# Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers

**James Reuther, Antony Jameson, Juan Jose Alonso, Mark J. Rimlinger and David Saunders**

The Research Institute of Advanced Computer Science is operated by Universities Space Research Association, The American City Building, Suite 212, Columbia, MD 21044, (410) 730-2656

---

Work reported herein was sponsored by NASA under contract NAS 2-96027 between NASA and the Universities Space Research Association (USRA).

# Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers

**J. Reuther***
Research Institute for Advanced Computer Science
NASA Ames Research Center, MS 227-6
Moffett Field, California 94035, U.S.A.

**A. Jameson**† and **J. J. Alonso**‡
Department of Mechanical and Aerospace Engineering
Princeton University
Princeton, New Jersey 08544, U.S.A.

**M. J. Rimlinger** and **D. Saunders**
Sterling Software
NASA Ames Research Center, MS 227-6
Moffett Field, California 94035, U.S.A.

## ABSTRACT

An aerodynamic shape optimization method that treats the design of complex aircraft configurations subject to high fidelity computational fluid dynamics (CFD), geometric constraints and multiple design points is described. The design process will be greatly accelerated through the use of both control theory and distributed memory computer architectures. Control theory is employed to derive the adjoint differential equations whose solution allows for the evaluation of design gradient information at a fraction of the computational cost required by previous design methods [5, 4, 24, 18]. The resulting problem is implemented on parallel distributed memory architectures using a domain decomposition approach, an optimized communication schedule, and the MPI (Message Passing Interface) standard for portability and efficiency. The final result achieves very rapid aerodynamic design based on a higher order CFD method.

In order to facilitate the integration of these high fidelity CFD approaches into future multi-disciplinary optimization (MDO) applications, new methods must be developed which are capable of simultaneously addressing complex geometries, multiple objective functions, and geometric design constraints. In our earlier studies [8, 9, 10, 11, 19, 15, 20, 21, 22, 23, 1], we coupled the adjoint based design formulations with unconstrained optimization algorithms and showed that the approach was effective for the aerodynamic design of airfoils, wings, wing-bodies, and complex aircraft configurations. In many of the results presented in these earlier works, geometric constraints were satisfied either by a projection into feasible space or by posing the design space parameterization such that it automatically satisfied constraints. Furthermore, with the exception of reference [9] where the second author initially explored the use of multipoint design in conjunction with adjoint formulations, our earlier works have focused on single point design efforts. Here we demonstrate that the same methodology may be extended to treat complete configuration designs subject to multiple design points and geometric constraints. Examples are presented for both transonic and supersonic configurations ranging from wing alone designs to complex configuration designs involving wing, fuselage, nacelles and pylons.

## INTRODUCTION

To realize the potential of CFD to produce superior aircraft designs, high fidelity aerodynamic analysis and maturing sensitivity analysis methods must be integrated in an MDO method. In an accompanying paper presented by the second author at this conference [12], the necessity of developing new methods capable of complete configuration design is discussed. Without such approaches, benefits obtained through, say, a wing only design are likely to deteriorate quickly once the configuration is integrated with the fuselage and nacelles. Further, for some problems such as transonic designs with significant aft wing loading, it will be necessary to perform the design work with viscous effects included. Despite the fact that flow analysis has matured to the extent that Navier-Stokes calculations are routinely carried out over very complex configurations, direct CFD based design is only just beginning to be used in the treatment of moderately complex three-dimensional configurations.

Existing CFD analysis methods have been used to treat the design problem by coupling them with numerical optimization methods [5, 4, 24, 18]. The essence of these methods, which incur heavy computational expense, is very simple: a numerical optimization procedure is used to extremize a chosen aerodynamic figure of merit which is evaluated by the given CFD code. The configuration is systematically modified through user specified design variables. Most of these optimization procedures require the gradient of the cost function with respect to changes in the design variables. The simplest of the methods to obtain these necessary gradients is the finite difference method. In this technique, the gradient components are

*Student Member AIAA
†James S. McDonnell Distinguished University Professor of Aerospace Engineering, AIAA Fellow
‡Student Member AIAA

estimated by independently perturbing each design variable with a finite step, calculating the corresponding value of the objective function using CFD analysis, and forming the ratio of the differences. The gradient is then used by the numerical optimization algorithm to calculate a search direction. After finding the minimum or maximum of the objective function along the search direction, the entire process is repeated until the gradient approaches zero and further improvement is not possible.

The finite-difference approach which has been historically used to calculate the aerodynamic sensitivities is especially inappropriate when complex configurations, requiring hundreds or even thousands of design variables, and multiple design points are to be considered. Nevertheless, it is attractive when compared with other traditional design strategies such as inverse methods, since it permits any choice of the aerodynamic figure of merit. The use of numerical optimization for transonic aerodynamic shape design was pioneered by Hicks, Murman and Vanderplaats [5]. They applied the method to two-dimensional profile design governed by the potential flow equation. The method was quickly extended to wing design by Hicks and Henne [4]. Later, in the work of Reuther, Cliff, Hicks and Van Dam, this method was successfully used for the design of supersonic wing-body transport configurations [18]. However all of these cases, which were confined to finite difference gradients on serial computer architectures, were limited in their geometric complexity simply due to computational expense. For example, the designs presented in [18] were limited to wing-body configurations. Yet it is well known that optimum performance (especially for supersonic configurations) will require highly tuned nacelle/airframe integrations. It was not possible to include nacelle/airframe considerations into the design problem outlined in [18] since the required number of mesh points, which more than doubles with the inclusion of nacelles, could not be afforded.

Recently, through our own work and that of other groups, alternative, less expensive methods for obtaining design sensitivities have been developed. These methods greatly reduce the computational costs of optimization. The most promising of these approaches is the adjoint formulation whereby the sensitivity at a single design point with respect to an arbitrary number of design variables is obtained with the equivalent of two flow calculations (one flow solution and one adjoint solution). Moreover, the adjoint solution (and to a lesser extent the accompanying flow solution) need not be highly converged to be useful, in significant contrast to the highly-converged flow solutions which are crucial to accurate finite difference gradients [25]. For the case of multiple design points, separate flow and adjoint solutions are needed for each design point. Thus, so long as the number of design points remains significantly smaller than the number of design variables the reduction in the computational time resulting from the employment of an adjoint approach will be convincing.

In spite of the large decrease in computational cost provided by an adjoint formulation, the aerodynamic optimization of a complete configuration, especially with the inclusion of viscous effects or taken in the context of even larger MDO problems, still remains a formidable computational task. The advent of reliable and efficient parallel computers using distributed memory is thus a key enabling technology for design calculations of complete aircraft configurations subject to multiple design conditions and for both linear and nonlinear constraints to be treated in an acceptable turnaround time.

The work presented in this paper combines the adjoint formulation and a parallel implementation such that objective function evaluations and aerodynamic sensitivities for any particular point in the design space may be calculated in a very short wall clock time. This approach is then coupled to a constrained sequential quadratic programming (SQP) optimization algorithm [2]. Linear constraints are treated by a projection into feasible space while nonlinear constraints are addressed through an augmented Lagrangian approach.

## FORMULATION OF THE ADJOINT EQUATIONS

The aerodynamic properties which define the cost function $I$ at a single design point are functions of the flow field variables, $w$, and the physical location of the boundary, which may be represented by the function $\mathcal{F}$. Then

$$I = I(w, \mathcal{F})$$

and a change in $\mathcal{F}$ results in a change

$$\delta I = \frac{\partial I^T}{\partial w}\delta w + \frac{\partial I^T}{\partial \mathcal{F}}\delta \mathcal{F} \tag{1}$$

in the cost function. The governing equation $R$ and its first variation express the dependence of $w$ and $\mathcal{F}$ within the flow field domain $D$:

$$R(w, \mathcal{F}) = 0, \quad \delta R = \left[\frac{\partial R}{\partial w}\right]\delta w + \left[\frac{\partial R}{\partial \mathcal{F}}\right]\delta \mathcal{F} = 0. \tag{2}$$

Next, introducing a Lagrange multiplier $\psi$, we have

$$\begin{aligned}\delta I &= \frac{\partial I^T}{\partial w}\delta w + \frac{\partial I^T}{\partial \mathcal{F}}\delta \mathcal{F} - \psi^T\left(\left[\frac{\partial R}{\partial w}\right]\delta w + \left[\frac{\partial R}{\partial \mathcal{F}}\right]\delta \mathcal{F}\right) \\ &= \left\{\frac{\partial I^T}{\partial w} - \psi^T\left[\frac{\partial R}{\partial w}\right]\right\}\delta w + \left\{\frac{\partial I^T}{\partial \mathcal{F}} - \psi^T\left[\frac{\partial R}{\partial \mathcal{F}}\right]\right\}\delta \mathcal{F}.\end{aligned}$$

Choosing $\psi$ to satisfy the adjoint equation

$$\left[\frac{\partial R}{\partial w}\right]^T \psi = \frac{\partial I}{\partial w} \tag{3}$$

the first term is eliminated, and we find that the desired gradient is given by

$$\mathcal{G}^T = \frac{\partial I^T}{\partial \mathcal{F}} - \psi^T\left[\frac{\partial R}{\partial \mathcal{F}}\right]. \tag{4}$$

Since (4) is independent of $\delta w$, the gradient of $I$ with respect to an arbitrary number of design variables can be determined without the need for additional flow field evaluations. The main cost is in solving the adjoint equation (3). In general, the adjoint problem is about as complex as a flow solution. If the number of design variables is large, it becomes compelling to take advantage of the cost differential between one adjoint solution and the large number of flow field evaluations required to determine the gradient by finite differences. To treat the multipoint problem a composite cost function is developed as a weighted sum of cost functions at each independent design point:

$$\mathcal{I} = \lambda_1 I_1 + \lambda_2 I_2,$$

where $\lambda_1$ and $\lambda_2$ are the relative weights of the two cost functions at different design points. The composite gradient is then obtained by taking the same weighted sum of the gradients developed by the above procedure for each point.

## MULTIBLOCK FLOW SOLUTION

In our most recent papers [17, 23] the adjoint based design formulation was extended for the Euler equations to treat complete aircraft configurations via a new multiblock implementation. This extension of the method from that presented in our earlier three-dimensional work required the replacement of the single block flow and adjoint solvers [10, 20, 22] with their multiblock counterparts. During the development of the new multiblock flow solver, care was taken to satisfy additional demands that design methods place upon analysis algorithms.

In order to use CFD in an automated design environment, the flow solver must meet fundamental requirements of accuracy, efficiency, and robust convergence. High accuracy is required since the predicted improvements in the design realized by the method can only be as good as the accuracy of the flow analysis. Efficiency of the flow solver is also critical since the optimization of the design will generally require the computation of many flow solutions or other solutions of comparable complexity. The last aspect, robust convergence, is also of significant importance. In highly refined aerodynamic design applications, the main benefit of aerodynamic optimization is in obtaining the last few percentage points in improved efficiency. In such cases the solutions must be highly converged so that the noise in the figure of merit, say drag at a fixed lift, is well below the level of realizable improvement.

In our three-dimensional single block applications, the FLO87 code written by the second author easily met all of the above criteria. FLO87 achieves fast convergence with the aid of multigridding and residual smoothing. It is normally easy to obtain solutions that converge to machine accuracy. The challenge in reference [23] was to meet these strict convergence requirements within the framework of a multiblock flow solver. The usage of a multiblock flow solution is the first step towards the treatment of more complex configurations. However, the use of the multiblock strategy to treat the complex configurations that are presented in this paper is not the only viable approach. Other alternatives, such as unstructured mesh solvers, are also currently under investigation.

The general strategy in developing the multiblock flow solver is to construct and update a halo of cells around each block such that the flow solution inside each block is transparent to the block boundaries. This task requires establishing the size and location of halo cells adjacent to block boundaries, and loading the halo cell values with appropriate flow field data at the appropriate times. To accomplish this task, a two-level halo is constructed around each block. The requirement of this double halo results from the necessity of keeping a complete stencil of calculated fluxes entering and leaving each cell in the entire domain without regard to block boundaries. Since both the convective and the dissipative fluxes are calculated at the cell faces (boundaries of the control volumes), all six neighboring cells are necessary, thus requiring the existence of a single level halo for each block in the multiblock calculation. The dissipative fluxes are composed of a blend of first and third order differences corresponding to terms that mimic second and fourth derivatives of the flow quantities [16]. This requires a second layer of halo cells at each block interface. Halo cells on the external boundary of the entire computational domain are constructed and updated by extrapolation and reflection. Coarse grids are computed in the usual fashion, by aggregating groups of eight cells and then repeating the above halo cell process. Once the halo configuration is set up for each block, standard methods for spatial discretization and time integration (including artificial dissipation, residual averaging, and multigridding) are employed to compute the flow solution within each individual block.

