

Control Theory Based Airfoil Design Using the Euler Equations

Antony Jameson and James Reuther

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 2-13721 between NASA and the Universities Space Research Association (USRA).

Control Theory Based Airfoil Design using the Euler Equations

*A. Jameson**

Department of Mechanical and Aerospace Engineering
Princeton University
Princeton, New Jersey 08544, U.S.A.

and

J. Reuther†

Research Institute for Advanced Computer Science
Mail Stop T20G-5
NASA Ames Research Center
Moffett Field, California 94035, U.S.A.

Abstract

This paper describes the implementation of optimization techniques based on control theory for airfoil design. In our previous work [9, 10, 15] it was shown that control theory could be employed to devise effective optimization procedures for two-dimensional profiles by using the potential flow equation with either a conformal mapping or a general coordinate system. The goal of our present work is to extend the development to treat the Euler equations in two-dimensions by procedures that can readily be generalized to treat complex shapes in three-dimensions. Therefore, we have developed methods which can address airfoil design through either an analytic mapping or an arbitrary grid perturbation method applied to a finite volume discretization of the Euler equations. Here the control law serves to provide computationally inexpensive gradient information to a standard numerical optimization method. Results are presented for both the inverse problem and drag minimization problem.

Nomenclature

A_1, A_2 Cartesian flux Jacobians

b design variable

\hat{b} bandwidth of full Jacobian matrix

B bounding of flowfield domain at far field

c, s transformation metric

\bar{c} chord length

C bounding surface of flowfield domain on airfoil

C_1, C_2 transformed flux Jacobians

C_d coefficient of drag

C_l coefficient of lift

C_p coefficient of pressure

C_p^* coefficient of pressure for sonic flow

D flowfield domain

E total energy

f, g components of Cartesian fluxes

\hat{f}, \hat{g} f, g without pressure terms

F, G components of transformed fluxes

\hat{F}, \hat{G} F, G without pressure terms

\mathcal{F} control law, physical location of boundary

\mathcal{G} gradient of design space

$\hat{\mathcal{G}}$ projected \mathcal{G} into admissible space

h modulus of conformal mapping transformation

H total enthalpy

I cost function

J Jacobian of generalized transformation

K grid transformation matrix

K_s K at the profile surface

K_1, K_2 grid transformation matrices

m number of flowfield evaluations per line search

M local Mach number

M_∞ Mach number at infinity

n number of design variables

\hat{n} number of grid points

\hat{n}_1, \hat{n}_2 components of a unit vector normal

p pressure

p_d desired pressure

q speed

q_∞ speed at infinity

r radial component in mapped plane

R generic governing equation for flowfield

\mathcal{R} normalized arc length in η

s arc length along airfoil surface

*James S. McDonnell Distinguished University Professor
of Aerospace Engineering, AIAA Fellow

†Student Member AIAA

© Copyright ©1994 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved

S	surface displacement in mapped plane
t	time
t_1, t_2	exponents on basis functions
u, v	Cartesian velocity components
U, V	contravariant velocity components
w	state vector of flowfield unknowns
x, y	Cartesian coordinates
ξ, η	body fitted coordinates
z	physical plane
Z	modified boundary condition for ψ
α	angle of attack
β	smoothing parameter
δ	variational
γ	ratio of specific heats
λ	distance along search direction
λ_1, λ_2	smoothing parameters
Λ	distance along search direction
Ω_1, Ω_2	cost function weighting factors
ψ	vector co-state variable
ρ	density
ρ_∞	freestream density
σ	conformally mapped plane
θ	angle around circle

Formulation of the design problem as a control problem

The final efficiency and success of a design depend on complex multi-disciplinary trade-offs between factors such as aerodynamic efficiency, structural weight, stability and control, and the volume required to contain fuel and payload. Typically a design is finalized after numerous iterations, cycling between the disciplines. While full multidisciplinary optimization is the ultimate goal, a necessary intermediate step is the development of efficient methods for aerodynamic shape optimization. Early investigations of aerodynamic optimization [16] relied on brute force methods, in which the influence of each design parameter was estimated by making a small change in that parameter and recalculating the flow. With a large number of parameters this becomes extremely expensive.

Alternatively, it has been recognized that the designer has an idea of the kind of pressure distribution that will lead to the desired performance. Thus, it is useful to consider the inverse problem of calculating the shape that will lead to a given pressure distribution. This method has the advantage that only one flow solution is required to obtain the desired design. Unfortunately, unless the pressure distribution satisfies certain constraints a physically realizable shape may not necessarily exist. Thus the problem must be very carefully formulated.

