increasingly complex simulations, CFD software is becoming correspondingly more difficult to develop and maintain. Even with a single

14

author it is hard to eliminate programming errors. When code development is undertaken by a number of authors, perhaps changing over time, there may be no one who knows precisely what is contained in a large software package. It becomes essential to use structured and carefully controlled programming techniques. If uncontrolled changes are permitted, a program which was thought to be validated on Monday may no longer be valid by Wednesday, because of newly introduced errors due to improper code modifications. In the field of structural analysis it is normal practice to rely on commercial software which is carefully validated and well supported. Outside the Aircraft Industry, a variety of commercial software packages are also used, but these products have yet to gain acceptance within the Industry for aerodynamic design.

In order to increase the confidence level in the use of CFD in industry, there is a strong desire that codes should be validated. In considering this requirement, it is important to distinguish between the issues of the correctness of the program and the suitability of the mathematical model to the intended application. Simply comparing experimental data with numerical results provides no way to distinguish the source of discrepancies, whether they are due to bad numerical approximation, to programming errors, or to deviations between the mathematical model and the true physics. Some measures that may be taken to assure program correctness are as follows:

1. Modular programming: it should then be possible to substitute logically identical but independently programmed versions of each module (with different loop structures, for example) and retain the identical results.

2. Mesh refinement studies: does the program give a definite answer as the mesh interval is successively decreased?

3. Consistency checks for established mathematical properties: for example, does a wing with a symmetric section produce no lift at zero angle of attack, or in steady inviscid flow is the stagnation enthalpy constant?

Ideally the numerical scheme and computer program should be checked and internally validated by tests of this kind to the point that discrepancies with experimental data can properly be attributed to inadequacy of the mathematical model, and not the computer program itself.

The software may include options to treat a range of mathematical models appropriate to different stages of the design process, and to a variety of applications such as internal and external flows, with or without chemical reaction. Assuming that the software is demonstrated to provide correct solutions to the mathematical models it contains, the criteria for its evaluation are:

1. The level of accuracy that it provides over a range of applications.

2. The computational costs and also the manpower costs associated with carrying out the simulations.

3. The turn-around time.

If either the computational or the manpower costs are excessive, the simulations are not viable for industrial use. Moreover, if the turn-around cannot be achieved within hours, or overnight, the simulations do not provide an effective tool for the aerodynamic design process. Interactive design requires turn-around of a few minutes, with immediate computer access. One route to this goal is the use of distributed computing equipment, with a network of linked workstations that are powerful enough to give the rapid turn-around that is needed.

7 Current status of numerical aerodynamic simulation

At this stage, the evolution of CFD has reached a plateau on which the key enabling problems for inviscid flow calculations are rather well understood. For example, a variety of high resolution non-oscillatory shock capturing schemes are available, while multigrid and other acceleration methods allow converged solutions to be obtained in 10–50 cycles. Effective techniques have been developed to treat complex configurations with both structured and unstructured meshes. As an example of a current industrial application, Plate 1 shows the results of Euler calculations used in the design of the Northrop YF-23. These calculations were performed

by R.J. Busch, Jr. using an O-O mesh [145]. The flow solver was the author's FLO57 program. It can be seen that the comparison with experimental data is quite good even at a fairly high angle of attack, and valuable information for the aerodynamic design was obtained by the use of a relatively inexpensive computational model.

Unstructured meshes offer the flexibility to treat arbitrarily complex configurations without the need to spend months developing an acceptable mesh. The next two plates show the results of Euler calculations using the AIRPLANE code [35]. Recent advances in the performance of microprocessors have made it possible to perform calculations like these for complete aircraft or space vehicles on workstations, in this case the IBM RISC 6000 series. Plate 3 shows calculations for supersonic transport configurations. The agreement with experimental data is quite good, and it has also been possible to predict the sonic boom signature [146]. The US Space Shuttle was simulated on a mesh with about 400,000 points, and the calculation required about 12 hours with an IBM 560 model workstation. The MD11 calculation was performed on a mesh with about 645,000 points. Forthcoming workstations will be several times faster. Thus at the level of inviscid modeling, complete configurations can easily be analyzed with distributed computing equipment readily accessible by engineering design groups.

Plates 4 and 5 show other examples of the use of unstructured meshes. Plate 4, provided by D.J. Mavriplis shows the simulation of a wing body nacelle combination on a mesh with 800,000 points, using a multigrid acceleration procedure [147], which yielded a three-fold reduction in computational costs. Plate 5 illustrates the power of unstructured meshes to treat configurations with moving boundaries by redistributing and reconnecting the mesh points. This calculation, performed by R. Löhner, simulates the flow over an emerging ejection seat. [148].

The analysis of viscous flow continues to present a severe challenge. Plate 6 shows direct Navier-Stokes (DNS) simulations of transition to turbulence. The first, supplied by M.Y. Hussaini, is an incompressible simulation using a spectral method [149]. The second, supplied by M.M. Rai, shows the result of a simulation at a Mach number of 0.1 and a Reynolds

number of 50,000/inch using a high order upwind biased finite difference scheme [132]. Viscous flow simulations of full configurations are feasible only with Reynolds averaging, and are still very expensive, while the results may be quite sensitive to the choice of turbulence models. Calculations which are representative of the current state of the art are presented in Plates 7–11. The first shows an analysis of a complete F-18 at a high angle of attack performed by Cummings et al. at NASA Ames [150]. They used a multiblock with about 900,000 mesh points. The agreement with the experimental data and visualization are quite good. Plate 8 shows a similar calculation performed by Ghaffari et al. at NASA Langley, which exhibits equally good agreement with the data [151]. They also used a multiblock mesh, in this case with about 1.24 million mesh points. The computational costs of these simulations are outside the acceptable range for routine use in the aerodynamic design process. Each of the calculations by Ghaffari et al. required 20–24 hours on the NAS Cray 2. The multiple authorship of each of these papers provides an indication of the magnitude of the effort absorbed by these calculations. In fact a team of ten persons has been working on the generation of a 16 million point multiblock mesh for the analysis of the Space Shuttle Launch Vehicle [152].

Plate 9 shows viscous simulations of the ONERA M6 wing using a variety of turbulence models. These results were obtained by C. Rumsey and V. Vatsa [48]. At lower angles of attack the various models agree quite well both with each other and with the experimental data. At higher angles of attack there is a complete divergence of the results. This illustrates that the accuracy attainable in viscous simulations continues to be limited by the accuracy of the turbulence modeling. The confidence level that can be placed in currently available models decreases sharply in the presence of significant flow separation. On the other hand, Plate 10, supplied by J. Yu, shows that useful wing simulations can be performed for modern transport aircraft [49]. An example of the prediction of a high lift configuration is shown in Plate 11, a calculation by S. Rogers [153].

Plates 12 and 13 illustrate the flexibility of numerical simulation with two examples of entirely different geometric configurations. Plate 12, supplied by

M.J. Siclari, shows the flow over a magnetically levitated (MAGLEV) train [154]. Plate 13, supplied by Kiris *et al.*, shows the simulation of the flow through a turbopump inducer [155]. Finally, Plates 14 and 15 illustrate simulations of flows that could not readily be duplicated in wind tunnels, because of the unusual or extreme conditions. Plate 14 shows a simulation of a hypersonic forebody by S. Lawrence [156], while Plate 15, supplied by H. Rieger [157], shows a simulation of the Hermes Space Shuttle.

The broad range of these calculations indicates the extent to which CFD is now a mature science. Inviscid simulations are now a routine design tool. Viscous calculations still present a challenge both to improve their reliability and accuracy, and to reduce their cost and turn-around time. CFD has played a particularly important role in the analysis of hypersonic vehicles such as the US Aerospace Plane and the European Space Shuttle, which could not be adequately tested in experimental facilities. A survey of computational studies of the Hermes which have been carried out at institutions throughout Europe, is given by Periaux *et al.* [158]. Thus, the first of the goals cited by Chapman, Mark and Pirtle, to provide otherwise infeasible simulations has indeed been realized.

8 Outlook and Conclusions

Better algorithms and better computer hardware have contributed about equally to the progress of computational science in the last two decades. In 1970 the Control Data 6600 represented the state of the art in computer hardware with a speed of about 10^6 operations per second (one megaflop), while in 1990 the 8 processor Cray YMP offered a performance of about 10^9 operations per second (one gigaflop). Correspondingly, steady state Euler calculations which required 5,000–10,000 steps prior to 1980 could be performed in 10–50 steps in 1990 using multigrid acceleration. With the advent of massively parallel computers it appears that the progress of computer hardware may even accelerate. Teraflop machines offering further improvement by a factor of 1,000 are likely to be available within a few years. Parallel architectures will force a reappraisal of existing algorithms, and their effective utilization will require the extensive development of new parallel

software.

One may ask whether comparable improvements can be anticipated in the efficiency of algorithms. Clearly there is no possibility of a thousandfold reduction in the number of cycles in calculations already being completed in less than 100 cycles. The only possibility of a radical reduction in the number of arithmetic operations would lie in improved discretization methods, such as higher order schemes and adaptive mesh refinement procedures. The value of higher order methods for appropriate applications, with sufficient smoothness in the solution, has been demonstrated in computational studies of transition from laminar to turbulent flow such as those illustrated in Plate 6. Adaptive mesh refinement is currently the subject of widespread and intensive investigation. While there are ongoing research efforts directed toward the development of optimal refinement techniques, automatic mesh refinement procedures have also reached the stage of production use in the Dassault finite element codes and Boeing's TRANAIR code.

In parallel with the transition to more sophisticated algorithms, the present challenge is to extend the effective use of CFD to more complex applications. A key problem is the treatment of multiple space and time scales. These arise not only in turbulent flows, but also in many other situations such as chemically reacting flows, combustion, flame fronts and plasma dynamics. Another challenge, is presented by problems with moving boundaries. Examples include helicopter rotors, and rotor-stator interaction in turbomachinery. Algorithms for these problems can be significantly improved by innovative concepts, such as the idea of time inclining. It can be anticipated that interdisciplinary applications in which CFD is coupled with the computational analysis of other properties of the design will play an increasingly important role. These applications may include structural, thermal and electromagnetic analysis. Aeroelastic problems and integrated control system and aerodynamic design are likely target areas.

Research must also be focused on the best way to use CFD to arrive at good designs. Inverse methods which determine shapes corresponding to specified pressure distributions are already in use [159, 160]. As the confidence level in CFD increases

and the computational costs decrease, it will be natural to combine CFD with automatic optimization techniques to derive superior designs [161]. One approach to this, explored by the present author [162, 163], is to regard the design problem as a control problem for a system governed by a partial differential equation (the flow equations) with boundary control (the shape of the boundary). The (infinitely dimensional) Frechet derivative of the cost function with respect to the shape can be calculated by solving a single adjoint partial differential equation, at a cost comparable to a flow solution. This method, which eliminates the need to estimate sensitivity coefficients by varying each parameter in turn, has been successfully applied to the design of transonic airfoils. Plate 16 shows an example in which an airfoil was modified in 8 iterations to essentially eliminate shock drag at Mach 0.72, while also satisfying a constraint limiting deviations from the original pressure distribution at a second subsonic design point. Such a technique promises opportunities for improvement in a wide variety of applications. It might, for example, be used for the design of a wing for a supersonic transport aircraft, or the design of hydrofoils to reduce cavitation.

The development of improved algorithms continues to be important in providing the basic building blocks for a numerical wind tunnel. In particular, better error estimation procedures must be developed and incorporated in the simulation software to provided error control. The basic simulation software is only one of the needed ingredients, however. The flow solver must be embedded in a user-friendly system for geometry modeling, output analysis, and data management that will provide a complete numerical design environment (NUDE).

In the long run, computational simulation should become the principal tool of the aerodynamic design process because of the flexibility it provides for the rapid and comparatively inexpensive evaluation of alternative designs, and because it can be integrated with a multi-disciplinary optimization (MDO) procedure. Experimental facilities are likely to be used principally for fundamental investigations of the basic physics of fluid flow, and for final verification of the design prior to flight testing. This is already the accepted procedure in the structural design process. The vision of Chapman, Mark, and Pirtle was

prescient. Although much remains to be done, the author believes that the numerical wind tunnel is certainly going to be a reality.

9   Acknowledgements

References

[1] D.R. Chapman, H. Mark, and M.W. Pirtle. Computers vs wind tunnels in aerodynamic flow simulations. *Astronautics and Aeronautics*, 13(4):22–30 and 35, 1975.

[2] J.L. Hess and A.M.O. Smith. Calculation of non-lifting potential flow about arbitrary three-dimensional bodies. Douglas Aircraft Report ES 40622, 1962.

[3] P.E. Rubbert and G.R. Saaris. A general three-dimensional potential flow method applied to V/STOL aerodynamics. SAE Paper 680304, 1968.

[4] F. Woodward. An improved method for the aerodynamic analysis of wing-body-tail configurations in subsonic and supersonic flow, Part 1 — theory and application. NASA CR 2228 Pt. 1, May 1973.

[5] F. Bauer, P. Garabedian, D. Korn, and A. Jameson. *Supercritical Wing Sections II.* Springer-Verlag, 1970. Lecture Notes in Economics and Mathematical Systems.

[6] R.T. Whitcomb. Review of NASA supercritical airfoils. In *9th ICAS Congress*, number 74-10, Haifa, 1974.

[7] E.M. Murman and J.D. Cole. Calculation of plane steady transonic flows. *AIAA Journal*, 9:114–121, 1971.

[8] E.M. Murman. Analysis of embedded shock waves calculated by relaxation methods. *AIAA Journal*, 12:626–633, 1974.

[9] A. Jameson. Iterative solution of transonic flows over airfoils and wings, including flows at Mach 1. *Comm. Pure. Appl. Math*, 27:283–309, 1974.

[10] A. Jameson. Transonic potential flow calculations in conservation form. In *Proc. AIAA 2nd Computational Fluid Dynamics Conference*, pages 148–161, Hartford, 1975.

[11] A. Eberle. A finite volume method for calculating transonic potential flow around wings from the minimum pressure integral. NASA TM 75324, 1978. Translated from MBB UFE 1407(0).

[12] M. Hafez, J.C. South, and E.M. Murman. Artificial compressibility method for numerical solutions of the transonic full potential equation. *AIAA Journal*, 17:838–844, 1979.