The system of equations solved here as well as the solution strategy follows that presented in many earlier works [16, 7, 6]. The three-dimensional Euler equations may be written as

$$\frac{\partial w}{\partial t} + \frac{\partial f_i}{\partial x_i} = 0 \quad \text{in } D, \tag{5}$$

where it is convenient to denote the Cartesian coordinates and velocity components by $x_1$, $x_2$, $x_3$ and $u_1$, $u_2$, $u_3$, and $w$ and $f_i$ are defined as

$$w = \begin{Bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{Bmatrix}, \quad f_i = \begin{Bmatrix} \rho u_i \\ \rho u_i u_1 + p\delta_{i1} \\ \rho u_i u_2 + p\delta_{i2} \\ \rho u_i u_3 + p\delta_{i3} \\ \rho u_i H \end{Bmatrix} \tag{6}$$

with $\delta_{ij}$ being the Kronecker delta function. Also,

$$p = (\gamma - 1)\rho\left\{E - \frac{1}{2}\left(u_i^2\right)\right\}, \tag{7}$$

and

$$\rho H = \rho E + p \tag{8}$$

where $\gamma$ is the ratio of the specific heats. Consider a transformation to coordinates $\xi_1$, $\xi_2$, $\xi_3$ where

$$K_{ij} = \left[\frac{\partial x_i}{\partial \xi_j}\right], \quad J = \det(K), \quad K_{ij}^{-1} = \left[\frac{\partial \xi_i}{\partial x_j}\right].$$

Introduce scaled contravariant velocity components as

$$U_i = Q_{ij} u_j$$

where

$$Q = JK^{-1}.$$

The Euler equations can now be written as

$$\frac{\partial W}{\partial t} + \frac{\partial F_i}{\partial \xi_i} = 0 \quad \text{in } D, \tag{9}$$

with

$$W = J\begin{Bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{Bmatrix}, \quad F_i = Q_{ij} f_j = \begin{Bmatrix} \rho U_i \\ \rho U_i u_1 + Q_{i1} p \\ \rho U_i u_2 + Q_{i2} p \\ \rho U_i u_3 + Q_{i3} p \\ \rho U_i H \end{Bmatrix}. \tag{10}$$

For the multiblock case, the above notation applies to each block in turn. The flow is thus determined as the steady-state solution to equation (9) in all blocks subject to the flow tangency conditions on all solid boundary faces of all blocks:

$$U_n = 0 \quad \text{on all } B_S \tag{11}$$

where $\eta$ is 1, 2, or 3 depending on the direction that is normal to face $B_S$ where a solid surface is assumed to exist. At the far field boundary faces, $B_F$, freestream conditions are specified for incoming waves, while outgoing waves are determined by the solution.

The time integration scheme follows that used in the single block strategy [16]. The solution proceeds by performing the cell flux balance, updating the flow variables, and smoothing the residuals, at each stage of the time stepping scheme and each level of the multigrid cycle. The main difference in the integration strategy is the need to loop over all blocks during each stage of the integration process. The use of the double-halo configuration permits standard single-block subroutines to be used, without modification, for the computation of the flow field within each individual block. This includes the single-block subroutines for convective and dissipative flux discretization, multistage time stepping, and multigrid convergence acceleration.

The only difference between the integration strategies is in the implementation of the residual averaging technique. In the single-block solution strategy, a tridiagonal system of equations is set up and solved using flow information from the entire grid. Thus, each residual is replaced by a weighted average of itself and the residuals of the entire grid. In the multiblock strategy, the support for the residual smoothing is reduced to the size of each block, in order to eliminate the need to solve scalar tridiagonal systems spanning the blocks, which would incur a penalty in communication costs. This change has no effect on the final converged solution, and in the present application has not led to any reduction in the rate of convergence.

## THE ADJOINT FORMULATION FOR THE EULER EQUATIONS

The application of control theory to aerodynamic design problems is illustrated by treating the case of three-dimensional design, using the Euler equations discussed above as the mathematical model for compressible flow. In our previous work, the illustrative problem most often used specified the cost function as a measure of the difference between the current and some desired pressure distribution. In the case of transonic flows over conventional commercial transport wings this aerodynamic figure of merit proves to be very effective since the tailoring of these pressure distributions to achieve close to optimum performance is well understood by most aerodynamicists. However, for either the case of supersonic design of three-dimensional configurations or designs which involve complicated geometries the specification of pressure distributions that will determine near optimum performance is considerably more challenging. Therefore, from the outset of our development of the adjoint formulation, many different cost functions or their combinations have been allowed. Here, for illustrative purposes we will use drag at a fixed lift as the cost function.

$$\begin{aligned} I = \quad & C_D \\ = \quad & C_A \cos\alpha + C_N \sin\alpha \\ = \quad & \frac{1}{\mathbf{S_{ref}}} \int\int_{B_S} C_P \left( S_x \cos\alpha + S_y \sin\alpha \right) d\xi_1 d\xi_2, \end{aligned}$$

where $S_x$ and $S_y$ define projected surface areas, $\mathbf{S_{ref}}$ is the reference area, and $d\xi_1$ and $d\xi_2$ are the two coordinate indices that are in the plane of the face in question. Note that the integral in the final expression above is carried out over all solid boundary faces. The design problem is now treated as a control problem where the control function is the geometry shape, which is chosen to minimize $I$, subject to the constraints defined by the flow equations (5–10). A variation in the shape will cause a variation $\delta p$ in the pressure and consequently a variation in the cost function

$$\delta I = \tilde{\delta} C_D + \frac{\partial C_D}{\partial \alpha} \delta\alpha$$

where $\tilde{\delta} C_D$ is the variation due to changes in the design parameters with $\alpha$ fixed. To treat the interesting problem of practical design, drag must be minimized at a fixed lift coefficient. Thus an additional constraint is given by

$$\delta C_L = 0,$$

which gives

$$\tilde{\delta} C_L + \frac{\partial C_L}{\partial \alpha} \delta\alpha = 0$$

Combining these two expressions to eliminate $\delta\alpha$ gives

$$\delta I = \tilde{\delta} C_D - \frac{\left(\frac{\partial C_D}{\partial \alpha}\right)}{\left(\frac{\partial C_L}{\partial \alpha}\right)} \tilde{\delta} C_L. \tag{12}$$

Since $p$ depends on $w$ through the equation of state (7–8), the variation $\delta p$ can be determined from the variation $\delta w$. If a fixed computational domain is used, the variations in the shape result in variations in the mapping derivatives. Define the Jacobian matrices

$$A_i = \frac{\partial f_i}{\partial w}, \quad C_i = Q_{ij} A_j. \tag{13}$$

Then the equation for $\delta w$ in the steady state becomes

$$\frac{\partial}{\partial \xi_i} (\delta F_i) = 0, \tag{14}$$

where in the domain

$$\delta F_i = C_i \delta w + \delta \left( Q_{ij} \right) f_j,$$

and on the solid surface,

$$\delta F_\eta = \begin{Bmatrix} 0 \\ Q_{\eta 1} \delta p \\ Q_{\eta 2} \delta p \\ Q_{\eta 3} \delta p \\ 0 \end{Bmatrix} + p \begin{Bmatrix} 0 \\ \delta \left( Q_{\eta 1} \right) \\ \delta \left( Q_{\eta 2} \right) \\ \delta \left( Q_{\eta 3} \right) \\ 0 \end{Bmatrix} \text{ on any } B_S. \tag{15}$$

Now, multiplying equation (14) by a vector co-state variable $\psi$, assuming the result is differentiable, and integrating by parts over the entire domain,

$$\int_D \left( \frac{\partial \psi^T}{\partial \xi_i} \delta F_i \right) d\xi_j - \int_B \left( \bar{n}_i \psi^T \delta F_i \right) d\xi_j = 0, \tag{16}$$

where $\bar{n}_i$ are components of a unit vector normal to the boundary. Equation (16) can now be subtracted from equation (12) without changing the value of $\delta I$. Then $\psi$ may be chosen to cancel the

explicit terms in $\delta w$ and $\delta p$. For this purpose $\psi$ is set to the steady-state solution of the adjoint equation

$$\frac{\partial \psi}{\partial t} - C_i^T \frac{\partial \psi}{\partial \xi_i} = 0 \quad \text{in } D, \tag{17}$$

with the surface boundary condition

$$\left(\psi_2 Q_{\eta 1} + \psi_3 Q_{\eta 2} + \psi_4 Q_{\eta 3}\right) = \mathcal{Q} \text{ on all } B_S, \tag{18}$$

where

$$\mathcal{Q} = \frac{1}{\frac{1}{2}\gamma M_\infty^2 \mathbf{S_{ref}}} \left\{ \left(S_x \cos\alpha + S_y \sin\alpha\right) + \Omega \left(S_y \cos\alpha - S_x \sin\alpha\right) \right\}.$$

At internal block boundaries, the face integrals cancel from the adjacent blocks. At the far field the choice of the adjoint boundary conditions depends on whether the flow is subsonic or supersonic. For subsonic flow, so long as the outer domain is very far from the configuration of interest, we may set

$$\psi_{1-5} = 0 \quad \text{on all} \quad B_F.$$

It is noted that the waves in the adjoint problem propagate in the opposite direction to those in the flow problem due to the transpose in equation (17).

Finally we obtain the expression

$$\delta I = \frac{1}{\mathbf{S_{ref}}} \int\int_{B_S} C_p \left\{ \left(\delta S_x \cos\alpha + \delta S_y \sin\alpha\right) + \Omega \left(\delta S_y \cos\alpha - \delta S_x \sin\alpha\right) \right\} d\xi_1 d\xi_2 + \int_D \psi^T \frac{\partial}{\partial \xi_i} \left(\delta Q_{ij} f_j\right) d\xi_k. \tag{19}$$

Details of the approach as well as the development for other cost functions have been presented in references [10, 11, 15, 20, 21, 23].

## MULTIBLOCK MESH VARIATIONS

In order to construct $\delta I$ in equation (19), the variation in the metric terms must be obtained in each block. One way to accomplish this is to use finite differences to calculate the necessary information. This approach avoids the use of multiple flow solutions to determine the gradient, but it unfortunately still requires the mesh generator to be used repeatedly. The number of mesh solutions required is proportional to the number of design variables. The inherent difficulty in the approach is two-fold. First, for complicated three-dimensional configurations, elliptic or hyperbolic partial differential equations must often be solved iteratively in order to obtain acceptably smooth meshes. These iterative mesh generation procedures are often computationally expensive. In the worst case they approach the cost of the flow solution process. Thus the use of finite difference methods for obtaining metric variations in combination with an iterative mesh generator leads to computational costs which strongly hinge on the number of design variables, despite the use of an adjoint solver to eliminate the flow variable variations. Second, multiblock mesh generation is by no means a trivial task. In fact no method currently exists that allows this to be accomplished as a completely automatic process for complex three-dimensional configurations.

In our earlier works [20, 19, 15, 8, 9, 10], two methods have been explored which avoid these difficulties. In the first method, a completely analytic mapping procedure was used for the mesh generation. This technique is not only fully automatic and results in smooth consistent meshes, but it also allows for complete elimination of finite difference information for the mesh metric terms. Since the mapping function fully determines the entire mesh based on the surface shape, this analytic relationship may be directly differentiated in order to obtain the required information without considering a finite step. An analytic mapping method requires the geometry topology to be built directly into the formulation, and only works for simple configurations. Nevertheless, within these limitations it has proven to be highly effective [8, 9, 10].

The second method that we have explored is the use of an analytic mesh perturbation technique. In this approach, a high quality mesh appropriate for the flow solver is first generated by any available procedure prior to the start of the design. In examples to be shown later, these meshes were created using the Gridgen software developed by Pointwise, Inc.[26]. This initial mesh becomes the basis for all subsequent meshes which are obtained by analytical perturbations. In the method that was initially developed for wing-body configurations it had been assumed that only one surface, say the wing, was perturbed during a design case. This permitted the use of a very simple algebraic mesh perturbation algorithm. New meshes were created by moving all the mesh points on an index line projecting from the surface by an amount which was attenuated as the arc length from the surface increased. If the outer boundary of the grid domain is held constant the modification to the grid has the form

$$x_i^{new} = x_i^{old} + \mathcal{S}^{old} \left(x_{s_i}^{new} - x_{s_i}^{old}\right) \tag{20}$$

where $x_i$ represents the volume grid points, $x_{s_i}$ represents the surface grid points and $\mathcal{S}$ represents the arc length along the radial mesh line measured from the outer domain, normalized so that $\mathcal{S} = 1$ at the inner surface. Unfortunately this simple logic breaks down in the case where multiple faces sharing common edges are allowed to move. Thus in order to use analytic mesh perturbations for the treatment of the more general problem where multiple faces of a given block may be simultaneously deformed, equation (20) had to be modified in a way that resembles transfinite interpolation (TFI) [27]. Unlike TFI, where there is no prior knowledge of the interior mesh, the perturbation algorithm developed here (WARP3D) does make use of the relative interior point distributions in the initial mesh.