The difficulty that the objective may be unattainable can be circumvented by regarding the design problem as a control problem in which the control is the shape of the boundary. A variety of alternative formulations of the design problem can then be

treated systematically within the framework of the mathematical theory for control of systems governed by partial differential equations [12]. This approach to optimal aerodynamic design was introduced by Jameson [9, 10], who examined the design problem for compressible flow with shock waves, and devised adjoint equations to determine the gradient for both potential flow and also flows governed by the Euler equations. More recently Ta'asan, Kuruvila, and Salas, implemented a one shot approach in which the constraint represented by the flow equations need only be satisfied by the final converged solution [20]. Pironneau has also studied the use of control theory for optimum shape design of systems governed by elliptic equations [14], and adjoint methods have also been used by Baysal and Eleshaky [2].

For flow about an airfoil or wing, the aerodynamic properties which define the cost function are functions of the flow-field variables (w) and the physical location of the boundary, which may be represented by the function \mathcal{F} , say. Then

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

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

$$\delta I = \frac{\partial I^T}{\partial w} \delta w + \frac{\partial I^T}{\partial \mathcal{F}} \delta \mathcal{F}, \quad (1)$$

in the cost function. As pointed out by Baysal and Eleshaky [2] each term in (1), except for δw , can be easily obtained. $\frac{\partial I}{\partial w}$ and $\frac{\partial I}{\partial \mathcal{F}}$ can be obtained directly without a flowfield evaluation since they are partial derivatives. $\delta \mathcal{F}$ can be determined by either working out the exact analytical values from a mapping, or by successive grid generation for each design variable, so long as this cost is significantly less than the cost of the flow solution. Brute force methods evaluate the gradient by making a small change in each design variable separately, and then recalculate both the grid and flow-field variables. This requires a number of additional flow calculations equal to the number of design variables. Using control theory, the governing equations of the flowfield are introduced as a constraint in such a way that the final expression for the gradient does not require multiple flow solutions. In order to achieve this δw must be eliminated from (1). The governing equation R expresses the dependence of w and \mathcal{F} within the flowfield domain D ,

$$R(w, \mathcal{F}) = 0,$$

Thus δw is determined from the equation

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

Next, introducing a Lagrange Multiplier ψ , we have

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

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

Choosing ψ to satisfy the adjoint equation

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

the first term is eliminated, and we find that

$$\delta I = \mathcal{G}^T \delta \mathcal{F} \quad (4)$$

where

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

The advantage is that (4) is independent of δw , with the result that 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, the cost differential between one adjoint solution and the large number of flowfield evaluations required to determine the gradient by brute force becomes compelling. Instead of introducing a Lagrange multiplier, ψ , one can solve (2) for δw as

$$\delta w = - \left[\frac{\partial R}{\partial w} \right]^{-1} \left[\frac{\partial R}{\partial \mathcal{F}} \right] \delta \mathcal{F}, \quad (5)$$

and insert the result in (1). This is the implicit gradient approach, which is essentially equivalent to the control theory approach, as has been pointed out by Shubin and Frank [18, 19]. In any event, there is an advantage in determining the gradient \mathcal{G} by the solution of the adjoint equation.

Once equation (4) is established, an improvement can be made by with a shape change

$$\delta \mathcal{F} = -\lambda \mathcal{G}$$

where λ is positive, and small enough that the first variation is an accurate estimate of δI . Then

$$\delta I = -\lambda \mathcal{G}^T \mathcal{G} < 0$$

After making such a modification, the gradient can be recalculated and the process repeated to follow a path of steepest descent until a minimum is reached. In order to avoid violating constraints, such as a minimum acceptable wing thickness, the gradient may be projected into the allowable subspace within

which the constraints are satisfied. In this way procedures can be devised which must necessarily converge at least to a local minimum. The efficiency may be improved by performing line searches to find the minimum in a search direction defined by the negative gradient, and also by the use of more sophisticated descent methods such as conjugate gradient or quasi-Newton algorithms. There is the possibility of more than one local minimum, but in any case the method will lead to an improvement over the original design. Furthermore, unlike the traditional inverse algorithms, any measure of performance can be used as the cost function.

Steps (1 – 4) may be applied to the discrete equations which approximate the governing differential equations. In the present method they are applied instead directly to the differential equations, and the resulting adjoint differential equation is then discretized in a manner similar to the discretization of the flow equations. The former approach is now gaining favor in the work of Newman and Taylor *et al.* [13, 11] and also Baysal, Eleshaky and Burgreen [1, 2, 3, 4]. The two methods can be very similar, depending upon the discretization of (3) or (5). The current method has an advantage in that the discretization and iteration scheme used to solve the flowfield system can also be applied directly to the adjoint system. Therefore the robust iteration algorithms and convergence acceleration techniques that have been matured for CFD algorithms can be directly ported for the solution of the adjoint system. Methods applying (1 – 4) to the discrete system often resort to matrix inversion methods to solve (3) or (5). While direct inversion techniques can be robust and reliable they suffer when the number of points becomes large because the operation count grows as $O(\hat{n}\hat{b}^2)$, where \hat{n} is the number of unknowns and \hat{b} is the bandwidth which can approach $O(\hat{n})$. When the number of mesh points becomes large, especially in the case of three-dimensional problems, the $O(\hat{n})$ operational counts of explicit iteration schemes used in CFD can significantly reduce the required time to solve the adjoint system.