[13] R.P. Fedorenko. The speed of convergence of one iterative process. *USSR Comp. Math. and Math. Phys.*, 4:227–235, 1964.

[14] A. Brandt. Multi-level adaptive solutions to boundary value problems. *Math. Comp.*, 31:333–390, 1977.

[15] T.L. Holst and W.F. Ballhaus. Fast conservative schemes for the full potential equation applied to transonic flows. *AIAA Journal*, 17:145–152, 1979.

[16] W.F. Ballhaus, A. Jameson, and J. Albert. Implicit approximate factorization schemes for steady transonic flow problems. *AIAA Journal*, 16:573–579, 1978.

[17] T.L. Holst and S.D. Thomas. Numerical solution of transonic wing flow fields. AIAA Paper 82-0105, 1982.

[18] J.C. South and A. Brandt. Application of a multi-level grid method to transonic flow calculations. In T.C. Adamson and M.F. Platzer, editors, *Proc. of Workshop on Transonic Flow Problems in Turbomachinery*, pages 180–206, Monterey, 1976. Hemisphere, 1977.

[19] W. Hackbusch. On the multi-grid method applied to difference equations. *Computing*, 20:291–306, 1978.

[20] A. Jameson. Acceleration of transonic potential flow calculations on arbitrary meshes by the multiple grid method. In *Proc. AIAA 4th Computational Fluid Dynamics Conference*, pages 122–146, Williamsburg, 1979.

[21] R.W. MacCormack. The effect of viscosity in hyper-velocity impact cratering. AIAA Paper 69-354, 1969.

[22] R.W. Beam and R.F. Warming. An implicit finite difference algorithm for hyperbolic systems in conservation form. *J. Comp. Phys.*, 23:87–110, 1976.

[23] T.H. Pulliam and J.L. Steger. Implicit finite difference simulations of three-dimensional compressible flow. *AIAA Journal*, 18:159–167, 1980.

[24] H.O. Kreiss. Stability theory for difference approximations of mixed initial boundary value problems, I. *Math. Comp.*, 22:703–714, 1968.

[25] B. Gustafsson, H.O. Kreiss, and A. Sundstrom. Stability theory of difference approximations for mixed initial boundary value problems. II. *Math. Comp.*, 26(119):649–686, 1972.

[26] B. Gustafsson. The convergence rate for difference approximations to mixed initial boundary value problems. *Math. Comp.*, 29(130):396–406, 1975.

[27] R.H. Ni. A multiple grid scheme for solving the Euler equations. *AIAA Journal*, 20:1565–1571, 1982.

[28] A. Jameson. Solution of the Euler equations by a multigrid method. *Applied Math. and Computation*, 13:327–356, 1983.

[29] A. Jameson. Multigrid algorithms for compressible flow calculations. In *Second European Conference on Multigrid Methods*, Cologne, October 1985. Princeton University Report MAE 1743.

[30] J.L. Steger and R.F. Warming. Flux vector splitting of the inviscid gas dynamics equations with applications to finite difference methods. *J. Comp. Phys.*, 40:263–293, 1981.

[31] A. Harten. High resolution schemes for hyperbolic conservation laws. *J. Comp. Phys.*, 49:357–393, 1983.

[32] S.R. Chakravarthy, A. Harten, and S. Osher. Essentially non-oscillatory shock capturing schemes of uniformly very high accuracy. AIAA Paper 86-0339, AIAA 24th Aerospace Sciences Meeting, Reno, January 1986.

[33] M.O. Bristeau, R. Glowinski, J. Periaux, P. Perrier, O. Pironneau, and C. Poirier. On the numerical solution of nonlinear problems in fluid dynamics by least squares and finite element methods (II), application to transonic flow simulations. *Computer Methods in Applied Mechanics and Engineering*, 51:363–394, 1985.

[34] R. Lohner, K. Morgan, J. Peraire, and O.C. Zienkiewicz. Finite element methods for high speed flows. In *Proc. AIAA 7th Computational Fluid Dynamics Conference*, Cincinnati, OH, 1985. AIAA Paper 85-1531.

[35] 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, January 1986.

[36] A. Jameson and T.J. Baker. Improvements to the aircraft Euler method. AIAA Paper 87-0452, AIAA 25th Aerospace Sciences Meeting, Reno, January 1987.

[37] B. Stoufflet, J. Periaux, F. Fezoui, and A. Dervieux. Numerical simulation of 3-D hypersonic Euler flows around space vehicles using adapted finite elements. AIAA paper 87-0560, January 1987.

[38] J.T. Batina. Implicit flux-split Euler schemes for unsteady aerodynamic analysis involving unstructured dynamic meshes. AIAA paper 90-0936, April 1990.

[39] D.J. Mavriplis and A. Jameson. Multigrid solution of the Navier-Stokes equations on triangular meshes. *AIAA Journal*, 28(8):1415–1425, August 1990.

[40] D.J. Mavriplis and L. Martinelli. Multigrid solution of compressible turbulent flow on unstructured meshes using a two-equation model. AIAA Paper 91-0237, January 1991.

[41] W.H. Jou. Boeing Memorandum AERO-B113B-L92-018, September 1992. To Joseph Shang.

[42] M.H. Ha. The impact of turbulence modelling on the numerical prediction of flows. In M. Napolitano and F. Solbetta, editors, *Proc. of the 13th International Conference on Numerical Methods in Fluid Dynamics*, pages 27–46, Rome, Italy, July 1992. Springer Verlag, 1993.

[43] T. Cebeci and A.M.O. Smith. *Analysis of Turbulent Boundary Layers*. Academic Press, 1974.

[44] B. Baldwin and H. Lomax. Thin layer approximation and algebraic model for separated turbulent flow. AIAA Paper 78-257, 1978.

[45] D. Deganiand and L. Schiff. Computation of turbulent supersonic flows around pointed bodies having crossflow separation. *J. Computational Physics*, 66:173–196, 1986.

[46] L. Martinelli, A. Jameson, and E. Malfa. Numerical simulation of three-dimensional vortex flows over delta wing configurations. In M. Napolitano and F. Solbetta, editors, *Proc. 13th International Confrence on Numerical Methods in Fluid Dynamics*, pages 534–538, Rome, Italy, July 1992. Springer Verlag, 1993.

[47] D. Johnson and L. King. A mathematically simple turbulence closure model for attached and separated turbulent boundary layers. *AIAA Journal*, 23:1684–1692, 1985.

[48] C.L. Rumsey and V.N. Vatsa. A comparison of the predictive capabilities of several turbulence models using upwind and centered - difference computer codes. AIAA Paper 93-0192, AIAA 31st Aerospace Sciences Meeting, Reno, January 1993.

[49] T.J. Kao, T.Y. Su, and N.J. Yu. Navier-Stokes calculations for transport wing-body configurations with nacelles and struts. AIAA Paper 93-2945, AIAA 24th Fluid Dynamics Conference, Orlando, July 1993.

[50] W.P. Jones and B.E. Launder. The calculation of low-Reynolds-number phenomena with a two-equation model of turbulence. *International J. of Heat Transfer*, 16:1119–1130, 1973.

[51] D.C. Wilcox. A half a century historical review of the k-ω model. AIAA Paper 91-0615, AIAA 29th Aerospace Sciences Meeting, Reno, NV, January 1991.

[52] C.G. Speziale, E.C. Anderson, and R. Abid. A critical evaluation of two-equation models for near wall turbulence. AIAA Paper 90-1481, June 1990. ICASE Report 90-46.

[53] R. Abid, C.G. Speziale, and S. Thangam. Application of a new k-τ model to near wall turbulent flows. AIAA Paper 91-0614, AIAA 29th Aerospace Sciences Meeting, Reno, NV, January 1991.

[54] F. Menter. Zonal two-equation k-ω turbulence models for aerodynamic flows. AIAA Paper 93-2906, AIAA 24th Fluid Dynamics Meeting, Orlando, July 1993.

[55] T.J. Coakley. Numerical simulation of viscous transonic airfoil flows. AIAA Paper 87-0416, AIAA 25th Aerospace Sciences Meeting, Reno, January 1987.

[56] B.S. Baldwin and T.J. Barth. A one-equation turbulence transport model for high Reynolds number wall-bounded flows. AIAA Paper 91-0610, AIAA 29th Aerospace Sciences Meeting, Reno, NV, January 1991.

[57] P. Spalart and S. Allmaras. A one-equation turbulent model for aerodynamic flows. AIAA Paper 92-0439, AIAA 30th Aerospace Sciences Meeting, Reno, NV, January 1992.

[58] V. Yakhot and S.A. Orszag. Renormalization group analysis of turbulence. I. Basic theory. *J. Scientific Computing*, 1:3–51, 1986.

[59] L.M. Smith and W.C. Reynolds. On the Yakhot-Orszag renormalization group for deriving turbulence statistics and models. *Physics of Fluids A*, 4:364–390, 1992.

[60] L. Martinelli and V. Yakhot. RNG-based turbulence transport approximations with applications to transonic flows. AIAA Paper 89-1950, AIAA 9th Computational Fluid Dynamics Conference, Buffalo, NY, June 1989.

[61] J.E. Melton, S.A. Pandya, and J.L. Steger. 3D Euler flow solutions using unstructured Cartesian and prismatic grids. AIAA Paper 93-0331, Reno, NV, January 1993.

[62] S.S. Samant, J.E. Bussoletti, F.T. Johnson, R.H. Burkhart, B.L. Everson, R.G. Melvin, D.P. Young, L.L. Erickson, and M.D. Madson. TRANAIR: A computer code for transonic analyses of arbitrary configurations. AIAA Paper 87-0034, 1987.

[63] M. Berger and R.J. LeVeque. An adaptive Cartesian mesh algorithm for the Euler equations in arbitrary geometries. AIAA Paper 89-1930, 1989.

[64] A.M. Landsberg, J.P. Boris, W. Sandberg, and T.R. Young. Naval ship superstructure design: Complex three-dimensional flows using an efficient, parallel method. *High Performance Computing 1993: Grand Challenges in Computer Simulation*, 1993.

[65] T.J. Baker. Mesh generation by a sequence of transformations. *Appl. Numer. Math.*, 2:515–528, 1986.

[66] P.R. Eiseman. A multi-surface method of coordinate generation. *J. Comp. Physics*, 33:118–150, 1979.

[67] L.E. Eriksson. Generation of boundary-conforming grids around wing-body configurations using transfinite interpolation. *AIAA Journal*, 20:1313–1320, 1982.

[68] R.E. Smith. Three-dimensional algebraic mesh generation. In *Proc. AIAA 6th Computational Fluid Dynamics Conference*, Danvers, MA, 1983. AIAA Paper 83-1904.

[69] J.F. Thompson, F.C. Thames, and C.W. Mastin. Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies. *J. Comp. Physics*, 15:299–319, 1974.

[70] J.F. Thompson, Z.U.A. Warsi, and C.W. Mastin. Boundary-fitted coordinate systems for numerical solution of partial differential equations: A review. *J. Comp. Phys.*, 47:1–108, 1982.

[71] R.L. Sorenson. Elliptic generation of compressible three-dimensional grids about realistic aircraft. In J. Hauser and C. Taylor, editors, *International Conference on Numerical Grid Generation in Computational Fluid Dynamics*, Landshut, F. R. G., 1986.

[72] R.L. Sorenson. Three-dimensional elliptic grid generation for an F-16. In J.L. Steger and J.F. Thompson, editors, *Three-Dimensional Grid Generation for Complex Configurations: Recent Progress*, 1988. AGARDograph.

[73] J.L. Steger and D.S. Chaussee. Generation of body-fitted coordinates using hyperbolic partial differential equations. *SIAM J. Scientific and Statistical Computing*, 1:431–437, 1980.

[74] N.P. Weatherill and C.A. Forsey. Grid generation and flow calculations for aircraft geometries. *J. Aircraft*, 22:855–860, 1985.

[75] K. Sawada and S. Takanashi. A numerical investigation on wing/nacelle interferences of USB configuration. In *Proceedings AIAA 25th Aerospace Sciences Meeting*, Reno, NV, 1987. AIAA paper 87-0455.

[76] J.A. Benek, P.G. Buning, and J.L. Steger. A 3-D Chimera grid embedding technique. In *Proceedings AIAA 7th Computational Fluid Dynamics Conference*, pages 507–512, Cincinnati, OH, 1985. AIAA Paper 85-1523.

[77] J.A. Benek, T.L. Donegan, and N.E. Suhs. Extended Chimera grid embedding scheme with applications to viscous flows. AIAA Paper 87-1126, AIAA 8th Computational Fluid Dynamics Conference, Honolulu, HI, 1987.

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

[79] G. Voronoi. Nouvelles applications des parametres continus a la theorie des formes quadratiques. Deuxieme memoire: Recherches sur les parallelloedres primitifs. *J. Reine Angew. Math.*, 134:198–287, 1908.

[80] R. Lohner and P. Parikh. Generation of three-dimensional unstructured grids by the advancing front method. AIAA Paper 88-0515, Reno, NV, January 1988.

[81] R.W. MacCormack and A.J. Paullay. Computational efficiency achieved by time splitting of finite difference operators. AIAA Paper 72-154, 1972.

[82] M.G. Hall. Cell vertex multigrid schemes for solution of the Euler equations. In *Proc. IMA Conference on Numerical Methods for Fluid Dynamics*, Reading, April 1985.

[83] A. Jameson. A vertex based multigrid algorithm for three-dimensional compressible flow calculations. In T.E. Tezduar and T.J.R. Hughes, editors, *Numerical Methods for Compressible Flow - Finite Difference, Element And Volume Techniques*, 1986. ASME Publication AMD 78.

[84] R. Radespiel, C. Rossow, and R.C. Swanson. An efficient cell-vertex multigrid scheme for the three-dimensional Navier-Stokes equations. In *Proc. AIAA 9th Computational Fluid Dynamics Conference*, pages 249–260, Buffalo, NY, June 1989. AIAA Paper 89-1953-CP.

[85] P.D. Lax and B. Wendroff. Systems of conservation laws. *Comm. Pure. Appl. Math.*, 13:217–237, 1960.

[86] P.D. Lax. Hyperbolic systems of conservation laws and the mathematical theory of shock waves. *SIAM Regional Series on Applied Mathematics*, 11, 1973.