The WARP3D algorithm has been modified from that presented in reference [23] and is now a three stage procedure [17]. The first stage shifts the internal mesh points to produce an interim block that is determined entirely by the new locations of the 8 corner points defining the block. The second stage corrects the perturbations resulting from the first stage by determining the distance each of the 12 edges of the stage 1 block needs to be moved to attain the desired edge locations. Finally with both corner and edge point motion accounted for, the third stage corrects the internal points for the relative motion of the six faces.

Since our current flow solver and design algorithm assume a point-to-point match between blocks, each block may be independently perturbed by WARP3D, provided that perturbed surfaces are treated continuously across block boundaries. The entire method of creating a new mesh is given by the following algorithm.

1. All faces that are directly affected by the design variables (active

faces) are explicitly perturbed.

2. All edges that touch an active face, either in the same block or in an adjacent block, are implicitly perturbed by (20).

3. All inactive faces that either include an implicitly perturbed edge or abut to an active face are implicitly perturbed by a quasi-3D form of WARP3D.

4. WARP3D is used on each block that has one or more explicitly or implicitly perturbed faces to determine the adjusted interior points.

Note that much of the mesh, especially away from the surfaces, will not require mesh perturbations and thus may remain fixed through the entire design process. Close to the surfaces, many blocks will either contain an active face or touch a block which contains an active face, either by an edge or by a corner. As the design variations affect the active faces, the above scheme ensures that the entire mesh will remain attached along block boundaries. Added complexity is needed to accomplish step (2) since the connectivity of the various edges and corners must be indicated somehow. Currently, pointers to and from a set of master edges and master corners are determined as a preprocessing step. During the design calculation, deflections to any edges or corners are fed to these master edges and master corners which in turn communicate these changes to all connected edges and corners.

Since this mesh perturbation algorithm is analytic it is possible to work out the analytical variations in the metric terms required for equation (19). This approach was followed in [20]. However since the mesh perturbation algorithm that was used in the current paper was significantly more complex, and it was discovered that the computational cost of repeatedly using the block perturbation algorithm was within reason, finite differences were used to calculate $\delta Q_{i,j}$ instead of deriving the exact analytical relationships.

## DESIGN VARIABLES AND CONSTRAINED OPTIMIZATION

It remains to choose design variables and impose constraints. In our earlier work, where the analytic mesh mapping strategy was used, each point of the surface mesh served as a design variable. This technique combined with smoothing and projecting into feasible space of the resulting gradient has proven to be highly effective for single-block design cases [8, 9, 10, 11, 15, 19, 20].

Alternatively, geometric design variables and constraints may be provided for selection by the designer to allow greater control over the design process. To develop this strategy, the multiblock approach has been coupled to the NPSOL algorithm of Gill, Murray, Saunders, and Wright [2]. NPSOL is a sequential quadratic programming (SQP) method in which the search direction is calculated by solving the quadratic subproblem where the Hessian is defined by a quasi-Newton approximation of an augmented Lagrangian merit function. The Lagrange multipliers in this merit function serve to scale the effect of any nonlinear constraints that the design may contain. Linear constraints are treated by solving the quadratic subproblem such that the projected search direction remains in feasible space. A complete treatment of the method and other optimization strategies is given by Gill, Murray, and Wright [3].

The primary control upon which the entire design process revolves is the variation in the aerodynamic surfaces. This is readily apparent from the development of the adjoint approach. Thus it is at the level of surface variations that the coupling with NPSOL is accomplished. Realizing that aircraft configurations are composed of separate entities (wings, fuselages, nacelles, etc.) upon which constraints are imposed, an underlying set of geometry entities acts as a starting input for the design process. Next, the user-specified design variables are allowed to act independently upon any of these geometry entities. Linear and nonlinear geometric constraints are also evaluated on these primary geometry entities. At any particular point in the design process, changes to the mesh surfaces are obtained by first intersecting all of the geometry entities to construct a set of parametric surfaces representing the complete configuration. The location of each surface mesh point on this parametric representation is determined for the initial configuration in a preprocessing step. The perturbed surface mesh point locations are determined by evaluating the parametric geometry surfaces at these predetermined locations. Once the surface mesh points have been updated, the volume mesh may be perturbed and either the gradient or the solution calculated. The important feature of this approach is that a set of simple geometry entities lies at the core of the entire design process. This technique retains the typical way in which aerodynamic vehicles are defined, and provides strict control over how, say, wing/body intersections are treated. Furthermore, since the chosen design variables act directly upon the geometry entities, at the end of the design process these entities may be output for future analysis.

In the current implementation, the input geometry entities are restricted to those defined by sets of points. However, in the future, CAD entities such as NURBS surfaces will also serve in this role, thereby allowing both the input and the output from the aerodynamic surface optimization to interface directly with a CAD database.

## DOMAIN DECOMPOSITION AND PARALLEL IMPLEMENTATION

The main strategies that are used to accomplish the parallelization of the design code are: a domain decomposition model, a SPMD (Single Program Multiple Data) strategy, and the MPI (Message Passing Interface) library for message passing. The choice of MPI was determined by the requirement that the resulting code be portable to different parallel computing platforms as well as to homogeneous and heterogeneous networks of workstations.

As one can see from the previous sections, obtaining the desired parallelization by domain decomposition entails the treatment of four separate parts: the solution of the flow equations, the solution of the adjoint equations, the calculation of the mesh perturbations, and the calculation of the gradient integral formulas. No attempt is made to parallelize the constrained SQP optimization algorithm or the calculation of the changes to the underlying geometry entities. It is thus assumed in this context that the determination of the step sizes and search directions provided by the optimization algorithm is computationally insignificant when compared with the other elements necessary during the design.

Since the flow and adjoint equations are to be solved using exactly the same efficient numerical techniques, the same parallelization techniques used for the flow equations apply to the solution of the adjoint equations. Therefore, all details of the parallel implementation corresponding to these first two parts of the program are identical. Furthermore, since the mesh perturbation algorithm

WARP3D also works on a block-by-block basis, the communication necessary to maintain mesh consistency can also be addressed by the same domain decomposition strategies that are used for the state and costate fields. The essential details of this decomposition strategy and parallel implementation as well as some calculated parallel speed-ups can be found in [17].

## COMPLETE MULTIBLOCK DESIGN ALGORITHM (SYN87-MB)

With all the necessary components defined for the multiblock adjoint-based design, it is now possible to outline the complete procedure:

1. Decompose the multiblock mesh into an appropriate number of processors, and create lists of pointers for the communication of the processor halo cells.
2. Solve the flow field governing equations (5-10) for each design point.
3. Solve the adjoint equations (17) subject to the boundary condition (18) for each design point.
4. For each of the $n$ design variables repeat the following:
   - Perturb the design variable by a finite step to modify the geometry entities.
   - Reintersect the geometry entities to form parametric geometry surfaces.
   - Explicitly perturb all face mesh points affected by the geometry changes by evaluating their locations on the parametric geometries.
   - Implicitly perturb all faces that share an edge with an explicitly perturbed face.
   - Obtain the new internal mesh point locations via WARP3D for those blocks with perturbed faces.
   - Calculate all the delta metric terms, $\delta Q_{i,j}$, within those blocks that were perturbed by finite differencing.
   - Integrate equation (19) to obtain $\delta I$ for those blocks that contain nonzero $\delta Q_{i,j}$, and for each design point, to determine the gradient component.
5. Calculate the search direction via NPSOL and perform a line search.
6. Return to (2) if a minimum has not been reached.

The basic method here builds on that used in [20] with the proper extensions to treat multiblock domains. In order to implement the method, equation (17) and boundary condition (18) must be discretized on the multiblock domain. In the current implementation, a cell centered, central difference stencil that mimics the flux balancing used for the flow solution is used. Since this choice of discretization differs from the one obtained if the discrete flow equation Jacobian matrix were actually transposed to form the adjoint system, the gradients obtained by the present method will not be exactly equal to the gradients calculated by finite differencing the discrete flow solutions. However, as the mesh is refined these differences should vanish. Continuing, the adjoint system so discretized is solved on the multiblock domain in a fashion identical to that used for the flow solution. Therefore, the adjoint solver, like the flow solver, uses an explicit multistage Runge-Kutta-like algorithm accelerated by residual smoothing and multigridding. Intra-block communication is again handled through a double halo which allows for the full transfer of information across boundaries except for the stencil of support for the implicit residual smoothing.

Step (4) in the above procedure is the portion of the method that is still treated by finite differences. Fortunately, all of these steps incur a small computational cost compared with a single flow analysis. It is therefore possible, without significant penalty, to leave this in finite difference form even for cases where many hundreds of design variables are used.

## NUMERICAL TESTS AND RESULTS

Numerical results will be presented for three classes of problems to demonstrate the versatility of our methods. Reference [23] gives a treatment of the reliability of the flow solver as well as the ability of the adjoint method to provide accurate gradients very efficiently. The parallel speed-ups attained by the method have been demonstrated in [13, 17].

### Transonic Multipoint Wing Design

The first test case explores the viability of using multipoint aerodynamic design for an isolated wing configuration under transonic flight conditions. The version of the adjoint based design code, originally presented by the second author [10, 11], is used as a test bed for the multipoint design. The design variables were the 4,224 surface grid points on the wing, each of which was free to move in the normal direction. The test case wing has a unit-semi-span, with 38 degrees leading edge sweep. The wing has a modified trapezoidal planform, with straight taper from a root chord of 0.38 to a tip chord of 0.10. The wing has an aspect ratio of 9.0 and a curved trailing edge in the inboard region blending into straight taper outboard of the 30% span station. The initial wing sections were based on an airfoil specially designed by the second author's two dimensional design method [8] to give shock free flow at Mach 0.78 with a lift coefficient of 0.6. This section, which has a thickness to chord ratio of 9.5%, was used at the tip. Similar sections with increased thickness and reduced camber were used inboard. The variation of thickness was nonlinear with a more rapid increase near the root, where the thickness to chord ratio of the basic section was multiplied by a factor of 1.44. The inboard sections were rotated upwards to give the initial wing 6 degrees of twist from root to tip.

The two-dimensional pressure distribution of the starting airfoil at its design point was introduced as a target pressure distribution uniformly across the span. This target is presumably not realizable since it would imply a lifting wing with zero vortex drag, but it serves to favor the establishment of a relatively benign pressure distribution. The total inviscid drag coefficient, due to the combination of vortex and shock wave drag, was also included in the cost function. Calculations were performed with the lift coefficient forced to approach a fixed value by adjusting the angle of attack every fifth iteration of the flow solution. A grid with 192x32x48 = 294,912 points was used.

Figures (1 – 7), show the results of the multipoint calculation at a Mach number of 0.85 and three different design lift coefficients

(0.500, 0.525 and 0.550). The plots show the initial wing geometry and pressure distribution, and the modified geometry and pressure distribution after 60 design cycles. Table 1 summarizes the predicted reduction in the total inviscid drags at the three design points. As is

| CL | Initial CD | Final CD |
|---|---|---|
| 0.500 | 0.0156 | 0.0114 |
| 0.525 | 0.0173 | 0.0123 |
| 0.550 | 0.0191 | 0.0135 |

**Table 1: Drag reduction at three different design points**

evident in the figures the shock strengths have been greatly reduced with most of the wing displaying shock free conditions at all three design points. At the low $C_L$ design point there is a very weak incipient double shock pattern, and at the high lift design point there is a very weak single shock. Single point optimization at $C_L = 0.500$ and 0.550 yielded drag coefficients of 0.0112 and 0.0133, so the compromise wing is within 2 drag counts of both point optimized wings at their design points.

## Transonic Constrained Aircraft Design

As a first demonstration of the multiblock optimization algorithm, a typical transonic business jet configuration is considered. The same geometry was also studied in [23, 1].

The initial multiblock mesh about the business jet wing, body, and nacelle has 72 blocks and 750K cells. The underlying geometry entities that are eligible for design changes include a wing with six defining stations and a fuselage. The initial configuration was designed for Mach = 0.8 and $C_L = 0.3$.