In our previous application of control theory to optimal aerodynamic shape design a successful numerical implementation was demonstrated using the potential flow equation, with either a conformal mapping or a finite volume discretization. In this work, the Euler equations are employed instead of the potential flow equation to provide a more accurate representation of flows containing strong shock waves and entropy variations. Again both an analytic mapping and general finite volume discretization are explored as possible vehicles for the implementation of design via control theory. This is intended to be a precursor for the three-dimensional problem, where the geometry may be very complex and both shocks and vortex roll-up must be considered. The

treatment of viscous flows with the Navier Stokes equations is a natural future step.

Design of airfoils using the Euler equations

This section develops the general application of control theory for aerodynamic shape optimization using the Euler equations. Consider the case for compressible flow over an airfoil. In the absence of separation and other strong viscous effects, the flow is well approximated by the Euler equations. In contrast to our previous implementations which relied on the isentropic potential equation [15], here strong inviscid shocks are modeled correctly with entropy production. Consider the flow in a domain D . The profile defines the inner boundary C , while the outer boundary B is assumed to be distant from the profile. Let p , ρ , u , v , E and H denote the pressure, density, Cartesian velocity components, total energy and total enthalpy. For a perfect gas

$$p = (\gamma - 1) \rho \left\{ E - \frac{1}{2} (u^2 + v^2) \right\} \quad (6)$$

and

$$\rho H = \rho E + p, \quad (7)$$

where γ is the ratio of the specific heats. The Euler equations may then be written as

$$\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0 \quad \text{in } D, \quad (8)$$

where x and y are Cartesian coordinates, t is the time coordinate and

$$w = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad f = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho u H \end{bmatrix}, \quad g = \begin{bmatrix} \rho v \\ \rho v^2 + p \\ \rho v H \end{bmatrix}. \quad (9)$$

The coordinate transformations may be defined by the transformation matrix

$$K = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix},$$

with ξ and η representing the computational coordinate system and the Jacobian

$$J = \det(K) = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi}.$$

Introduce contravariant velocity components

$$\begin{bmatrix} U \\ V \end{bmatrix} = K^{-1} \begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{J} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}.$$

The Euler equations can be written in divergence form as

$$\frac{\partial W}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial G}{\partial \eta} = 0 \quad \text{in } D, \quad (10)$$

with

$$W = J \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad F = J \begin{bmatrix} \rho U \\ \rho U u + \frac{\partial \xi}{\partial x} p \\ \rho U v + \frac{\partial \xi}{\partial y} p \\ \rho U H \end{bmatrix}, \quad G = J \begin{bmatrix} \rho V \\ \rho V u + \frac{\partial \eta}{\partial x} p \\ \rho V v + \frac{\partial \eta}{\partial y} p \\ \rho V H \end{bmatrix}. \quad (11)$$

Assume now that the computational coordinate system conforms to the airfoil section in such a way that the surface C is represented by $\eta = 0$. Then the flow is determined as the steady state solution of the equation (11) subject to the flow tangency condition

$$V = 0 \quad \text{on } C. \quad (12)$$

At the far field boundary B , conditions are specified for incoming waves, while outgoing waves are determined by the solution.

As a first example of the use of control theory, consider the case of the inverse problem where the cost function may be defined as

$$I = \frac{1}{2} \int_C (p - p_d)^2 ds = \frac{1}{2} \int_C (p - p_d)^2 \left(\frac{ds}{d\xi} \right) d\xi,$$

where p_d is the desired pressure. The design problem is now treated as a control problem where the control function is the airfoil shape, which is to be chosen to minimize I subject to the constraints defined by the flow equations (10–12). A variation in the shape will cause a variation δp in the pressure in addition to a variation in the geometry and consequently the

variation in the cost function becomes

$$\begin{aligned}\delta I &= \int_C (p - p_d) \delta p \left(\frac{ds}{d\xi} \right) d\xi \\ &+ \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi. \quad (13)\end{aligned}$$

Since p depends on w through the equation of state (8–9), the variation δp can be determined from the variation δw . Define the Jacobian matrices

$$A_1 = \frac{\partial f}{\partial w}, \quad A_2 = \frac{\partial g}{\partial w}, \quad C_i = \sum_j J K_{ij}^{-1} A_j. \quad (14)$$