[87] S.K. Godunov. A difference method for the numerical calculation of discontinuous solutions of hydrodynamic equations. *Mat. Sbornik*, 47:271–306, 1959. Translated as JPRS 7225 by U.S. Dept. of Commerce, 1960.

[88] J.P. Boris and D.L. Book. Flux corrected transport, 1, SHASTA, a fluid transport algorithm that works. *J. Computational Physics*, 11:38–69, 1973.

[89] B. Van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second order scheme. *J. Computational Physics*, 14:361–370, 1974.

[90] B. Van Leer. Flux vector splitting for the Euler equations. In E. Krause, editor, *Proc. 8th International Conference on Numerical Methods in Fluid Dynamics*, pages 507–512, Aachen, 1982. Springer, 1982.

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

[92] S. Osher and F. Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Math. Comp.*, 38:339–374, 1982.

[93] S. Osher and S. Chakravarthy. High resolution schemes and the entropy condition. *SIAM J. Num Analysis*, 21:955–984, 1984.

[94] P.K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Num. Analysis*, 21:995–1011, 1984.

[95] B.K. Anderson, J.L. Thomas, and B. Van Leer. A comparison of flux vector splittings for the Euler equations. AIAA Paper 85-0122, Reno, NV, January 1985.

[96] A. Jameson. Nonoscillatory shock capturing scheme using flux limited dissipation. In B.E. Engquist, S. Osher, and R.C.J. Sommerville, editors, *Lectures in Applied Mathematics, Vol. 22, Part 1, Large Scale Computations in Fluid Mechanics*, pages 345–370, 1985.

[97] H.C. Yee. On symmetric and upwind TVD schemes. In *Proc. 6th GAMM Conference on Numerical Methods in Fluid Mechanics*, Gottingen, September 1985.

[98] 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. *Computer Methods in Applied Mechanics and Engineering*, 59:223–231, 1986.

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

[100] S.T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Computational Physics*, 31:335–362, 1979.

[101] A. Jameson. Artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence in transonic and hypersonic flows. AIAA Paper 93-3359, AIAA 11th Computational Fluid Dynamics Conference, Orlando, FL, July 1993.

[102] C. Hirsch, C. Lacol, and H. Deconinck. Convection algorithms based on a diagonalization procedure for the multi-dimensional Euler equations. In *Proc. AIAA 8th Computational Fluid Dynamics Conference*, pages 667–676, Hawaii, June 1987. AIAA Paper 87-1163.

[103] K.G. Powell and B. van Leer. A genuinely multidimensional upwind cell-vertex scheme for the Euler equations. AIAA Paper 89-0095, AIAA 27th Aerospace Sciences Meeting, Reno, January 1989.

[104] B. Van Leer. Progress in multi-dimensional upwind differencing. In M. Napolitano and F. Solbetta, editors, *Proc. 13th International Conference on Numerical Methods in Fluid Dynamics*, pages 1–26, Rome, July 1992. Springer Verlag, 1993.

[105] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984.

[106] C.W. Shu, T.A. Zang, G. Erlebacher, D. Whitaker, and S. Osher. High-order ENO schemes applied to two- and three-dimensional compressible flow. *Applied Numerical Mathematics*, 9:45–71, 1992.

[107] C.W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[108] C.W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes II. *J. Comput. Phys.*, 83:32–78, 1989.

[109] T.J. Barth and D.C. Jespersen. The design and application of upwind schemes on unstructured meshes. AIAA paper 89-0366, January 1989.

[110] T.J. Barth and P.O. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. AIAA paper 90-0013, January 1990.

[111] S.M. Deshpande. On the Maxwellian distribution, symmetric form and entropy conservation for the Euler equations. NASA TP 2583, 1986.

[112] J.P. Croisille and P. Villedieu. Kinetic flux splitting schemes for hypersonic flows. In M. Napolitano and F. Sobetta, editors, *Proc 13th International Congress on Numerical Methods in Fluid Dynamics*, pages 310–313, Rome, July 1992. Springer Verlag.

[113] W. Eppard and B. Grossman. A multidimensional kinetic-based upwind solver for the Euler equations. AIAA Paper 93-3303, AIAA 11th Computational Fluid Dynamics Conference, Orlando, FL, July 1993.

[114] K. Xu and K. Prendergast. Numerical hydrodynamics from gas kinetic theory. To appear in Journal Computational Physics.

[115] C. Mitchell and R. Walters. k-exact reconstruction for the Navier-Stokes equations on arbitrary grids. AIAA Paper 93-0536, AIAA 31st Aerospace Sciences Meeting, Reno, NV, January 1993.

[116] G. Strang and G. Fix. *Analysis of the Finite Element Method*. Prentice Hall, 1993.

[117] V. Venkatakrishnan. Newton solution of inviscid and viscous problems. AIAA paper 88-0413, January 1988.

[118] M. Giles, M. Drela, and W.T. Thompkins. Newton solution of direct and inverse transonic Euler equations. AIAA Paper 85-1530, Cincinnati, 1985.

[119] R. Lohner and D. Martin. An implicit linelet-based solver for incompressible flows. AIAA paper 92-0668, AIAA 30th Aerospace Sciences Meeting, Reno, NV, January 1992.

[120] A. Jameson and E. Turkel. Implicit schemes and LU decompositions. *Math. Comp.*, 37:385–397, 1981.

[121] S. Obayashi and K. Kuwakara. LU factorization of an implicit scheme for the compressible Navier-Stokes equations. AIAA Paper 84-1670, AIAA 17th Fluid Dynamics and Plasma Dynamics Conference, Snowmass, June 1984.

[122] S.R. Chakravarthy. Relaxation methods for unfactored implicit upwind schemes. AIAA Paper 84-0165, AIAA 22nd Aerospace Sciences Meeting, Reno, January 1984.

[123] S. Yoon and A. Jameson. Lower-upper Symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations. AIAA Paper 87-0600, AIAA 25th Aerospace Sciences Meeting, Reno, January 1987.

[124] R. Chipman and A. Jameson. Fully conservative numerical solutions for unsteady irrotational transonic flow about airfoils. AIAA Paper 79-1555, AIAA 12th Fluid and Plasma Dynamics Conference, Williamsburg, VA, July 1979.

[125] A. Jameson. Transonic flow calculations for aircraft. In F. Brezzi, editor, *Lecture Notes in Mathematics, Numerical Methods in Fluid Dynamics*, pages 156–242. Springer Verlag, 1985.

[126] A. Rizzi and L.E. Eriksson. Computation of flow around wings based on the Euler equations. *J. Fluid Mech.*, 148:45–71, 1984.

[127] D.A. Caughey. A diagonal implicit multigrid algorithm for the Euler equations. AIAA Paper 87-453, 25th Aerospace Sciences Meeting, Reno, January 1987.

[128] W.K. Anderson, J.L. Thomas, and D.L. Whitfield. Multigrid acceleration of the flux split Euler equations. AIAA Paper 86-0274, AIAA 24th Aerospace Sciences Meeting, Reno, January 1986.

[129] P.W. Hemker and S.P. Spekreijse. Multigrid solution of the steady Euler equations. In *Proc. Oberwolfach Meeting on Multigrid Methods*, December 1984.

[130] A. Jameson and D.J. Mavriplis. Multigrid solution of the Euler equations on unstructured and adaptive grids. In S. McCormick, editor, *Multigrid Methods, Theory, Applications and Supercomputing. Lecture Notes in Pure and Applied Mathematics*, volume 110, pages 413–430, April 1987.

[131] M.H. Lallemand and A. Devrieux. A multigrid finite-element method for solving the two-dimensional Euler equations. In S.F. McCormick, editor, *Proceedings of the Third Copper Mountain Conference on Multigrid Methods, Lecture Notes in Pure and Applied Mathematics*, pages 337–363, Copper Mountain, April 1987.

[132] M.M. Rai and P. Moin. Direct numerical simulation of transition and turbulence in a spatially evolving boundary layer. AIAA Paper 91-1607 CP, AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, June 1991.

[133] S.K. Lele. Compact finite difference schemes with spectral-like resolution. CTR Manuscript 107, 1990.

[134] M.H. Carpenter, D. Gottlieb, and S. Abarbanel. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes. ICASE Report 93-9, Hampton, VA, March 1993.

[135] S. Orszag and D. Gottlieb. Numerical analysis of spectral methods. *SIAM Regional Series on Applied Mathematics*, 26, 1977.

[136] C. Canuto, M.Y. Hussaini, A. Quarteroni, and D.A. Zang. *Spectral Methods in Fluid Dynamics*. Springer Verlag, 1987.

[137] G.E. Karniadakis and S.A. Orszag. Nodes, modes and flow codes. *Physics Today*, pages 34–42, March 1993.

[138] M. Berger and A. Jameson. Automatic adaptive grid refinement for the Euler equations. *AIAA Journal*, 23:561–568, 1985.

[139] J.F. Dannenhoffer and J.R. Baron. Robust grid adaptation for complex transonic flows. AIAA Paper 86-0495, AIAA 24th Aerospace Sciences Meeting, Reno, January 1986.

[140] 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, January 1986.

[141] D.G. Holmes and S.H. Lamson. Adaptive triangular meshes for compressible flow solutions. In *Proceedings First International Conference on Numerical Grid Generation in Computational Fluid Dynamics*, pages 413–424, Landshut, FRG, July 1986.

[142] J.J. Quirk. An alternative to unstructured grids for computing gas dynamics flows about arbitrarily complex two-dimensional bodies. ICASE Report 92-7, Hampton, VA, February 1992.

[143] D. Lefebre, J.Peraire, and K.Morgan. Finite element least squares solutions of the Euler equations using linear and quadratic approximations. *International Journal of Computational Fluid Dynamics*, 1:1–23, 1993.

[144] J.T. Oden, L. Demkowicz, T. Liszka, and W. Rachowicz. h-p adaptive finite element methods for compressible and incompressible flows. In S. L. Venneri A. K. Noor, editor, *Proceedings of the Symposium on Computational Technology on Flight Vehicles*, pages 523–534, Washington, D.C., November 1990. Pergamon.

[145] R.J. Busch Jr. Computational fluid dynamics in the design of the Northrop McDonnell Douglas YF-23 ATF prototype. AIAA Paper 91-1627, AIAA 21st Fluid Dynamics, Plasma Dynamics and Laser Conference, Honolulu, HI, June 1991.

[146] S.E. Cliff and S.D. Thomas. Euler/experimental correlations of sonic boom pressure signatures. AIAA Paper 91-3276, AIAA 9th Applied Aerodynamics Conference, Baltimore, September 1991.

[147] D.J. Mavriplis. Unstructured mesh algorithms for aerodynamic calculations. In M. Napolitano and F. Solbetta, editors, *Proc. 13th International Conference on Numerical Methods in Fluid Dynamics*, pages 57–77, Rome, July 1992. Springer Verlag, 1993.

[148] J.D. Baum and R. Lohner. Numerical simulation of a pilot/seat ejection from an F-16. AIAA Paper 93-0783, AIAA 31st Aerospace Sciences Meeting, Reno, January 1993.

[149] M.Y. Hussaini. Computational fluid dynamics - a personal view. In D.L. Dwoyer, M.Y. Hussaini, and R.G. Voigt, editors, *Proc. 11th International Conference on Numerical Methods in Fluid Dynamics*, pages 3–17, Williamsburg, June 1988. Springer Verlag, 1989.

[150] R.M. Cummings, Y.M. Rizk, L.B. Schniff, and N.M. Chaderjian. Navier-Stokes predictions for the F-18 wing and fuselage at large incidence. *J. of Aircraft*, 29:565–574, 1992.

[151] F. Ghaffari, B.L. Bates, J.M. Luckring, J.L.Thomas, and R.T. Biedron. Navier-Stokes solutions about the F/A-18 wing-lex-fuselage configuration with multi-block structured grids. AIAA Paper 91-3291, AIAA 9th Applied Aerodynamics Conference, Baltimore, September 1991.

[152] D.G. Pearce, S.A. Stanley, F.W. Martin Jr., R.J. Gomez, G.J. Le Beau, P.G. Buning, W.M. Chan, I. Chiu, A. Wulf, and V. Akdag. Development of a large scale Chimera grid system for the space shuttle launch vehicle. AIAA Paper 93-0533, 31st AIAA Aerospace Sciences Meeting, Reno, NV, January 1993.

[153] S.E. Rogers. Progress in high-lift aerodynamic calculations. AIAA Paper 93-0194, AIAA 31st Aerospace Science Meeting, Reno, January 1993.

[154] M.J. Siclari, G. Carpenter, and R. Ende. Navier-Stokes computations for a magnetically levitated vehicle (MAGLEV) in ground effect. AIAA Paper 93-2950, AIAA 24th Fluid Dynamics Conference, Orlando, July 1993.

[155] C. Kiris, L. Chang, D. Kwak, and S. Rogers. Incompressible Navier-Stokes computations of rotating flows. AIAA Paper 93-0678, AIAA 31st Aerospace Sciences Meeting, Reno, January 1993.

[156] S.L. Lawrence. Numerical performance estimates for a generic hypersonic forebody. AIAA Paper 91-1695, AIAA 22nd Fluid Dynamics, Plasma Dynamics and Lasers Conference, Honolulu, HI, June 1991.

[157] K. Dortmann, S. Leicher, and H. Rieger. Computation for HERMES alternate methods programme. Technical report.

[158] J. Periaux, M. Mallet, R. Abgroll, J. Desideri, P. Perrier, and B. Stoufflet. The European hypersonic data base: A new CFD validation tool for the design of space vehicles. AIAA Paper 93-3045, AIAA 24th Fluid Dynamics Conference, Orlando, July 1993.

[159] R.M. Hicks and P.A. Henne. Wing design by numerical optimization. AIAA Paper 79-0080, 1979.

[160] N.J. Yu. Transonic wing analysis and design using Navier-Stokes equations. In M. Napolitano and F. Solbetta, editors, *Proc. 13th International Conference on Numerical Methods in Fluid Dynamics*, pages 504–508, Rome, Italy, July 1992. Springer Verlag, 1993.

[161] P.A. Henne. An inverse transonic wing design method. AIAA Paper 80-0330, 1980.

[162] A. Jameson. Aerodynamic design via control theory. *J. Scientific Computing*, 3:233–260, 1988.