In the first design case (Test Case 1), a single point constrained design is attempted in which the Mach number is pushed to 0.82. The objective is to minimize configuration pressure drag at a fixed lift coefficient of 0.3 by modifying the wing shape. Eighteen Hicks-Henne design variables are chosen for five of the six defining sections for a total of 90 design variables. (The section at the symmetry plane is not being modified.) Spar thickness constraints were also enforced at each defining station at $x/c = 0.2$ and $x/c = 0.8$. Maximum thickness was forced to be preserved at $x/c = 0.4$ for all six defining sections. Each section was also constrained to have the thickness preserved at $x/c = 0.95$ to ensure an adequate included angle at the trailing edge. A total of 30 linear geometric constraints were imposed on the configuration. Figure (8) shows an iso-$C_p$ colored representation of the initial design and the final design after 5 NPSOL design iterations. It is clearly seen that the rather low $C_p$ region terminated by a strong shock spanning the entire wing upper surface has been largely eliminated in the final design. Figure (9) shows overlays of the $C_p$ distributions for the initial and final designs at four stations along the wing. It is seen that the final result has reached a near-shock-free condition over much of the outboard wing panel. The drop in configuration pressure drag for this case was 22.5%. Noting that most of this drag reduction came from a decrease in wing wave drag implies that further improvements may be possible through the reshaping of other components.

In a second design example for this business jet configuration, a multipoint design is attempted. The three design points are Mach = 0.81 with a $C_L = 0.35$, Mach = 0.82 with a $C_L = 0.30$, and Mach = 0.83 with a $C_L = 0.25$. Figures (10 – 12) show the initial and final $C_p$ distributions achieved using the same 90 design variables and 30 geometric constraints and 5 NPSOL design iterations as for Test Case 1. Note that for each design point the strong shocks present on both the upper and lower surfaces in the initial configuration have been eliminated. Figures (13 – 15) show comparisons of the three-point solution with the single point solution of Test Case 1 at all three design points. Interestingly, the upper surface shapes for both final designs are very similar. However, in the case of the single-point solution, a strong lower surface shock appears at the Mach = 0.83, $C_L = 0.25$ design point. The three-point solution is able to suppress the formation of this lower surface shock and shows a 9 count drag benefit over the single-point design at this condition. However, it has a 1 count penalty at the design condition for which the single-point case was optimized. As in the wing alone multipoint design, a weak single shock is seen for one of the three design points while a very weak double shock is seen at another design point. Table 2 summarizes the drag results for the two design cases. (The $C_D$ values have been normalized by the drag of the initial configuration at the second design point.) Before proceeding to the next section,

| Design Mach | Conditions CL | Initial CD | Test Case 1 CD | Test Case 2 CD |
|---|---|---|---|---|
| 0.81 | 0.35 | 1.00257 | 0.85003 | 0.85413 |
| 0.82 | 0.30 | 1.00000 | 0.77350 | 0.77915 |
| 0.83 | 0.25 | 1.08731 | 0.81407 | 0.76836 |

**Table 2: Drag reduction for single- and multipoint designs**

it should be noted that these business jet design examples are only representative of the potential for automated design, and are not intended to provide designs for actual construction. First, in each case only 5 NPSOL steps were taken where considerably more could have improved the designs slightly. More importantly, for the case of transonic designs, the inclusion of viscous effects may prove to have an important impact on the optimized shape. In our future transonic studies, the Navier-Stokes equations will be employed in place of the Euler flow solver used here.

## Supersonic Constrained Aircraft Design

In the case of supersonic design, it is conjectured that as long as turbulent flow is assumed over the entire configuration, the inviscid Euler equations suffice for aerodynamic design since the pressure drag does not seem to be greatly affected by the inclusion of viscous effects, and a flat plate skin friction estimate of viscous drag is often close to the mark. In our current study, the generic supersonic transport configuration used in the design studies of reference [17] is revisited.

The baseline supersonic transport configuration was sized to accommodate 300 passengers with a gross take-off weight of 750,000 lbs. The supersonic cruise point is Mach 2.2 with a $C_L$ of 0.105. As can be seen in Figure 16, the planform has a break in the leading edge sweep. The inboard leading edge sweep is 68.5 degrees while the outboard is 49.5 degrees. Since the Mach angle at M = 2.2 is

63 degrees it is clear that some leading edge bluntness may be used inboard without a significant wave drag penalty. Airfoils which use blunt leading edges were created that range from 4% thick at the root to 2.5% thick at the leading edge break point. The symmetric initial airfoils were chosen with the purpose of accommodating thick spars at roughly 10% and 80% chord over the span up to the leading edge break. Outboard of the leading edge break where the wing sweep is ahead of the Mach cone, a sharp leading edge was used to avoid undue wave drag. The airfoils were chosen to be symmetric, biconvex shapes modified to have a region of constant thickness over the midchord. The four-engine configuration features axisymmetric nacelles tucked close to the wing lower surface. This layout favors reduced wave drag by minimizing the exposed diverter area. However, it may be problematic because of the channel flows occurring in the juncture region of the diverter, wing, and nacelle at the wing trailing edge. The leading edge heights of the diverters are determined by the boundary layer local displacement thickness such that entrainment of boundary layer flow into the engines is avoided. Since the distances from the wing leading edge to the diverter leading edge are different for the two nacelles, this causes a corresponding diverter height difference.

The computational mesh on which the design is run has 180 blocks and 1,500K mesh cells, while the underlying geometry entities define the wing with 16 sectional cuts and the body with 200 sectional cuts. In this case, where we hope to optimize the shape of the wing, care must be taken to ensure that the nacelles remain properly attached with the diverter heights maintained. To accomplish this without the inclusion of additional geometry entities, the portions of the nacelles and diverters that are actually below the wing planform outline take their associated surface mesh point motion from their projected locations on the lower parametric wing surfaces.

The objective of the design is to reduce the drag at the single design point (Mach = 2.2, $C_L$ = 0.105) by modifying the wing shape. Just as in the transonic cases, 18 design variables of the Hicks-Henne type are chosen for a given wing defining section. However, instead of applying them to all 16 sections, they are applied to 8 of the sections and then lofted linearly to the neighboring sections. Spar constraints are imposed for all wing defining sections at $x/c = 0.05$ and $x/c = 0.8$. An additional maximum thickness constraint is specified along the span at $x/c = 0.5$. A final thickness constraint is enforced at $x/c = 0.95$ to ensure a reasonable trailing edge included angle. An iso-$C_p$ representation of the initial and final designs is depicted in Figure 16 for both the upper and lower surfaces.

It is noted that the strong oblique shock evident near the leading edge of the upper surface on the initial configuration is largely eliminated in the final design after 5 NPSOL design iterations. It is also seen that the upper surface pressure distribution in the vicinity of the nacelles has formed an unexpected pattern. However, it is recalled that thickness constraints abound in this design, and these upper surface pressure patterns are conjectured to be the result of sculpting of the lower surface near the nacelles which affects the upper surface shape through the thickness constraints. For the lower surface, the leading edge has developed a suction region while the shocks and expansions around the nacelles have been somewhat reduced. Figure 17 shows the pressure coefficients and (scaled) airfoil sections for four sectional cuts along the wing. These further demonstrate the removal of the oblique shock on the upper surface, and the addition of a suction region on the leading edge of the lower surface. The airfoil sections have been scaled by a factor of 2 so that shape changes may be seen more easily. Most notably, the section at 38.7% span has had the lower surface drastically modified such that a large region of the aft airfoil has a forward-facing portion near where the pressure spike from the nacelle shock impinges on the surface. The final overall pressure drag was reduced by 8%, from $C_D$ = 0.0088 to $C_D$ = 0.0081.

## CONCLUSIONS

In the period since this approach to optimal shape design was first proposed by the second author [8], the method has been verified by numerical implementation for both potential flow and flows modeled by the Euler equations [9, 19, 15, 11]. In two accompanying papers at this conference, the method is being extended to treat the Navier-Stokes equations [12, 14]. It has been demonstrated that the method can be used successfully with a finite volume formulation to perform calculations with arbitrary numerically generated grids [19, 15]. Further, results have been presented for three-dimensional calculations using both the analytic mapping and general finite volume implementations [20]. In the last year the technique has been adopted by some industry participants to perform the aerodynamic design of future configurations [1]. With the parallel implementation of the multiblock design algorithm now complete, the technology has advanced to the degree that aerodynamic shape design of complete aircraft configurations with very rapid turnaround is possible.

In this paper we have shown how the complicated design of both transonic and supersonic aircraft configurations including airframe/nacelle integration effects can be accomplished in a routine fashion for multiple design points and with the inclusion of constraints. While the results presented in this paper have been restricted to the inviscid Euler equations, this limitation will soon be overcome in our future work. The focus here was instead to demonstrate the feasibility of realistic designs governed by simple geometry entities that are assembled and treated during the design process. Furthermore, with the proven coupling to NPSOL and the addition of multiple design point capability, the door is truly open for the method to act as a crucial element of a high fidelity MDO technique capable of revolutionizing aircraft design.

All analysis and design cases were performed on parallel architecture machines in less than one day, demonstrating that complete configuration designs may be achieved with rapid turn-around even with the most conservative estimates of available computational resources. In future efforts, additional disciplines will be coupled into the techniques presented here while work continues on the unstructured grid approaches and on the inclusion of viscous effects.

## ACKNOWLEDGMENTS

This research has benefited greatly from the generous support of the AFOSR under grant number AFOSR-91-0391, ARPA under grant number N00014-92-J-1976, USRA through RIACS, the High Speed Research branch of NASA Ames Research Center, and IBM.

### References

[1] J. Gallman, J. Reuther, N. Pfeiffer, W. Forrest, and D. Bernstorf. Business jet wing design using aerodynamic shape optimiza-

tion. *AIAA paper 96-0554*, 34th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1996.

[2] P.E. Gill, W. Murray, M.A. Saunders, and M.A. Wright. User's guide for NPSOL (version 4.0). a FORTRAN package nonlinear programming. *Technical Report SOL86-2*, Stanford University, Department of Operations Research, 1986.

[3] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, 1981.

[4] R. M. Hicks and P. A. Henne. Wing design by numerical optimization. *Journal of Aircraft*, 15:407–412, 1978.

[5] R. M. Hicks, E. M. Murman, and G. N. Vanderplaats. An assessment of airfoil design by numerical optimization. *NASA TM X-3092*, Ames Research Center, Moffett Field, California, July 1974.

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

[7] A. Jameson. Multigrid algorithms for compressible flow calculations. In W. Hackbusch and U. Trottenberg, editors, *Lecture Notes in Mathematics, Vol. 1228*, pages 166–201. Proceedings of the 2nd European Conference on Multigrid Methods, Cologne, 1985, Springer-Verlag, 1986.

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

[9] A. Jameson. Automatic design of transonic airfoils to reduce the shock induced pressure drag. In *Proceedings of the 31st Israel Annual Conference on Aviation and Aeronautics, Tel Aviv*, pages 5–17, February 1990.

[10] A. Jameson. Optimum aerodynamic design via boundary control. In *AGARD-VKI Lecture Series, Optimum Design Methods in Aerodynamics*. von Karman Institute for Fluid Dynamics, 1994.

[11] A. Jameson. *Optimum Aerodynamic Design Using Control Theory, Computational Fluid Dynamics Review 1995*. Wiley, 1995.

[12] A. Jameson. Re-engineering the design process through computation. *AIAA paper 97-0641*, 35th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1997.

[13] A. Jameson and J.J. Alonso. Automatic aerodynamic optimization on distributed memory architectures. *AIAA paper 96-0409*, 34th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1996.

[14] A. Jameson, L. Martinelli, and N. Pierce. Optimum aerodynamic design using the navier-stokes equations. *AIAA paper 97-0101*, 35th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1997.

[15] A. Jameson and J. Reuther. Control theory based airfoil design using the Euler equations. *AIAA paper 94-4272*, 5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, FL, September 1994.

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

[17] J. Reuther, J.J. Alonso, M.J. Rimlinger, and A. Jameson. Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on parallel computers. *AIAA paper 96-4045*, 6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Bellevue, WA, September 1996.

[18] J. Reuther, S. Cliff, R. Hicks, and C.P. van Dam. Practical design optimization of wing/body configurations using the Euler equations. *AIAA paper 92-2633*, 1992.

[19] J. Reuther and A. Jameson. Control theory based airfoil design for potential flow and a finite volume discretization. *AIAA paper 94-0499*, 32nd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1994.

[20] J. Reuther and A. Jameson. Aerodynamic shape optimization of wing and wing-body configurations using control theory. *AIAA paper 95-0123*, 33rd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1995.

[21] J. Reuther and A. Jameson. A comparison of design variables for control theory based airfoil optimization. Technical report, 6th International Symposium on Computational Fluid Dynamics, Lake Tahoe, Nevada, September 1995.

[22] J. Reuther and A. Jameson. Supersonic wing and wing-body shape optimization using an adjoint formulation. Technical report, The Forum on CFD for Design and Optimization, (IMECE 95), San Francisco, California, November 1995.

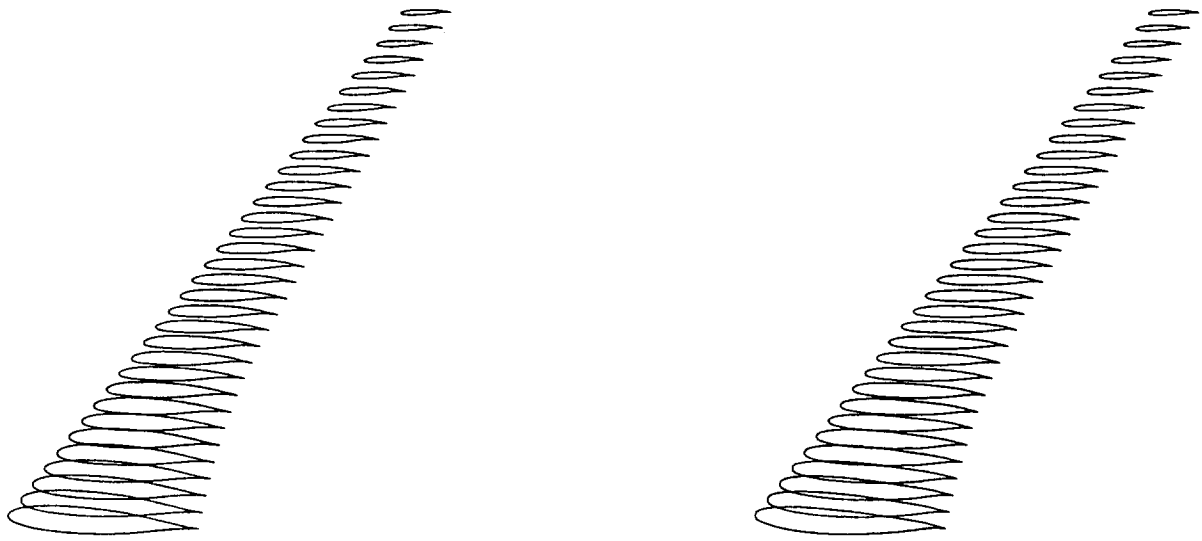
[23] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, and D. Saunders. Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. *AIAA paper 96-0094*, 34th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1996.

[24] J. Reuther, C.P. van Dam, and R. Hicks. Subsonic and transonic low-Reynolds-number airfoils with reduced pitching moments. *Journal of Aircraft*, 29:297–298, 1992.

[25] J. J. Reuther. Aerodynamic shape optimization using control theory. *Ph. D. Dissertation*, University of California, Davis, Davis, CA, June 1996.

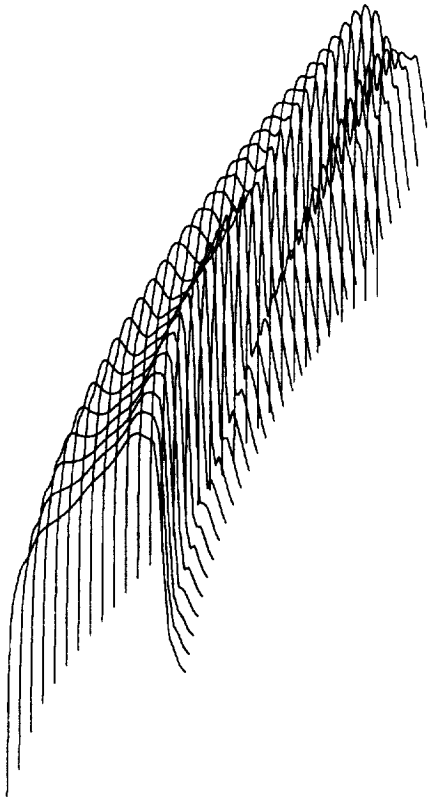
[26] J.P. Steinbrenner, J.R. Chawner, and C.L. Fouts. The GRIDGEN 3D multiple block grid generation system. Technical report, Flight Dynamics Laboratory, Wright Research and Development Center, Wright-Patterson Air Force Base, Ohio, July 1990.

[27] J.F. Thompson, Z.U.A Warsi, and C.W. Mastin. *Numerical Grid Generation, Foundations and Applications*. Elsevier Science Publishing Company, New York, NY, 1985.



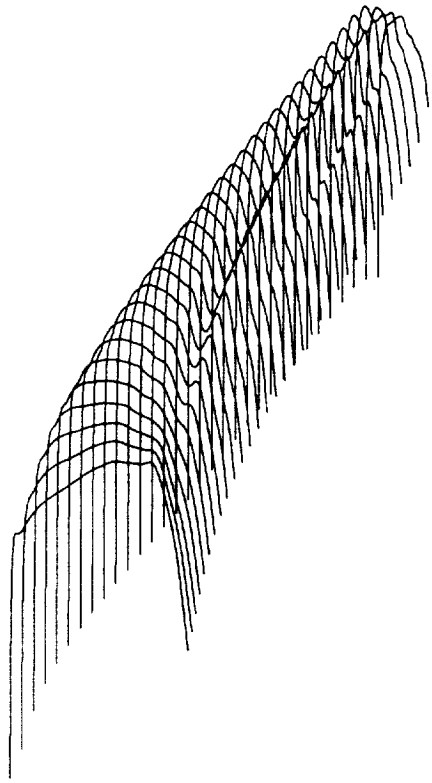
**1a: Initial Wing** **1b: 60 Design Iterations**

**Figure 1: Lifting Design Case, $M = 0.85$, Fixed Lift Mode. Multipoint Drag Reduction, Initial and Final Wings.**



UPPER SURFACE PRESSURE

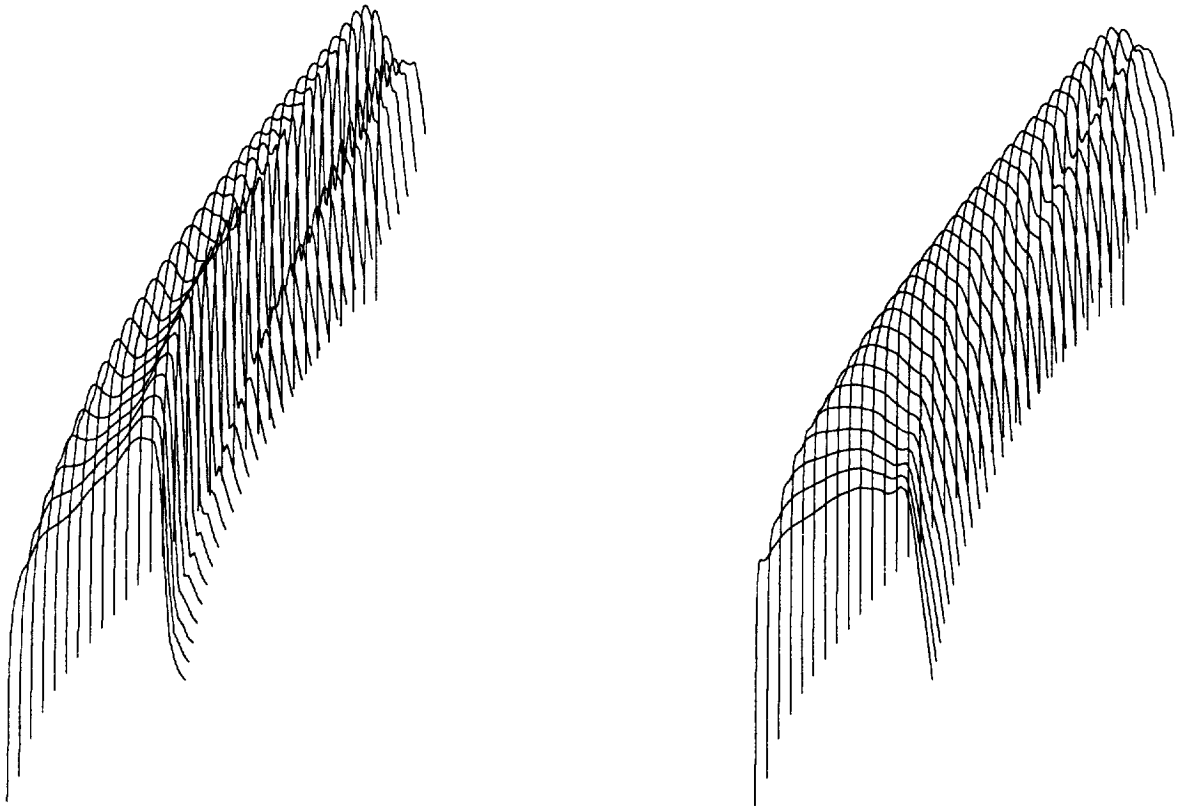
**2a: Initial Wing**
$C_L = 0.4995, C_D = 0.0156, \alpha = -1.471°$



UPPER SURFACE PRESSURE

**2b: 60 Design Iterations**
$C_L = 0.5000, C_D = 0.0114, \alpha = -0.933°$

**Figure 2: Multipoint Lifting Design Case: First Design Point, $M = 0.85$, $C_L = 0.5000$ Fixed Lift Mode, Drag Reduction.**



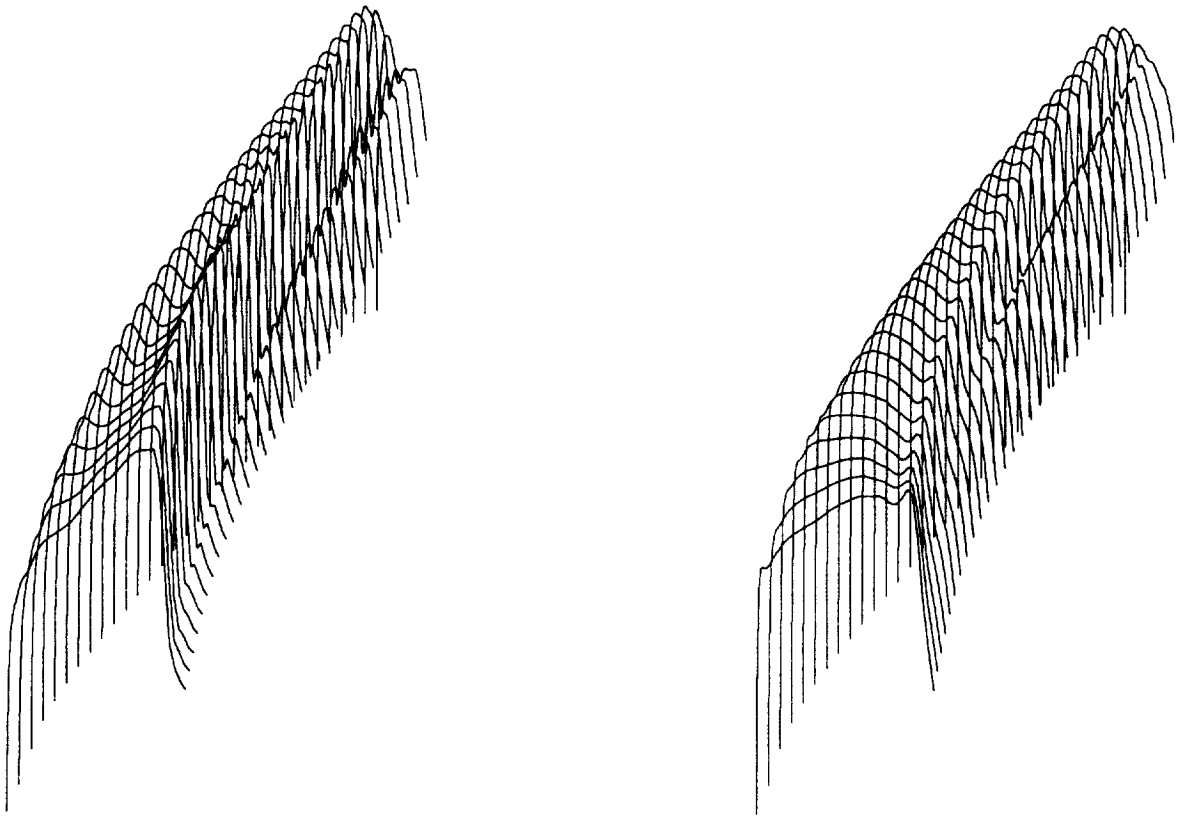
UPPER SURFACE PRESSURE

**3a: Initial Wing**
$C_L = 0.5243, C_D = 0.0173, \alpha = -1.300°$

UPPER SURFACE PRESSURE

**3b: 60 Design Iterations**
$C_L = 0.5251, C_D = 0.0123, \alpha = -0.770°$

**Figure 3: Multipoint Lifting Design Case: Second Design Point, $M = 0.85, C_L = 0.5250$ Fixed Lift Mode, Drag Reduction.**