[163] A. Jameson. Computational algorithms for aerodynamic analysis and design. MAE Report 1966, Princeton University, Princeton, NJ, December 1992. To appear in J. Appl. Num. Math.

25
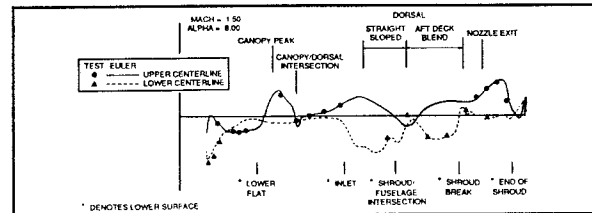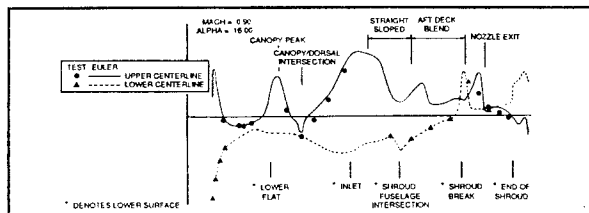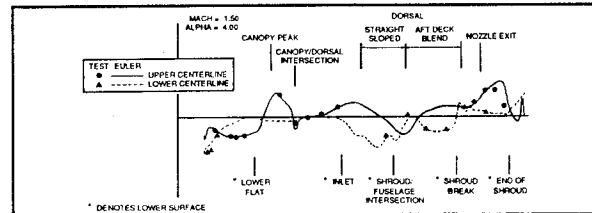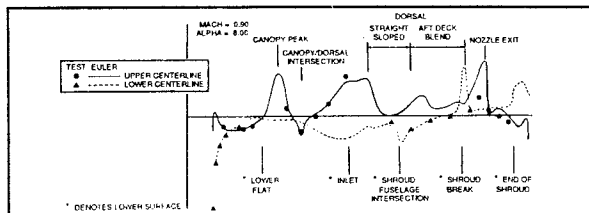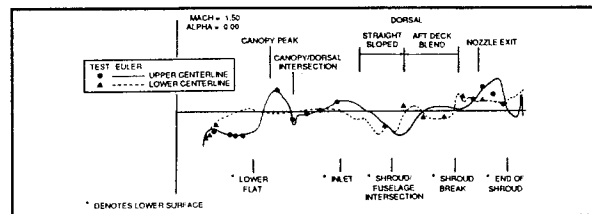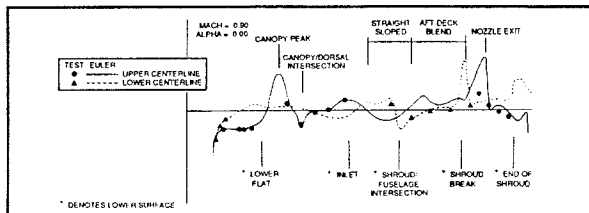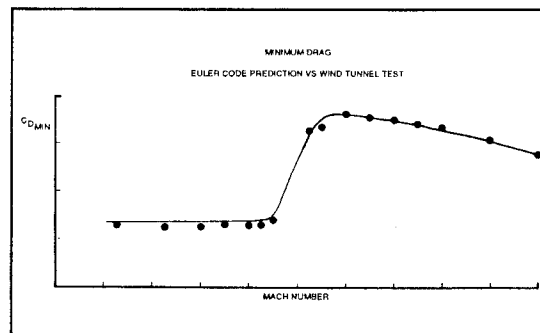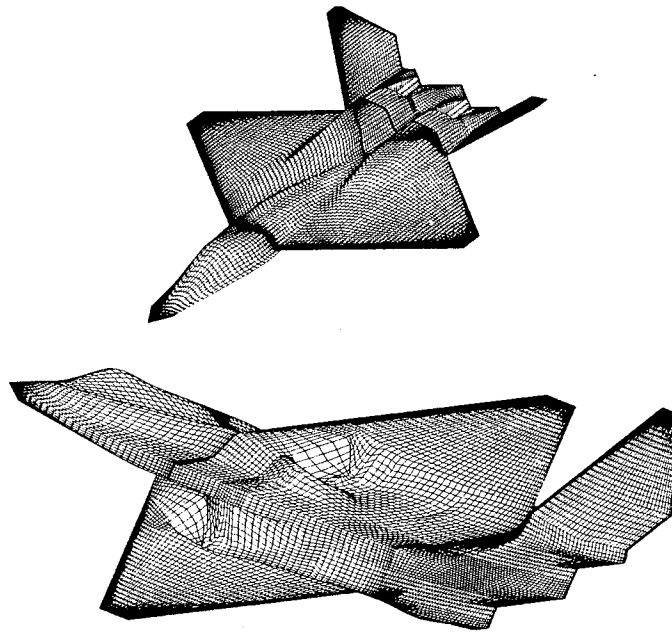
Plate 1: Northrop YF-23.
Supplied by R.J. Busch, Jr.

2b: McDonnell Douglas MD11.
Mach 0.825, $\alpha = 2.5°$.

2a: U.S. Space Shuttle.
Mach 1.25, $\alpha = -5.1°$.

Plate 2: *AIRPLANE Code (A. Jameson and T.J. Baker).*
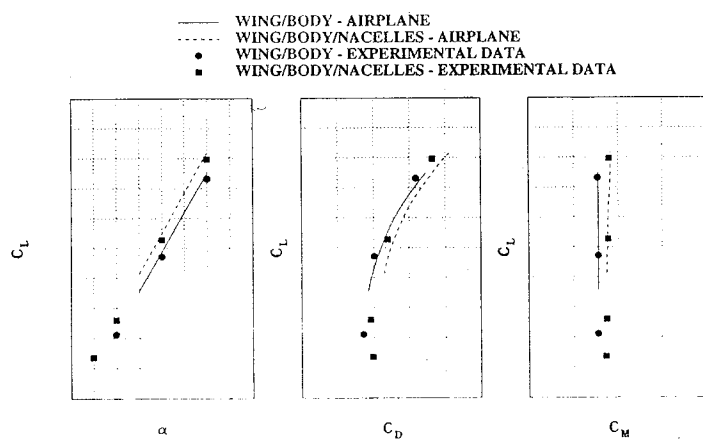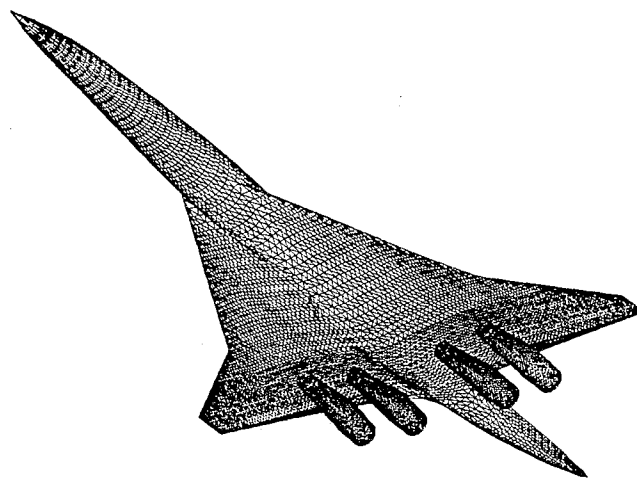
3a: Force Coefficients, Mach 2.1.
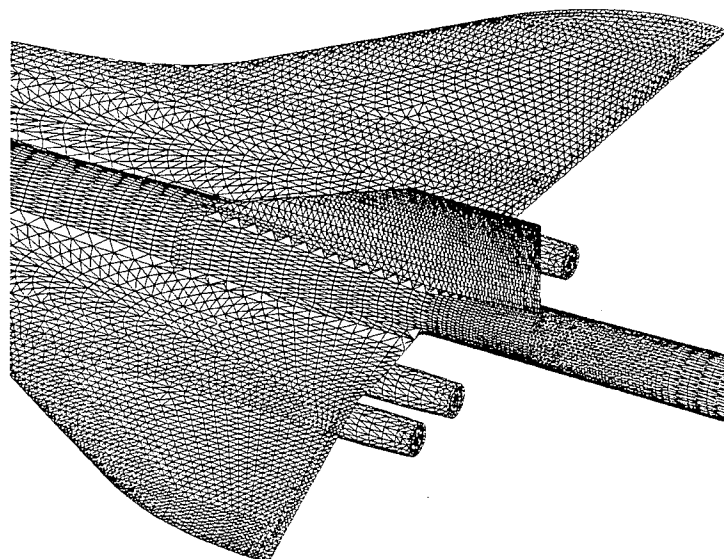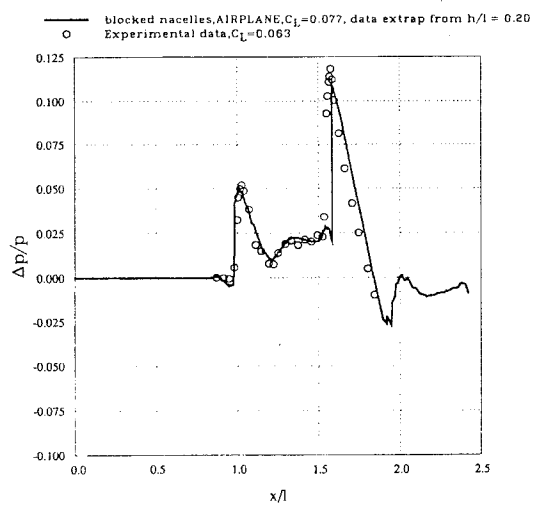
3b: Sonic Boom Prediction, Mach 2.5.

Plate 3: Supersonic Transport Calculations.
AIRPLANE code (A. Jameson and T.J. Baker).

Mach Contours

	1.88
	1.71
	1.55
	1.38
	1.22
	1.05
	0.88
	0.72
	0.55
	0.39
	0.22
	0.06

Single Grid

V-Cycle

W-Cycle

Log (Error)

Cycles
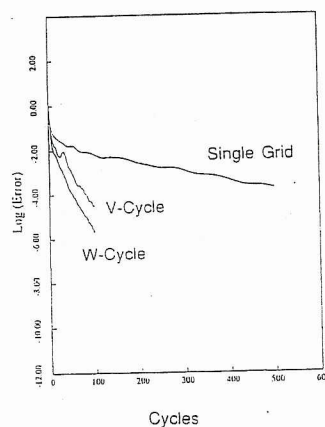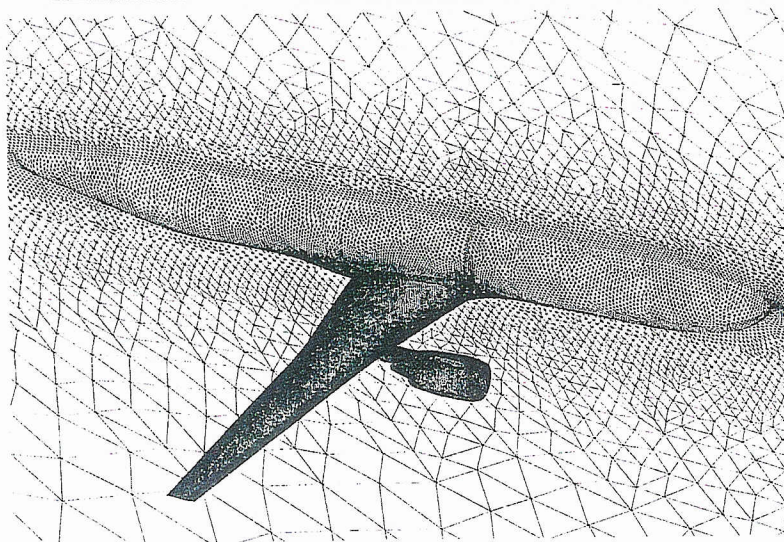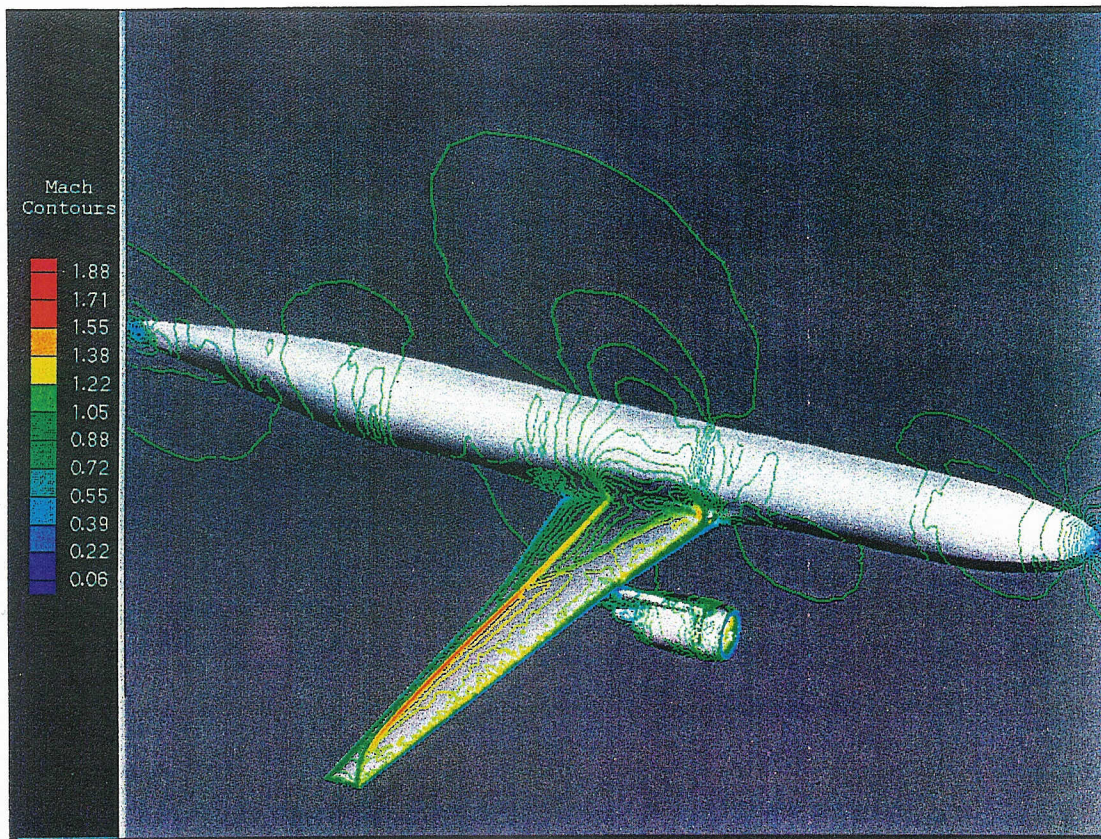
— 800,000 Grid Points