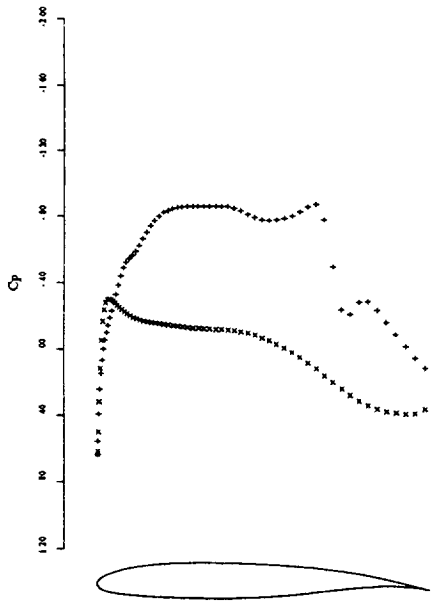
UPPER SURFACE PRESSURE

**4a: Initial Wing**
$C_L = 0.5490, C_D = 0.0191, \alpha = -1.128°$

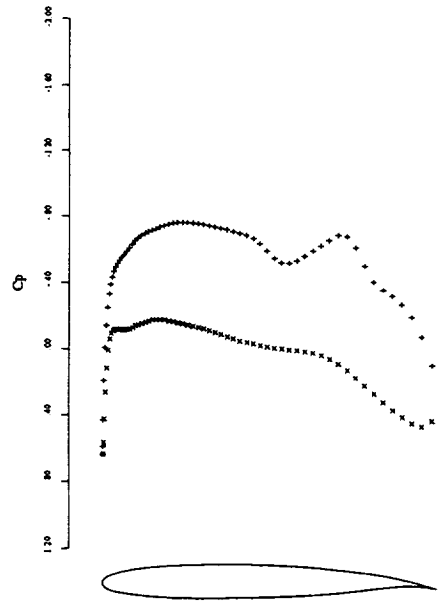
UPPER SURFACE PRESSURE

**4b: 60 Design Iterations**
$C_L = 0.5501, C_D = 0.0135, \alpha = -0.608°$

**Figure 4: Multipoint Lifting Design Case: Third Design Point, $M = 0.85, C_L = 0.5500$ Fixed Lift Mode, Drag Reduction.**

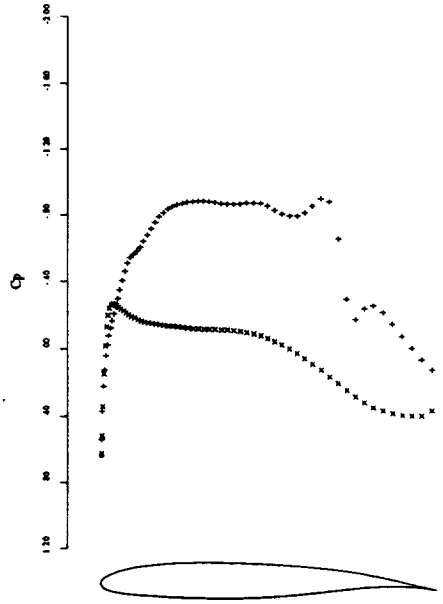


**5a: Initial Wing, Span Station** $z = 0.484$

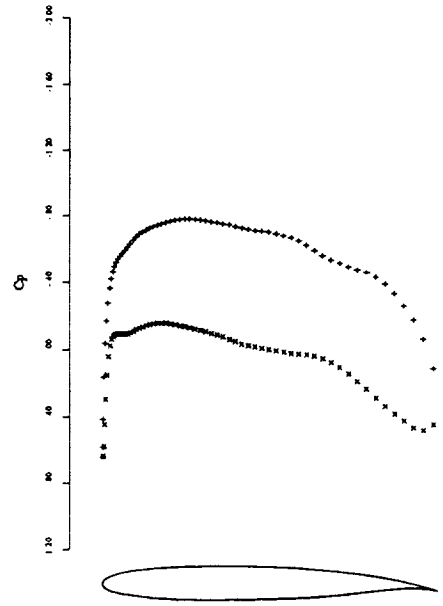


**5b: Design After 60 Cycles, Span Station** $z = 0.484$

**Figure 5: Multipoint Lifting Design Case, Design Point 1:**
**Fixed Lift Drag Minimization.**
$M = 0.85$, $C_L = 0.500$, **Pressure Coefficients of Initial and Final Wing Sections.**

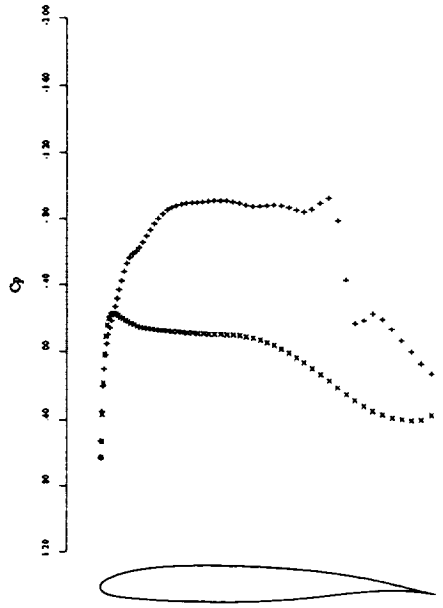


6a: Initial Wing, Span Station $z = 0.484$

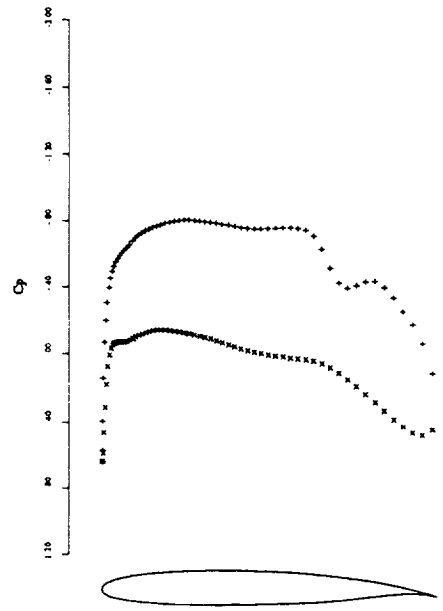


6b: Design After 60 Cycles, Span Station $z = 0.484$

**Figure 6: Multipoint Lifting Design Case, Design Point 2: Fixed Lift Drag Minimization.**
$M = 0.85$, $C_L = 0.525$, **Pressure Coefficients of Initial and Final Wing Sections.**

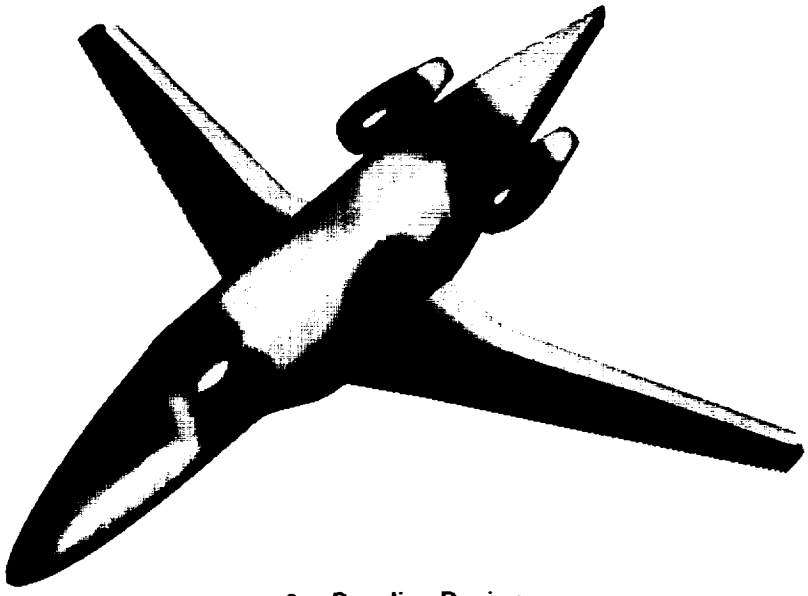


7a: Initial Wing, Span Station $z = 0.484$

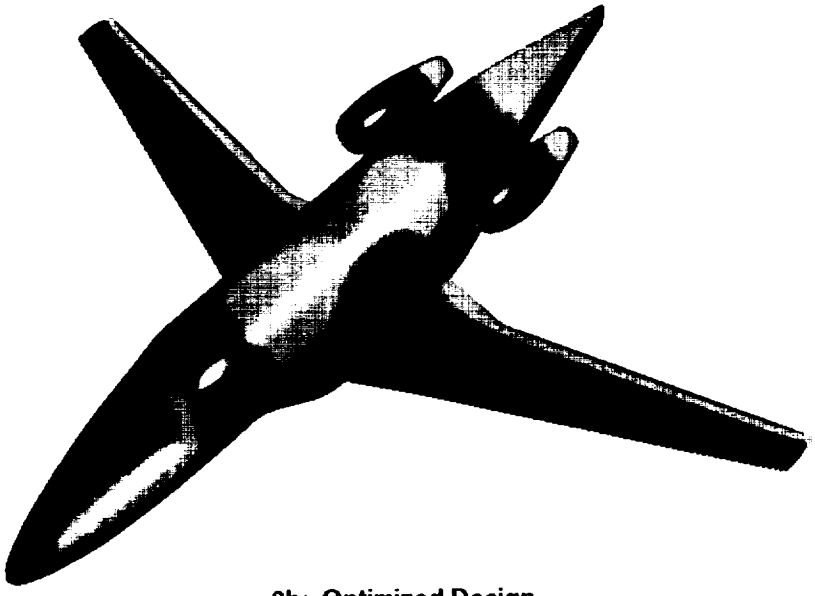


7b: Design After 60 Cycles, Span Station $z = 0.484$

**Figure 7: Multipoint Lifting Design Case, Design Point 3: Fixed Lift Drag Minimization.**
$M = 0.85$, $C_L = 0.550$, **Pressure Coefficients of Initial and Final Wing Sections.**

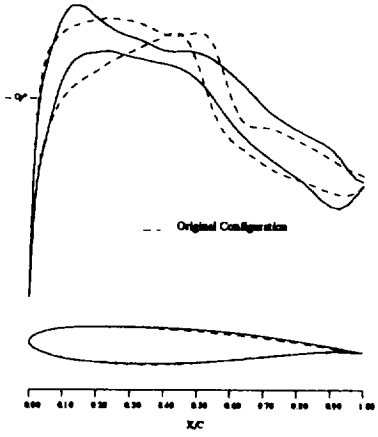


**8a: Baseline Design**

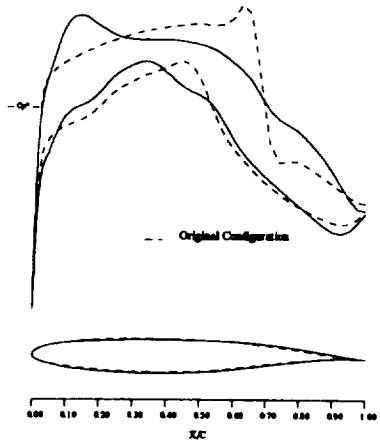


**8b: Optimized Design**

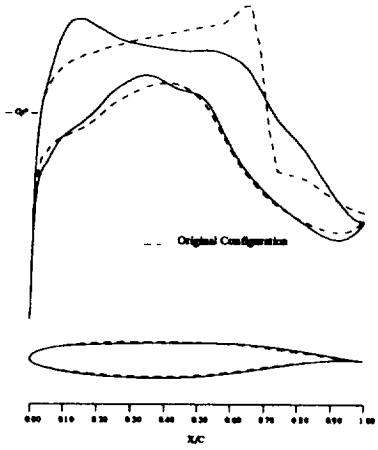
**Figure 8: Transonic Business Jet Configuration**
**Iso-Cp Contours.**
**Baseline and Optimized Designs.**
$M = 0.82,\ C_L = 0.30$



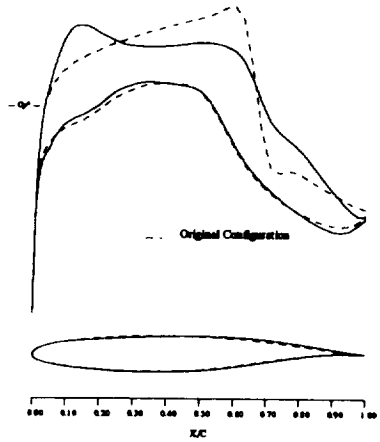

**9a: span station** $z = 0.190$




**9b: span station** $z = 0.475$

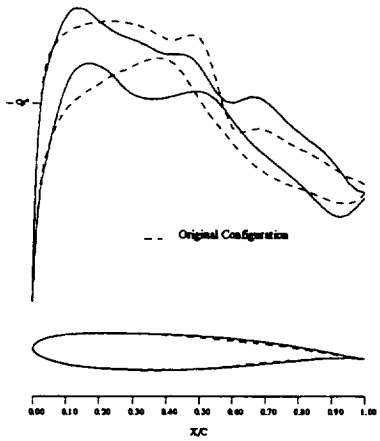



**9c: span station** $z = 0.665$

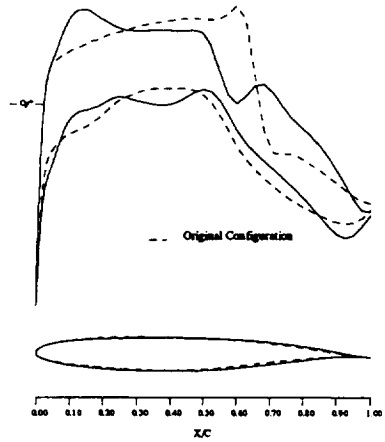



**9d: span station** $z = 0.856$

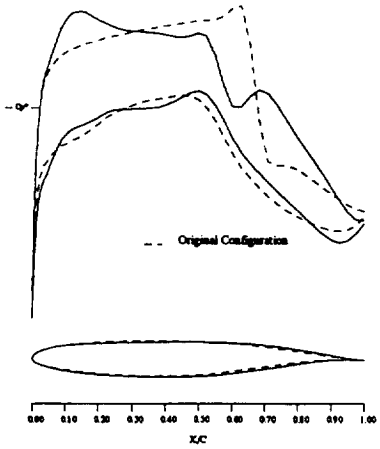
**Figure 9: Business Jet Configuration Configuration. Drag Minimization at Fixed Lift.**
**$M = 0.82, C_L = 0.3$**
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Initial Pressures**
**—, Pressures After 5 Design Cycles.**



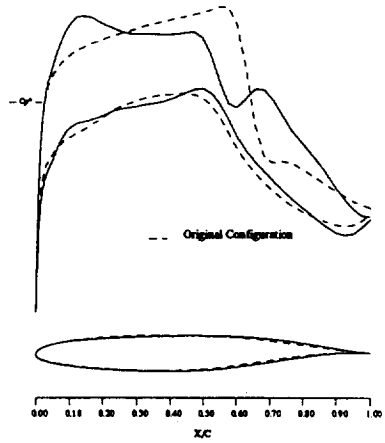

**10a: span station** $z = 0.190$




**10b: span station** $z = 0.475$

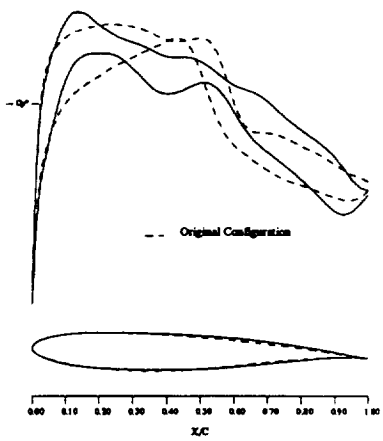



**10c: span station** $z = 0.665$

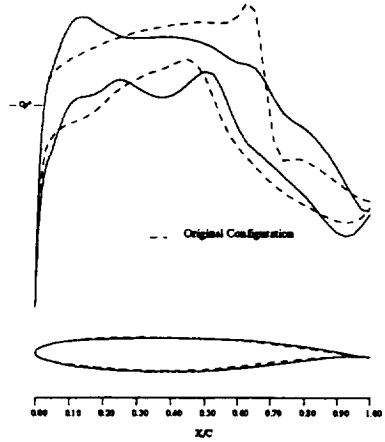



**10d: span station** $z = 0.856$

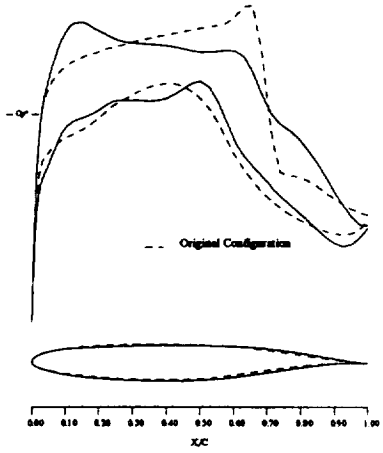
**Figure 10: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 1,** $M = 0.81, C_L = 0.35$
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Initial Pressures**
**—, Pressures After 5 Design Cycles.**



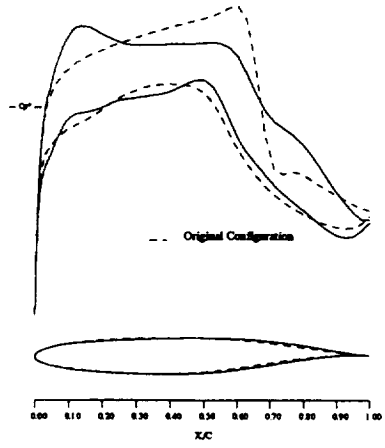

**11a: span station** $z = 0.190$




**11b: span station** $z = 0.475$

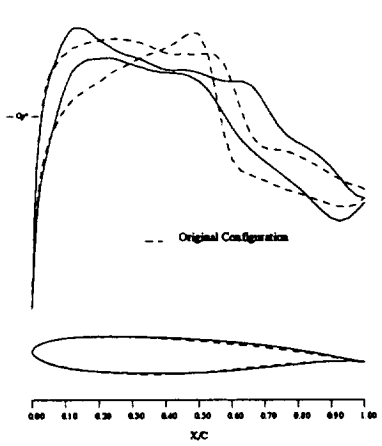



**11c: span station** $z = 0.665$

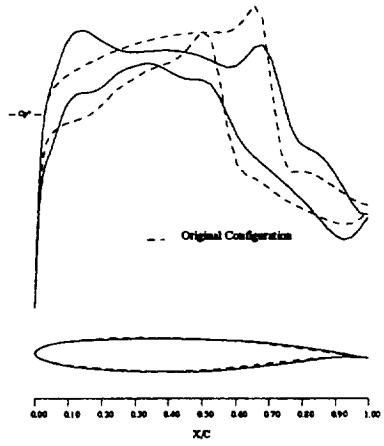



**11d: span station** $z = 0.856$

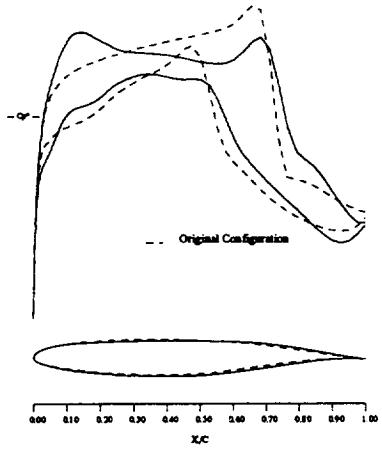
**Figure 11: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 2,** $M = 0.82$, $C_L = 0.30$
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Initial Pressures**
**—, Pressures After 5 Design Cycles.**



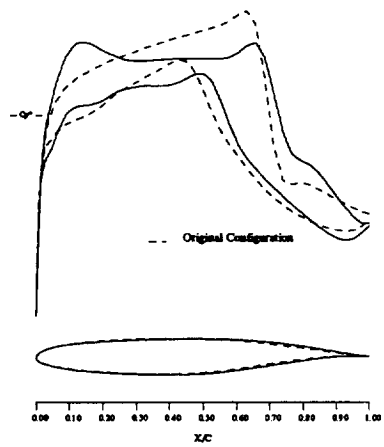

**12a: span station** $z = 0.190$




**12b: span station** $z = 0.475$

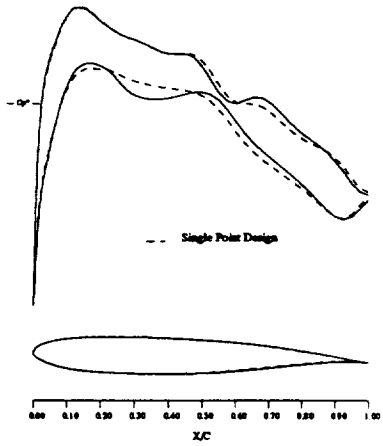



**12c: span station** $z = 0.665$

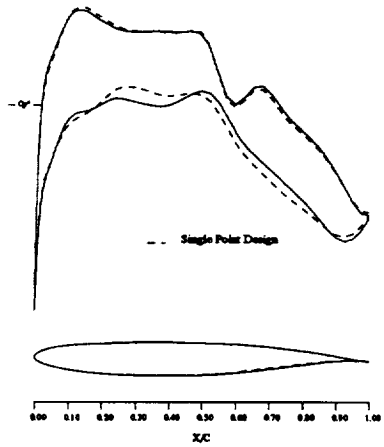



**12d: span station** $z = 0.856$

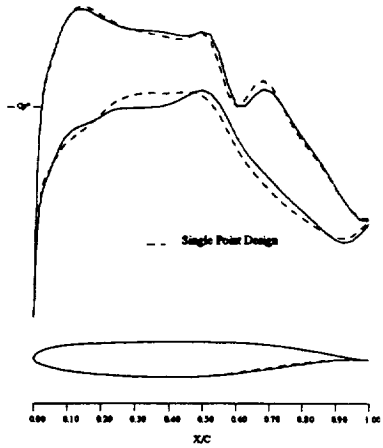
**Figure 12: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 3,** $M = 0.83$, $C_L = 0.25$
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Initial Pressures**
**—, Pressures After 5 Design Cycles.**



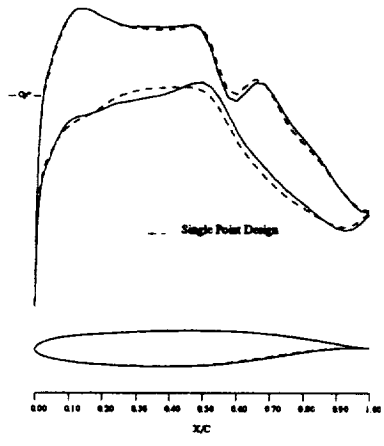

13a: span station $z = 0.190$




13b: span station $z = 0.475$

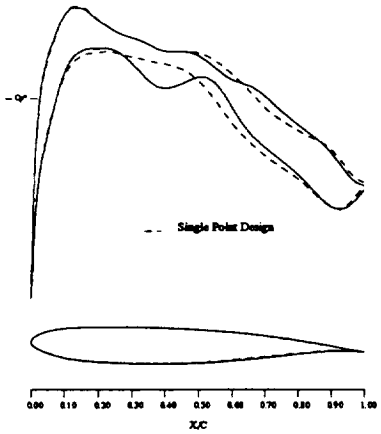



13c: span station $z = 0.665$

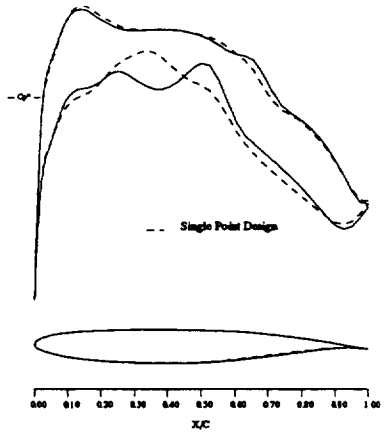



13d: span station $z = 0.856$

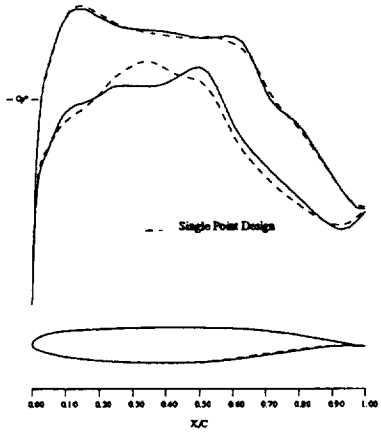
**Figure 13: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 1, $M = 0.81$, $C_L = 0.35$**
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Pressures for Design Test Case 1.**
**—, Pressures After 5 Design Cycles.**