— 4.5 Million Cells

— 96 Million Words

— Time to Solution

○ 2 hours on CRAY YMP-1 (100 Mflops)

○ 16 minutes on CRAY-YMP-8 (750 Mflops)

○ 4 minutes on CRAY-YMP-C90-16 (3.2 Gflops)

○ 10 minutes on Intel Touchstone Delta (1.2 Gflops)

Plate 4: Multigrid Calculations of a Wing-Body-Nacelle Configuration.
Supplied by D.J. Mavriplis.
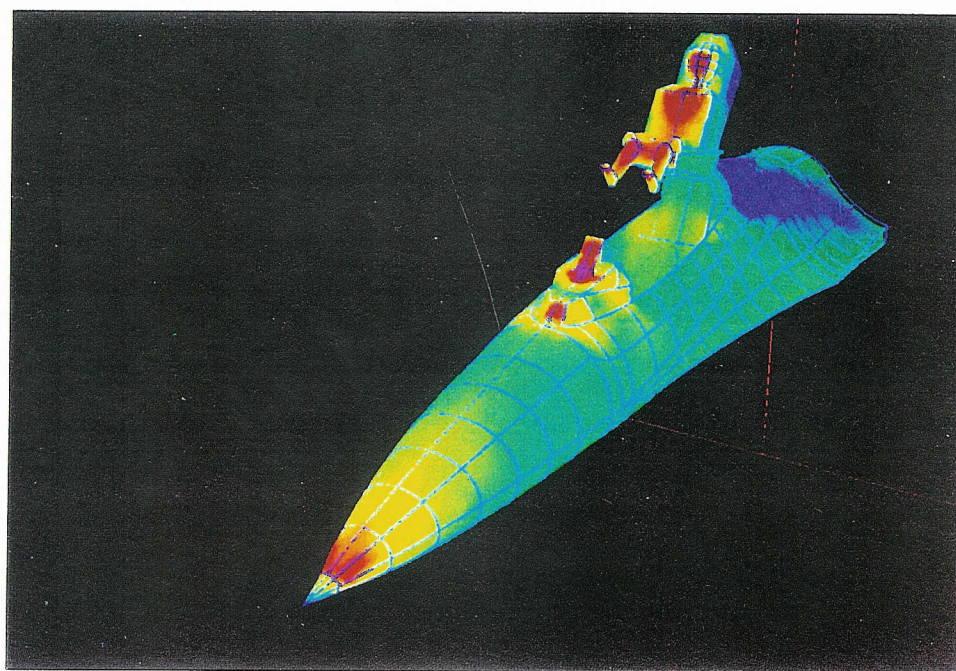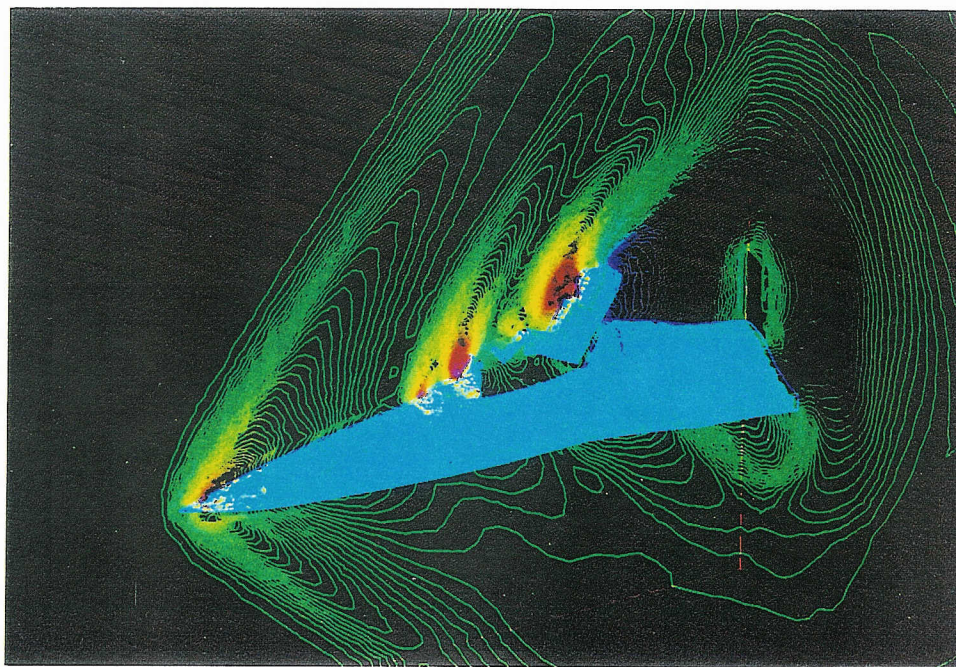
Plate 5: F-16 Seat Ejection at Mach 1.2.
Supplied by R. Löhner.

LAMINAR BREAKDOWN

EXPERIMENT	DIRECT SIMULATION

$$Z=\lambda_Z/64$$
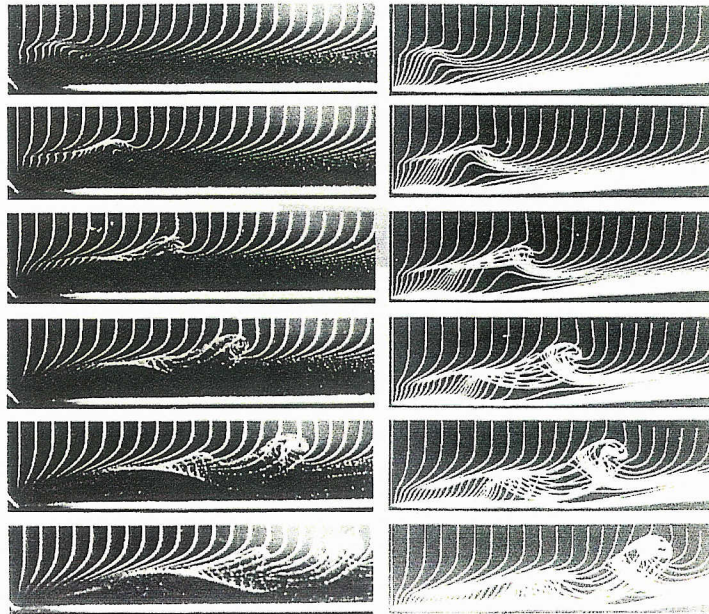
6a: Spectral Simulation of Incompressible Flow.
Supplied by M.Y. Hussaini.
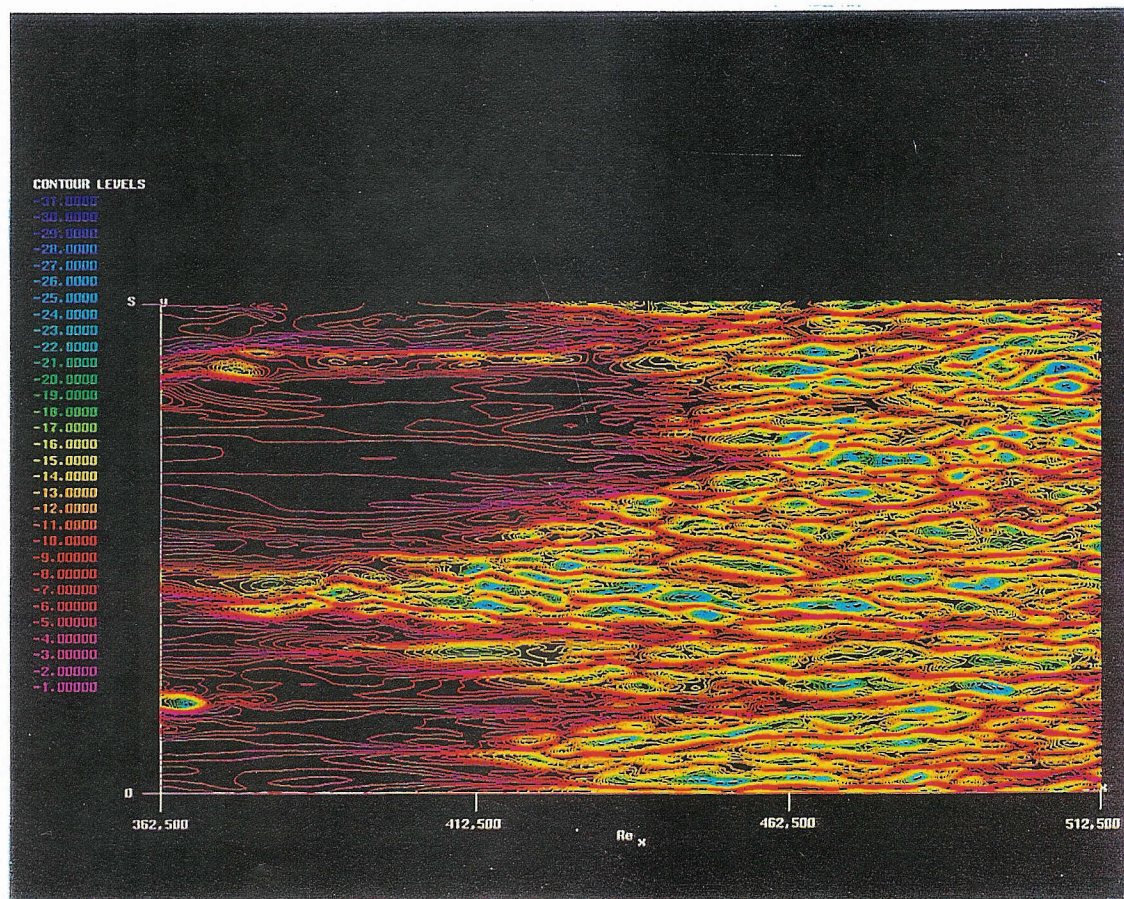
6b: Finite Difference Simulation at Mach 0.1, Reynolds Number 50000/Inch.
Supplied by M.M. Rai.

Plate 6: Direct Navier-Stokes Simulations of Transition to Turbulence.

FLIGHT   COMPUTATION

OFF-SURFACE INSTANTANEOUS STREAMLINES
FLOW WITH TANGENTIAL SLOT BLOWING
$M_\infty = 0.243$, $\alpha = 30.3°$, $Re_{\bar{c}} = 11.0 \times 10^6$, $\beta = 0°$

SLOT

NO BLOWING   BLOWING

HIGH ALPHA
NASA AMES
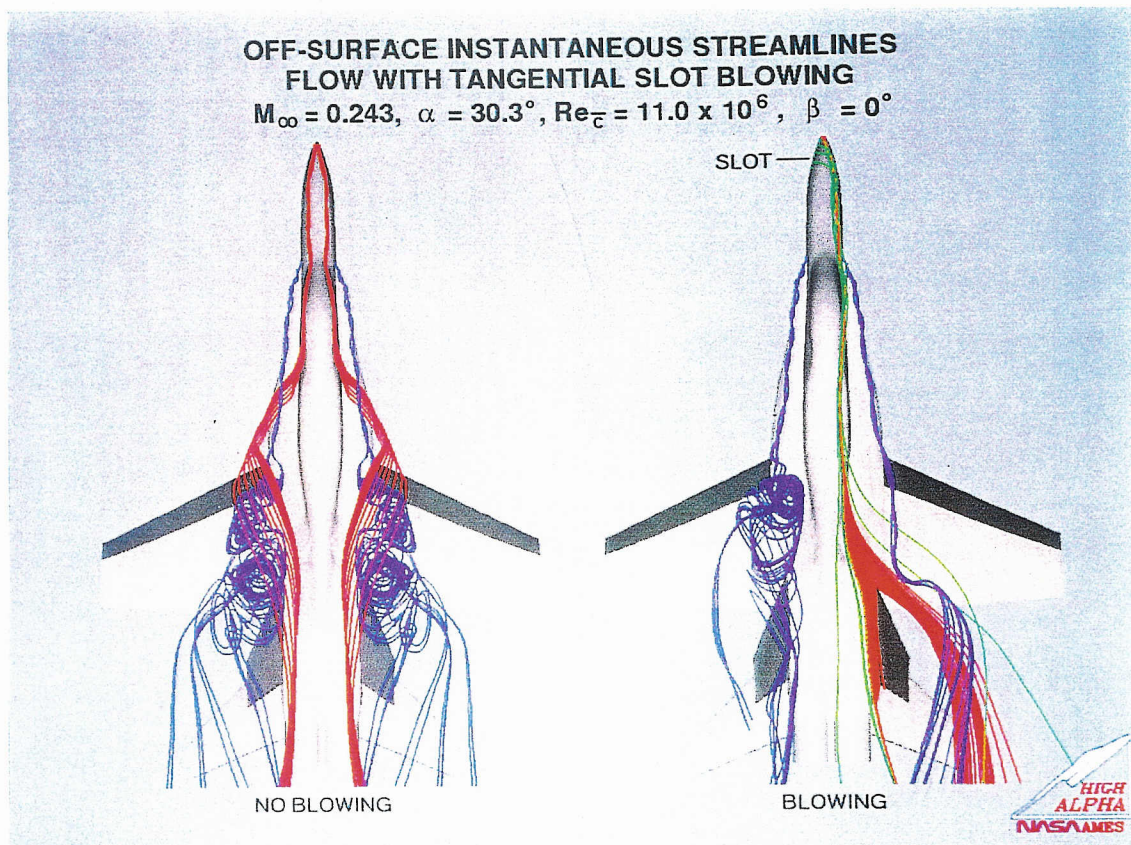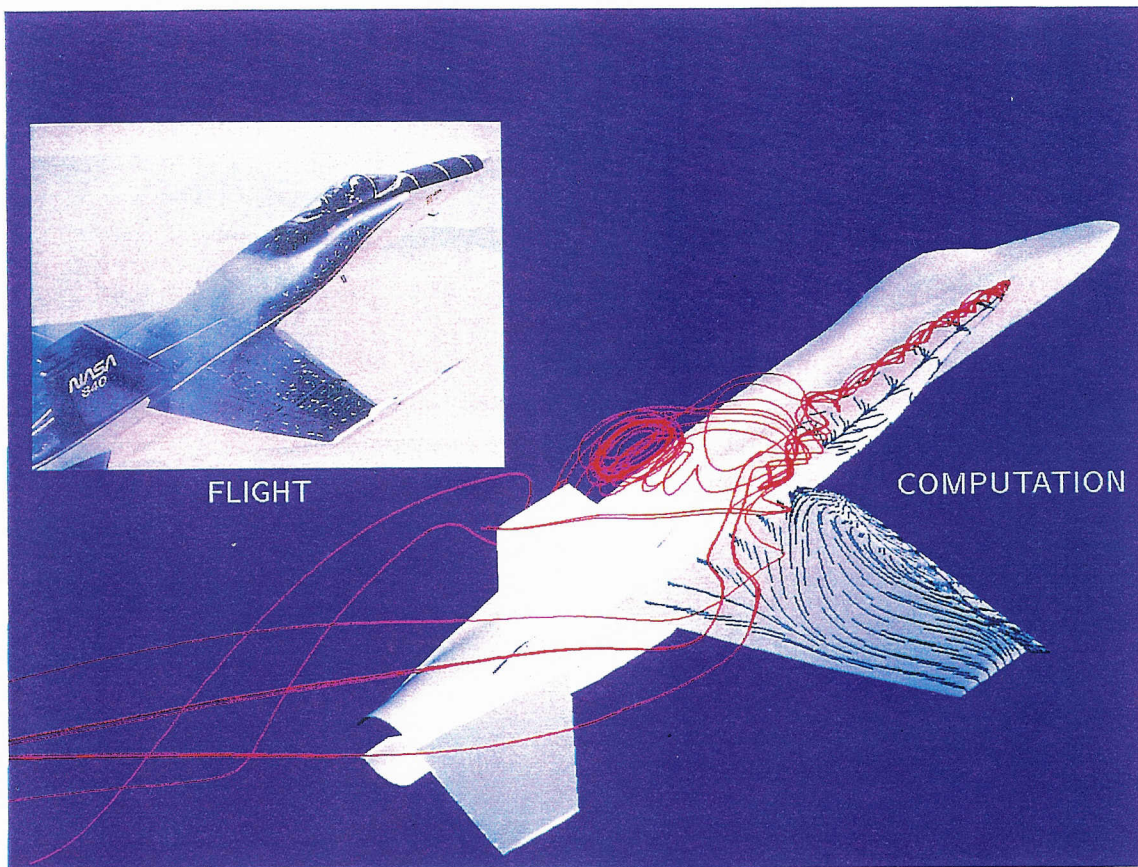
Plate 7: Navier-Stokes Predictions for the F-18 Wing-Fuselage at Large Incidence .
Supplied by R.M. Cummings, Y.M. Rizk, L.B. Schiff, and N.M. Chaderjian.

Unrestricted Surface Flow Pattern

$M_\infty = 0.34$, $R_{\bar{c}} = 13.5 \times 10^6$, Alpha = $19°$

Undeflected Flap

Blended Flap

LaRC
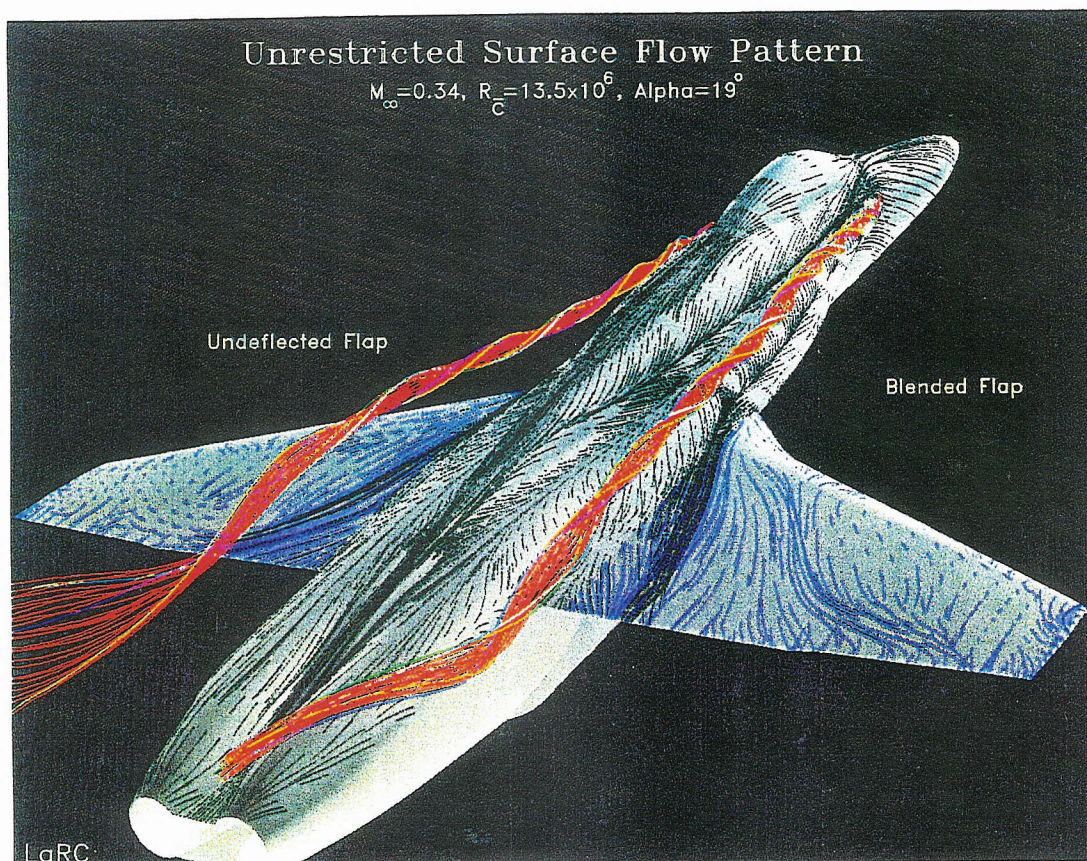
F/A-18 COMPUTED AND FLIGHT-TEST PRESSURES

$\alpha = 30°$; $Re_{\bar{c}} = 10 \times 10^6$

o Flight-test HARV
— Navier-Stokes CFL3D

FS 253   FS 296   FS 357

FS 357

FS 253

FS 296

Port
Section A-A

Starboard
Section C-C
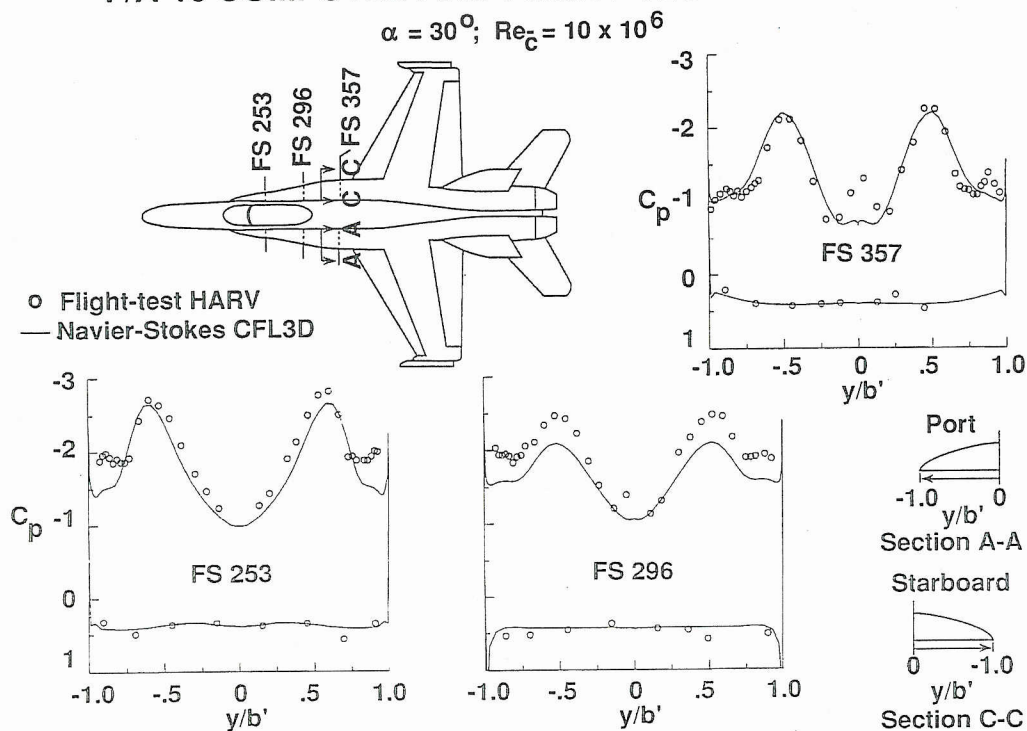
Plate 8:   Navier-Stokes Solutions about the F-18 Wing-LEX-Fuselage Configuration.
Supplied by F. Ghaffari, B.L. Bates, J.M. Luckring, J.L. Thomas, and R.T. Biedron.

a) 2y/B = 0.65

b) 2y/B = 0.90
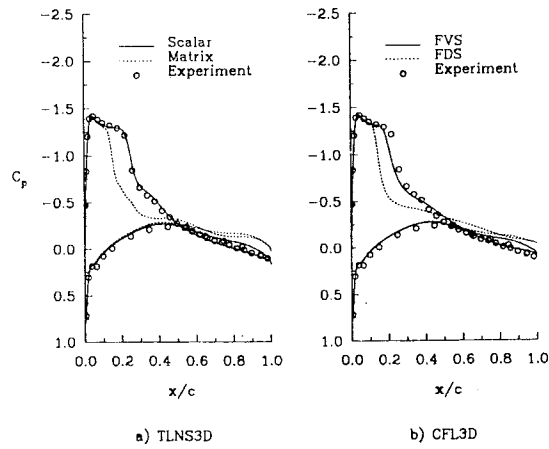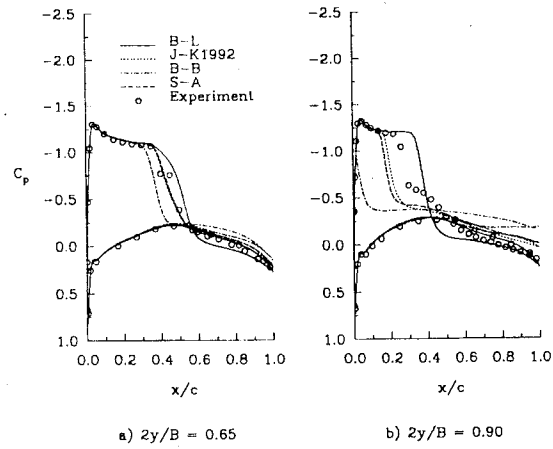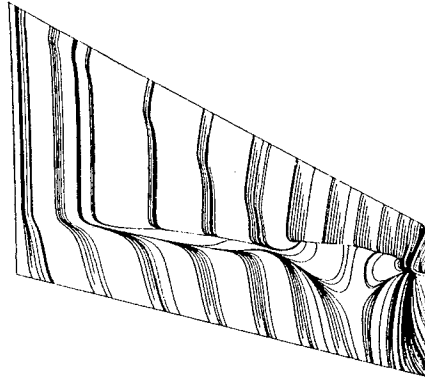
a) TLNS3D

b) CFL3D

Plate 9:   Navier-Stokes Solutions for the ONERA M6 Wing using TLNS3D and
CFL3D with Different Turbulence Models.
Supplied by C.L. Rumsey and V.N. Vatsa.
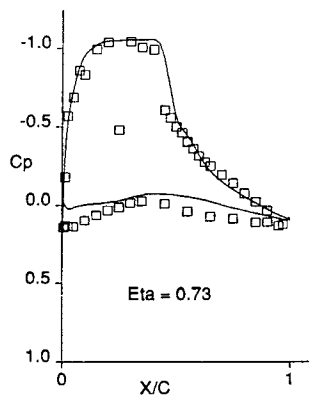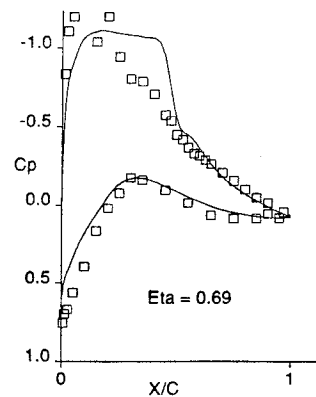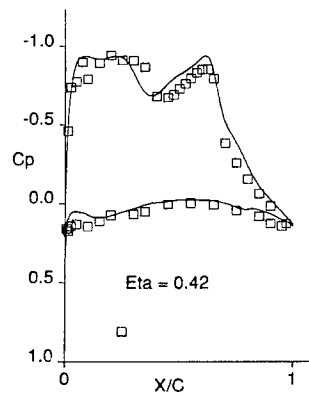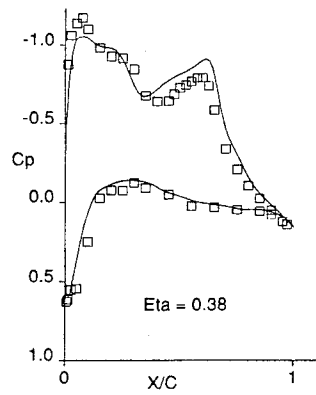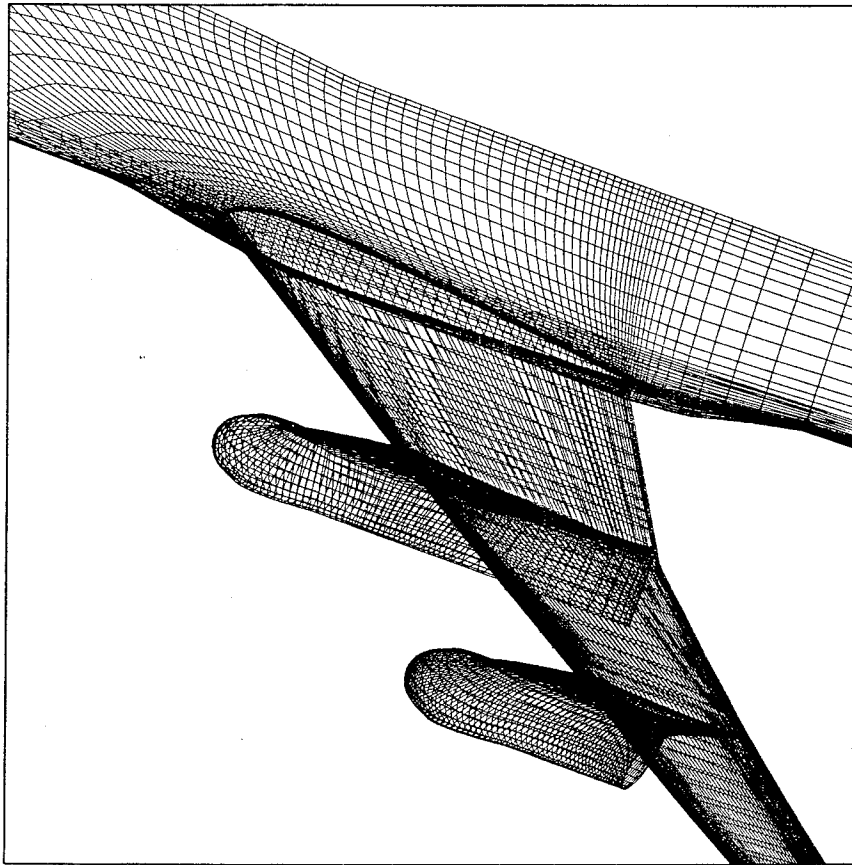
Plate 10: Navier-Stokes Solutions for a Boeing 747/200 Wing-Body Configuration with Engines.
Supplied by J. Yu.