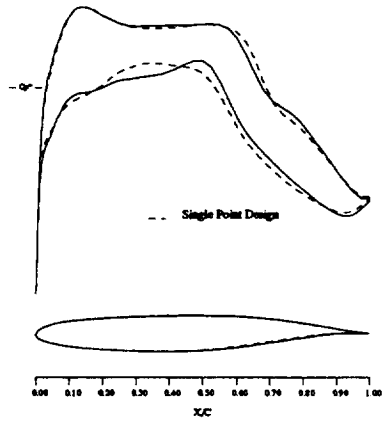

**14a: span station** $z = 0.190$




**14b: span station** $z = 0.475$

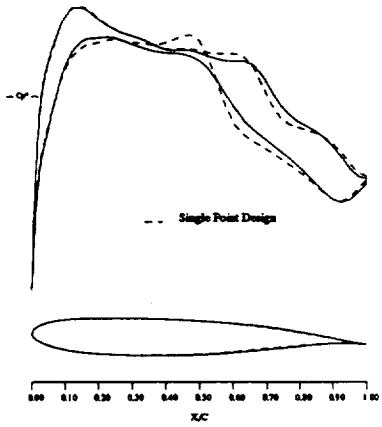



**14c: span station** $z = 0.665$

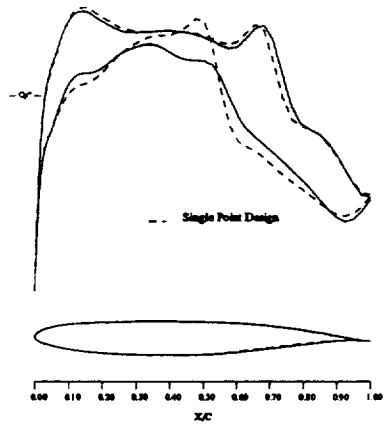



**14d: span station** $z = 0.856$

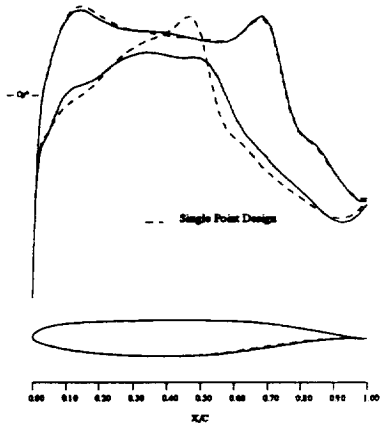
**Figure 14: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 2,** $M = 0.82$, $C_L = 0.30$
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Pressures for Design Test Case 1.**
**—, Pressures After 5 Design Cycles.**



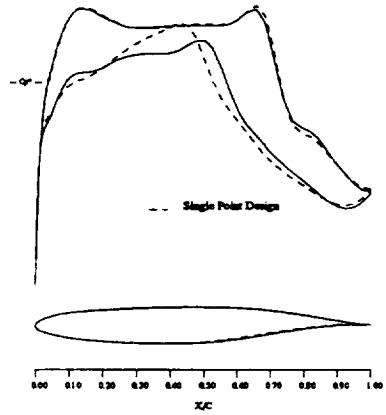

**15a: span station** $z = 0.190$




**15b: span station** $z = 0.475$

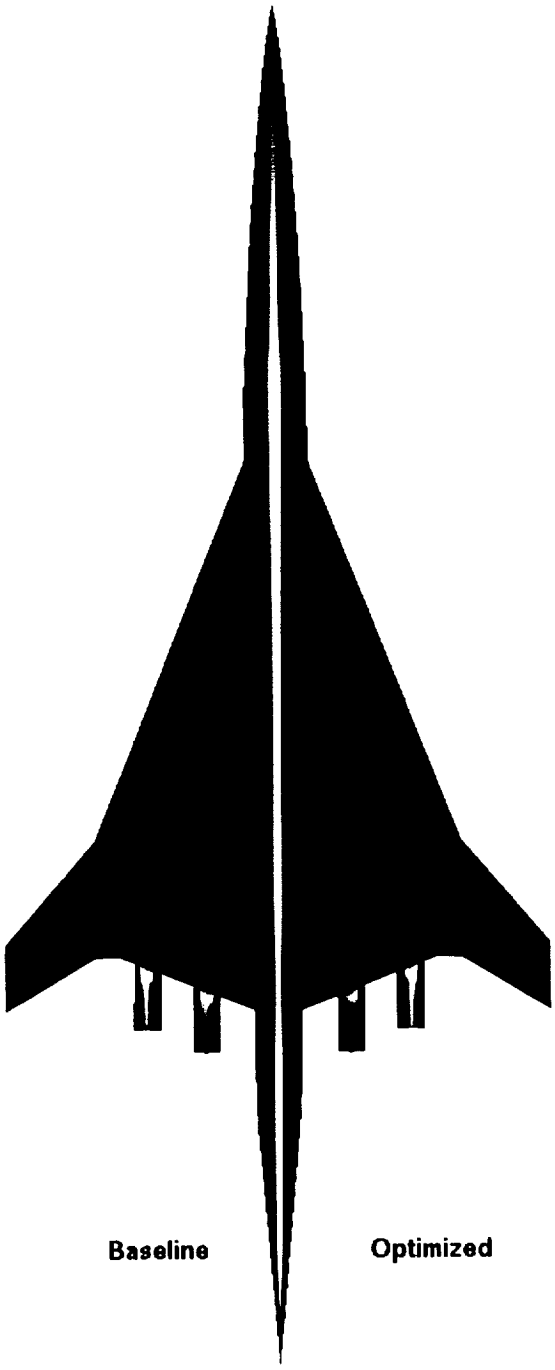



**15c: span station** $z = 0.665$

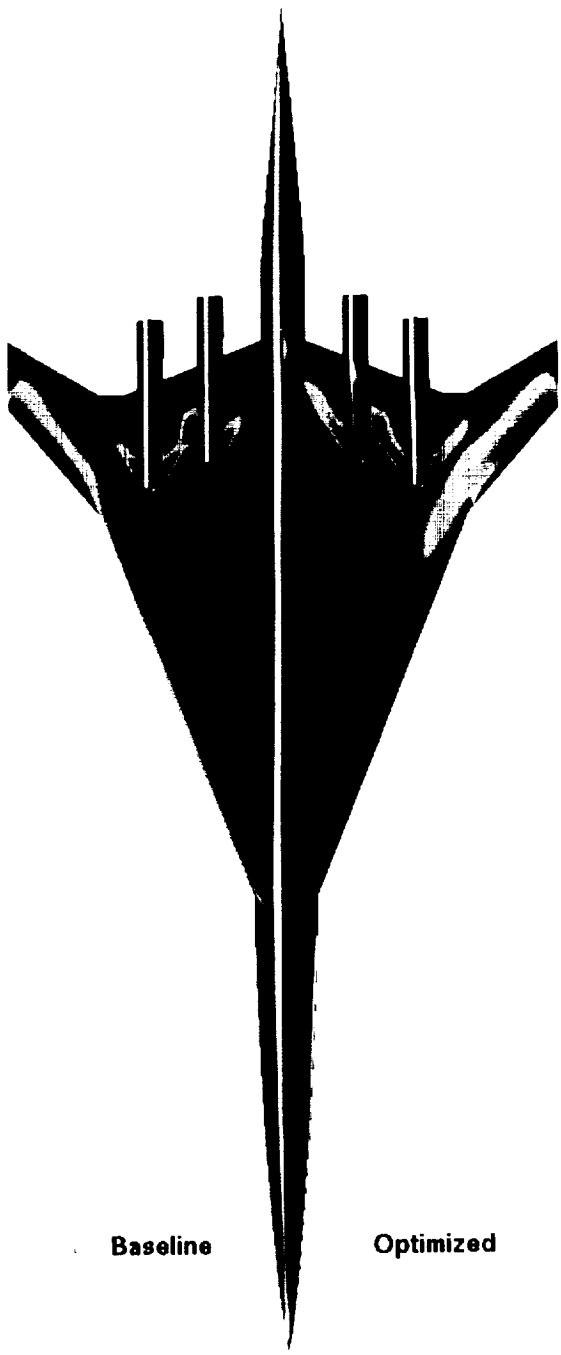



**15d: span station** $z = 0.856$

**Figure 15: Business Jet Configuration. Multipoint Drag Minimization at Fixed Lift.**
**Design Point 3,** $M = 0.83, C_L = 0.25$
**90 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Pressures for Design Test Case 1.**
**—, Pressures After 5 Design Cycles.**

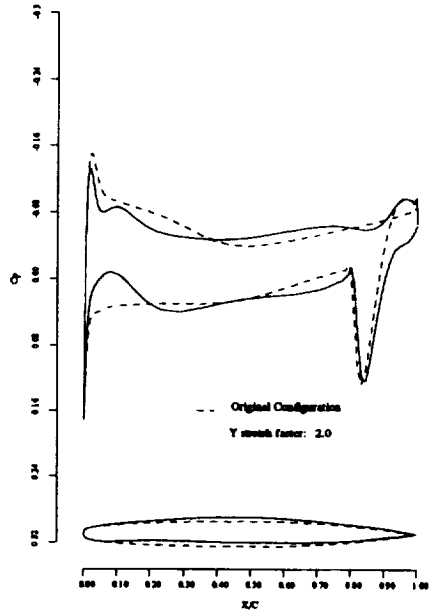



16a: Upper Surface

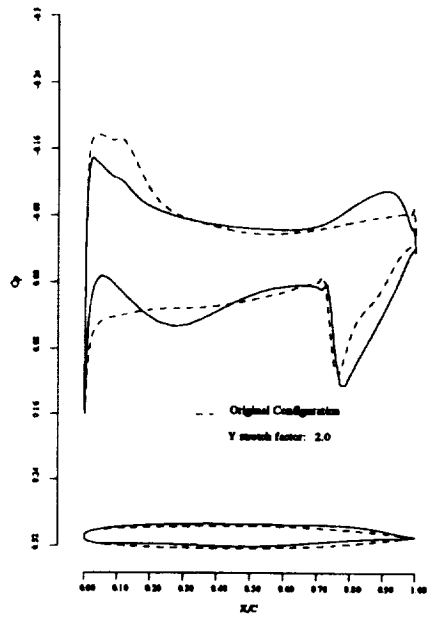



16b: Lower Surface

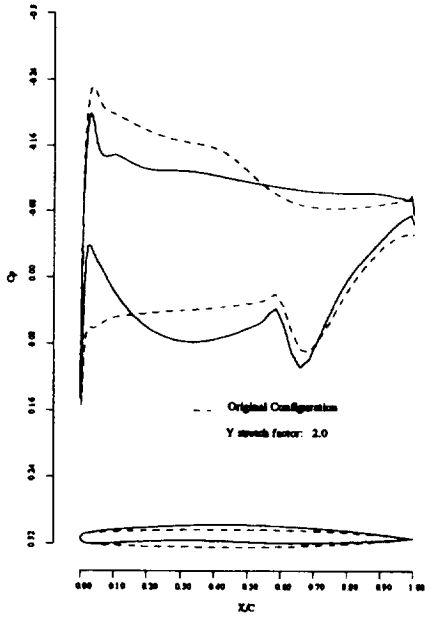
**Figure 16: Supersonic Transport Configuration.**
**Iso-Cp Contours on Upper and Lower Surfaces.**
**Baseline and Optimized Designs.**
**$M = 2.2, C_L = 0.105$**



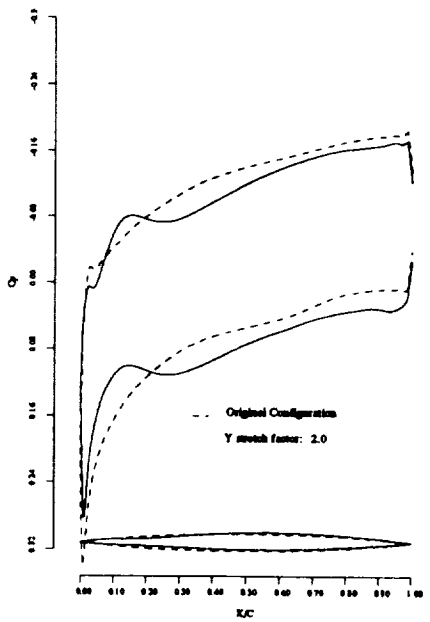

**17a: span station** $z = 0.194$




**17b: span station** $z = 0.387$




**17c: span station** $z = 0.581$




**17d: span station** $z = 0.775$

**Figure 17: Supersonic Transport Configuration. Drag Minimization at Fixed Lift.**
$M = 2.20, C_L = 0.105$
**144 Hicks-Henne variables. Spar Constraints Active.**
**- - -, Initial Pressures**
**—, Pressures After 5 Design Cycles.**