121 x 71 x 41	GRID 1
61 x 23 x 41	GRID 2
31 x 29 x 21	GRID 3
81 x 41 x 31	GRID 4
41 x 21 x 31	GRID 5
41 x 21 x 31	GRID
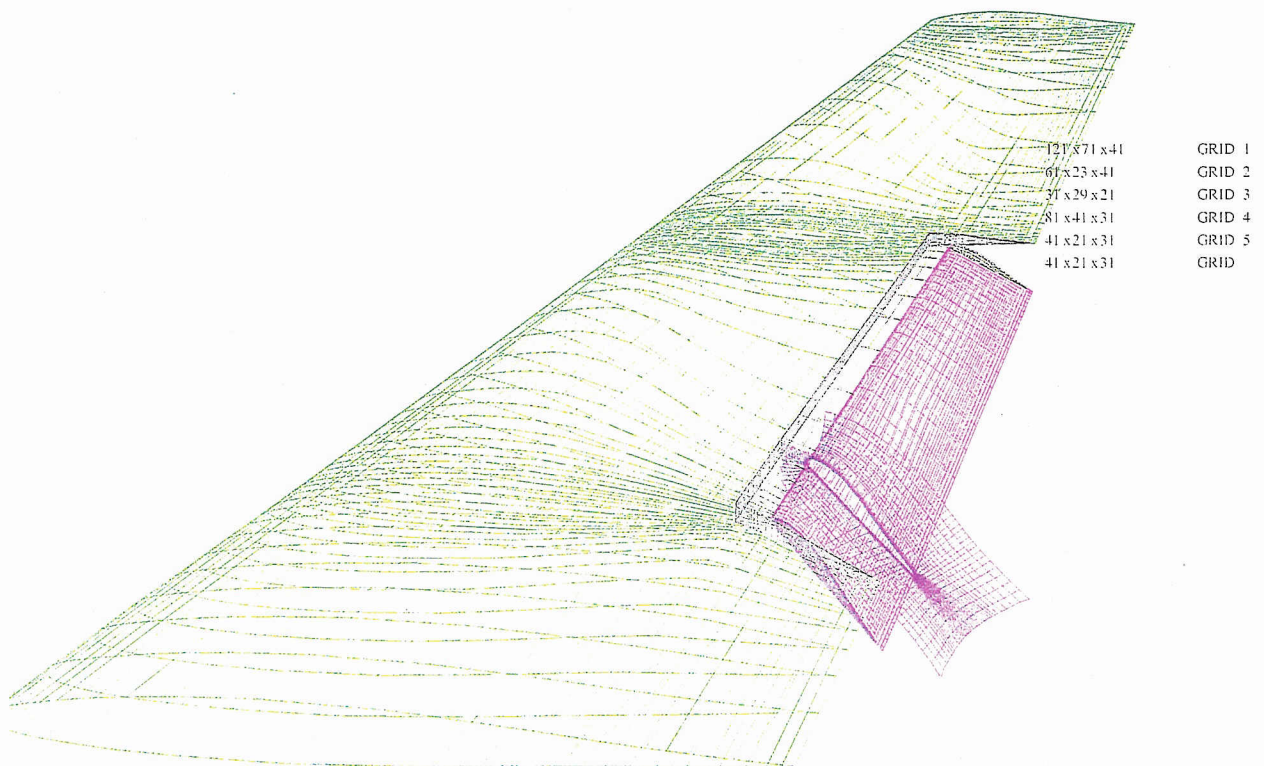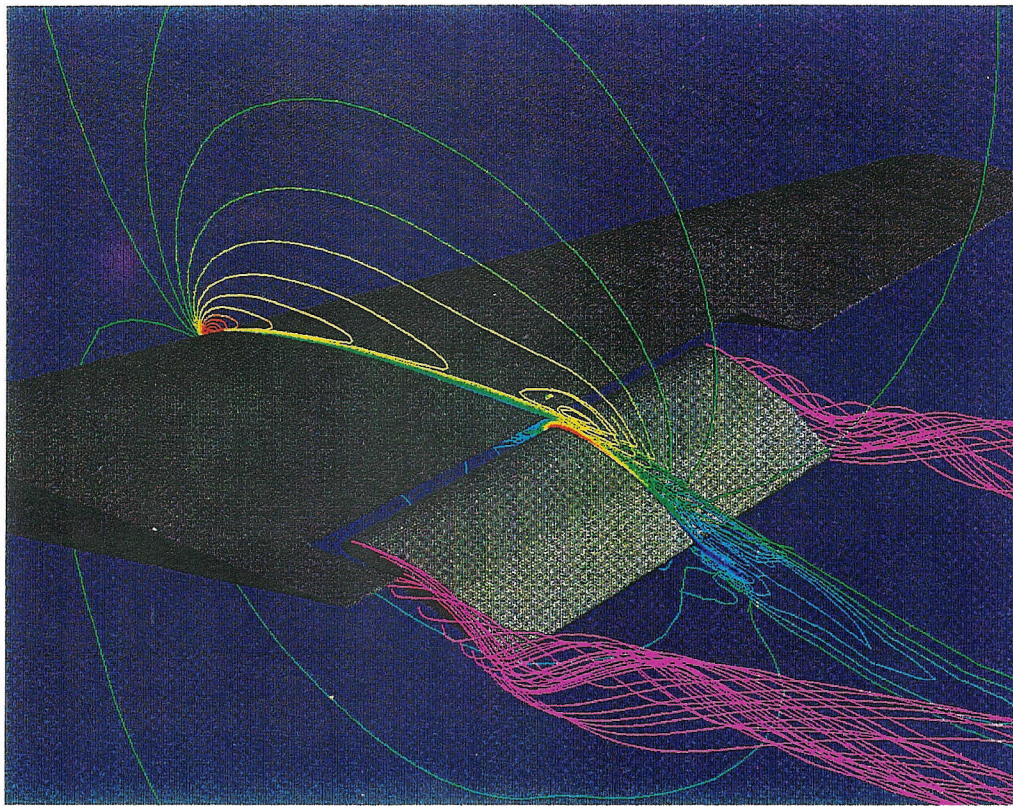
Plate 11: Navier-Stokes Simulation of the T-39 Wing and Flap.
Supplied by S.E. Rogers.
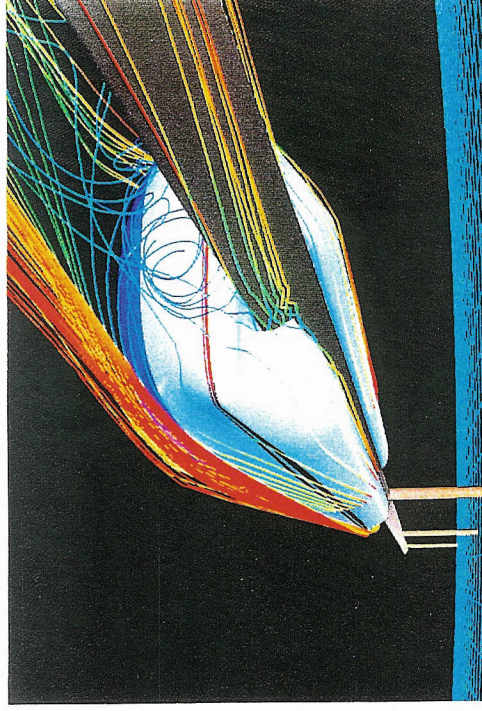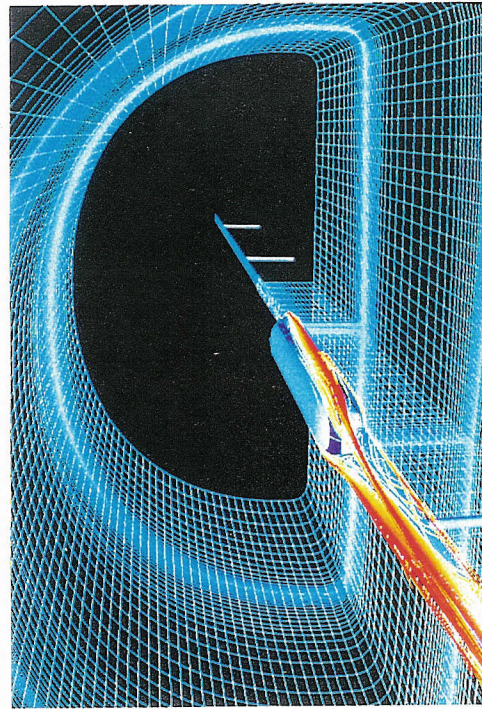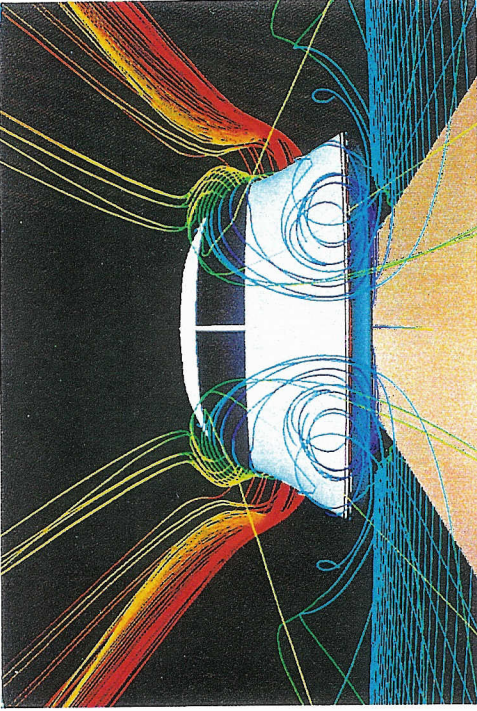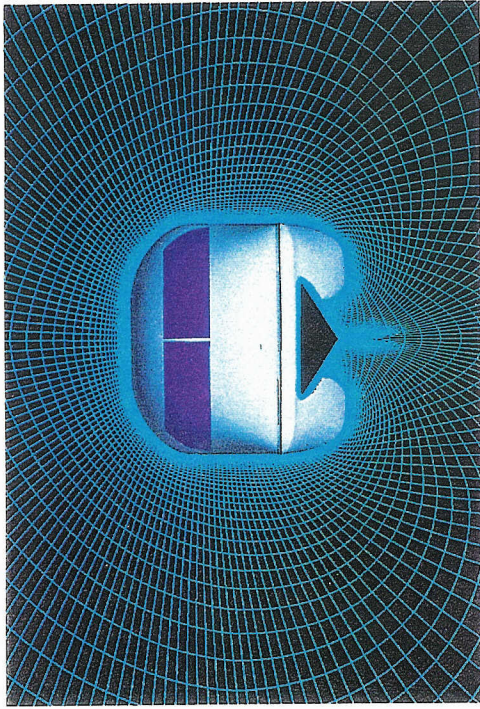
Plate 12:   Navier-Stokes Simulation of the MAGLEV Train.
Supplied by M.J. Siclari.

Inducer Code Validation

Design Speed : 3600 rpm

Design Flow : 2236 gal/min

Tip Diameter : 6 in.   **Reynolds Number : 1.918e+5**

TEST CONDITIONS

Radial Tip Clearance : 0.008 in.

Fluid Medium : water at 70 F
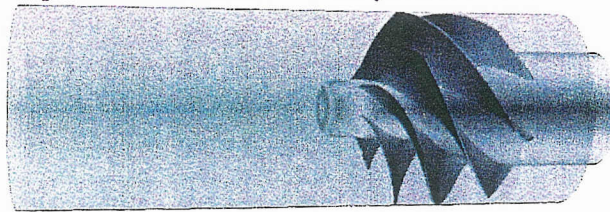
Geometry
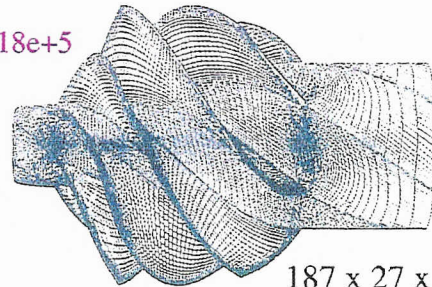
187 x 27 x 35

Computational Grid

Pressure

- 0.69
- 0.14
- −0.40
- −0.95
- −1.50
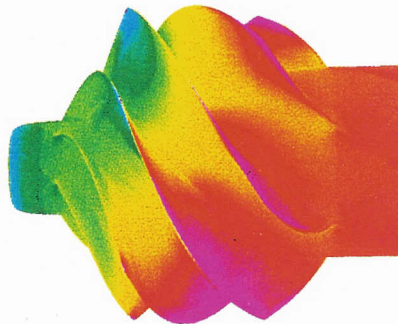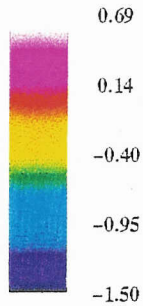
Surface Colored by Pressure
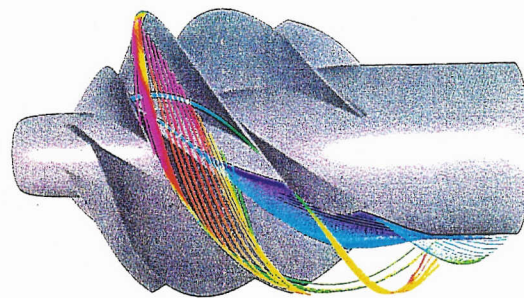
INS3D−UP with Baldwin−Barth One−equation model

Plate 13: Incompressible Navier-Stokes Simulation of a Turbopump Inducer.
Supplied by D. Kwak and S. Rogers.
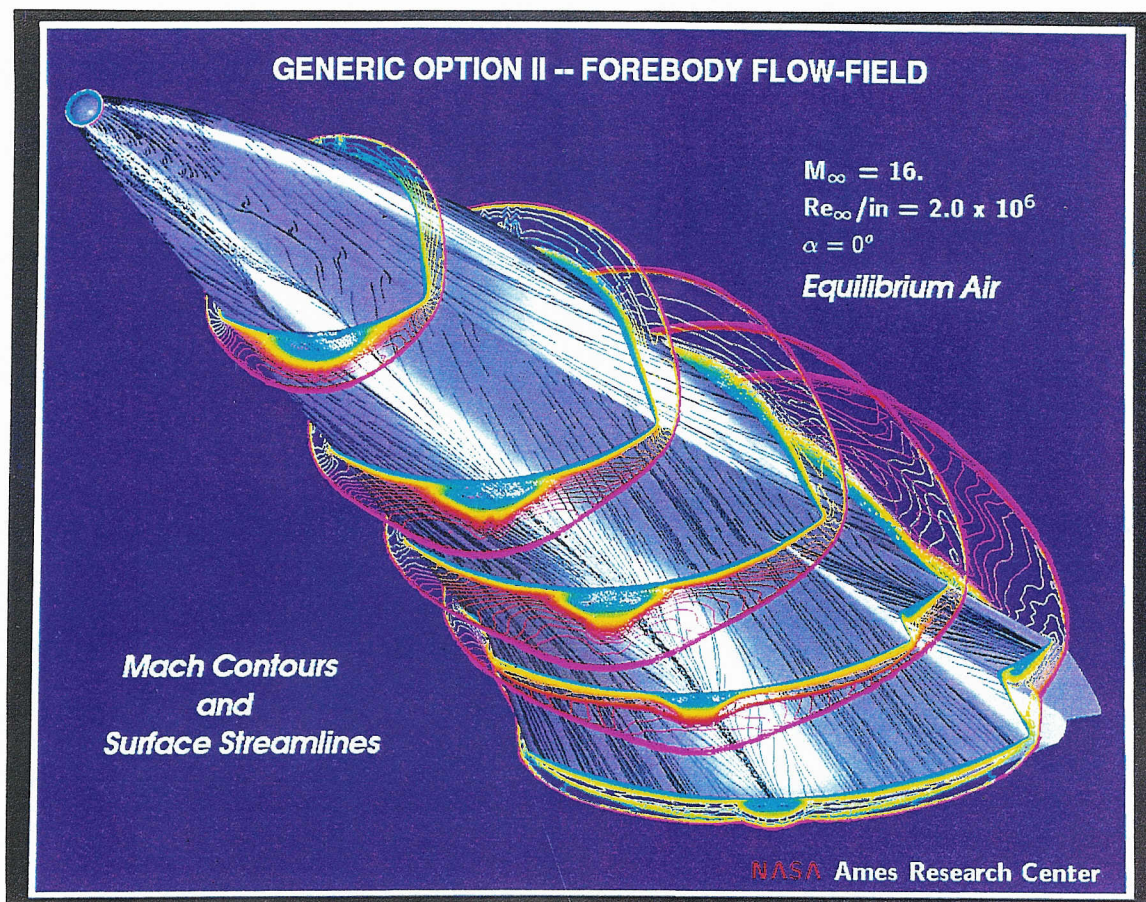
Plate 14: Numerical Simulation of a Hypersonic Forebody.
Supplied by S.L. Lawrence.

Wall Heat Flux (kW/m^2)

Wall Heat Flux (kW/m^2)

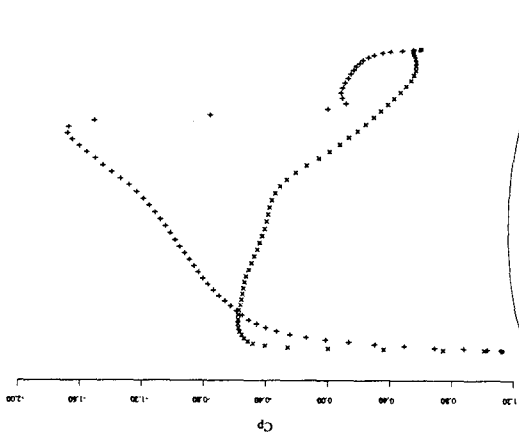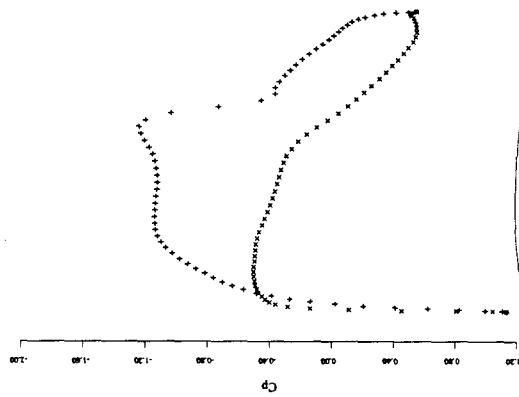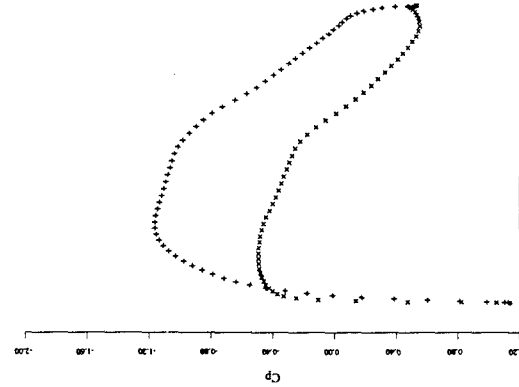|										
0	50	100	150	200	250	300	350	400	450	500
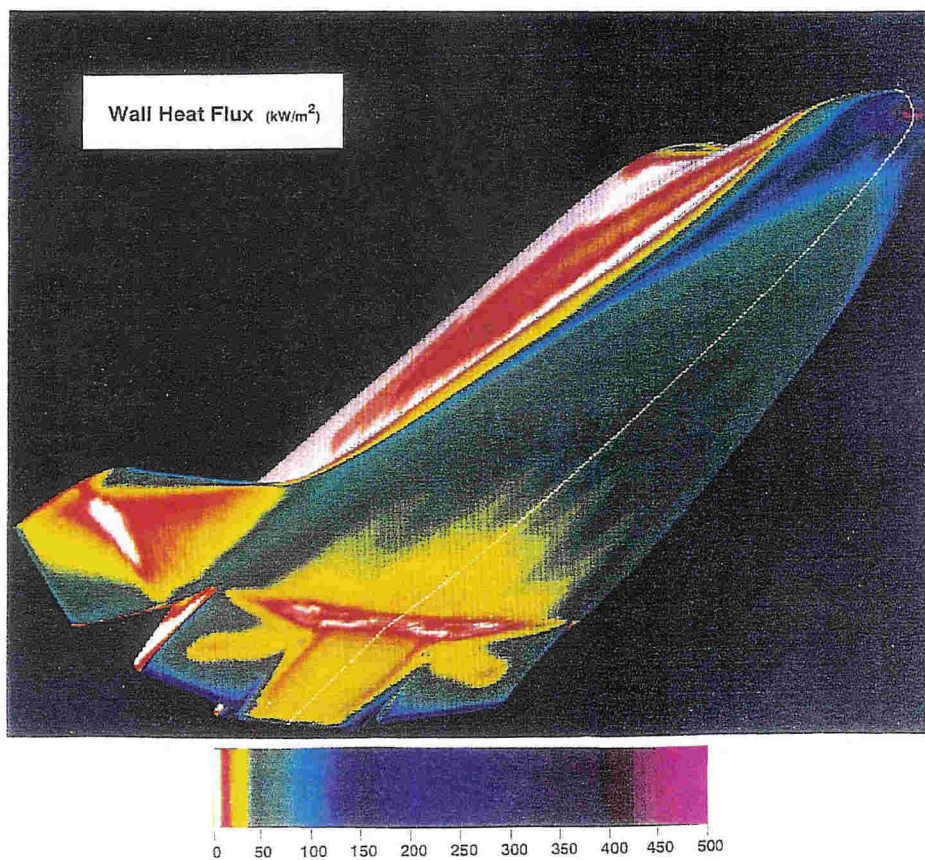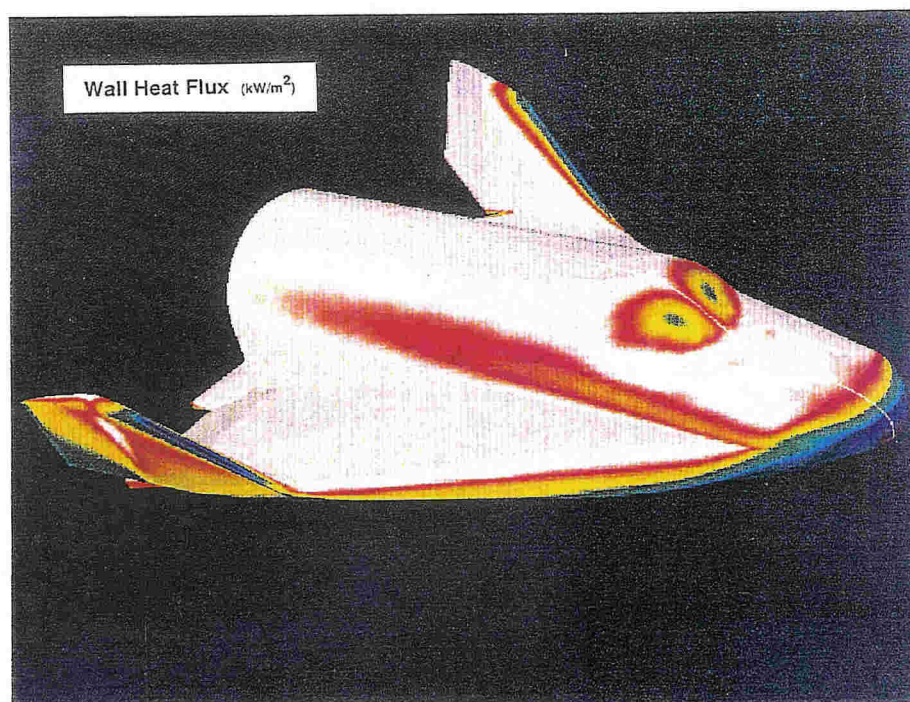
Plate 15: Numerical Simulation of the Hermes Space Shuttle.
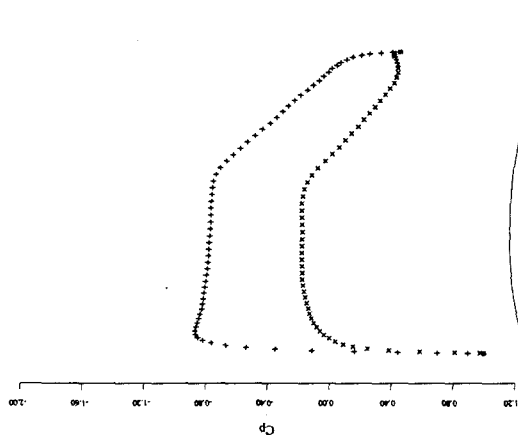*Supplied by K. Dortmann, S. Leicher, and H. Rieger.*

16a: C_p after Zero Design Cycles.
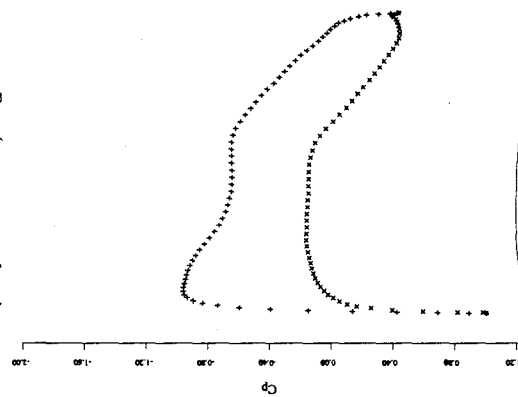Mach 0.72, $C_l = 0.5982$, $C_d = 0.0191$.

16b: C_p after Four Design Cycles.
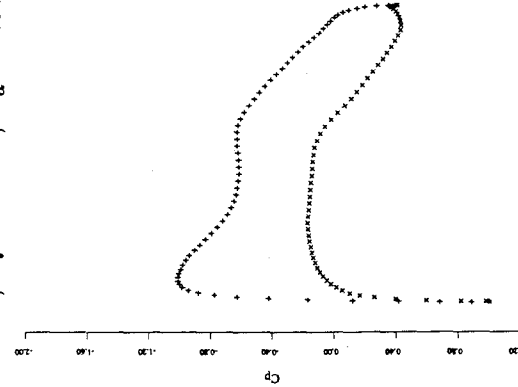Mach 0.72, $C_l = 0.6001$, $C_d = 0.0015$.

16c: C_p after Eight Design Cycles.
Mach 0.72, $C_l = 0.5999$, $C_d = 0.0001$.

16d: C_p after Zero Design Cycles.
Mach 0.2, $C_l = 0.5998$, $C_d = -0.0001$.

16e: C_p after Four Design Cycles.
Mach 0.2, $C_l = 0.5998$, $C_d = -0.0001$.

16f: C_p after Eight Design Cycles.
Mach 0.2, $C_l = 0.5998$, $C_d = -0.0001$.

Plate 16: Optimization of an Airfoil at Two Design Points:
Transonic Drag Reduction with Minimum Modification of the Subsonic Pressure Distribution.