Then the equation for δw in the steady state becomes

$$\frac{\partial}{\partial \xi} (\delta F) + \frac{\partial}{\partial \eta} (\delta G) = 0, \quad (15)$$

where

$$\begin{aligned}\delta F &= C_1 \delta w + \delta \left(J \frac{\partial \xi}{\partial x} \right) f + \delta \left(J \frac{\partial \xi}{\partial y} \right) g \\ \delta G &= C_2 \delta w + \delta \left(J \frac{\partial \eta}{\partial x} \right) f + \delta \left(J \frac{\partial \eta}{\partial y} \right) g.\end{aligned}$$

Now, multiplying by a vector costate variable ψ and integrating over the domain we have

$$\int_D \psi^T \left(\frac{\partial \delta F}{\partial \xi} + \frac{\partial \delta G}{\partial \eta} \right) d\xi d\eta = 0.$$

If ψ is differentiable this may be integrated by parts to give

$$\begin{aligned}&\int_D \left(\frac{\partial \psi^T}{\partial \xi} \delta F + \frac{\partial \psi^T}{\partial \eta} \delta G \right) d\xi d\eta = \\ &\int_B (\bar{n}_1 \psi^T \delta F + \bar{n}_2 \psi^T \delta G) d\xi \\ &+ \int_C (\bar{n}_1 \psi^T \delta F + \bar{n}_2 \psi^T \delta G) d\xi,\end{aligned}$$

where \bar{n}_i are components of a unit vector normal to the boundary. No boundary integrals appear in the η direction because the mesh is assumed to be of O -type, with the result that the solution is periodic in the ξ coordinate thereby cancelling the η boundary integrals. Thus the variation in the cost function may now be written

$$\begin{aligned}\delta I &= \int_C (p - p_d) \delta p \left(\frac{ds}{d\xi} \right) d\xi \\ &+ \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi \\ &+ \int_D \left(\frac{\partial \psi^T}{\partial \xi} \delta F + \frac{\partial \psi^T}{\partial \eta} \delta G \right) d\xi d\eta \\ &- \int_B (\bar{n}_1 \psi^T \delta F + \bar{n}_2 \psi^T \delta G) d\xi \\ &- \int_C (\bar{n}_1 \psi^T \delta F + \bar{n}_2 \psi^T \delta G) d\xi.\end{aligned}$$

On the profile $\bar{n}_1 = 0$ and $\bar{n}_2 = -1$. It follows from equation (12) that

$$\delta G = J \begin{bmatrix} 0 \\ \frac{\partial \eta}{\partial x} \delta p \\ \frac{\partial \eta}{\partial y} \delta p \\ 0 \end{bmatrix} + p \begin{bmatrix} 0 \\ \delta \left(J \frac{\partial \eta}{\partial x} \right) \\ \delta \left(J \frac{\partial \eta}{\partial y} \right) \\ 0 \end{bmatrix}. \quad (16)$$

Suppose now that ψ is the steady state solution of the adjoint equation

$$\frac{\partial \psi}{\partial t} - C_1^T \frac{\partial \psi}{\partial \xi} - C_2^T \frac{\partial \psi}{\partial \eta} = 0 \quad \text{in } D. \quad (17)$$

At the outer boundary incoming characteristics for ψ correspond to outgoing characteristics for δw . Consequently, one can choose boundary conditions for ψ such that

$$\bar{n}_i \psi^T C_i \delta w = 0.$$

Then if the coordinate transformation is such that $\delta(JK^{-1})$ is negligible in the far field, the only remaining boundary term is

$$\int_C \psi^T \delta G \frac{ds}{d\xi} d\xi.$$

Thus by letting ψ satisfy the boundary condition,

$$J \left(\psi_2 \frac{\partial \eta}{\partial x} + \psi_3 \frac{\partial \eta}{\partial y} \right) = -(p - p_d) \frac{ds}{d\xi} \quad \text{on } C, \quad (18)$$

we find finally that

$$\begin{aligned}\delta I &= \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi \\ &+ \int_D \frac{\partial \psi^T}{\partial \xi} \left(\delta \left(J \frac{\partial \xi}{\partial x} \right) f + \delta \left(J \frac{\partial \xi}{\partial y} \right) g \right) \\ &\quad + \frac{\partial \psi^T}{\partial \eta} \left(\delta \left(J \frac{\partial \eta}{\partial x} \right) f + \delta \left(J \frac{\partial \eta}{\partial y} \right) g \right) d\xi d\eta \\ &+ \int_C \left\{ \psi_2 \delta \left(J \frac{\partial \eta}{\partial x} \right) + \psi_3 \delta \left(J \frac{\partial \eta}{\partial y} \right) \right\} p d\xi \\ &= \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi \\ &+ \int_D \frac{\partial \psi^T}{\partial \xi} \left(\delta \left(\frac{\partial y}{\partial \eta} \right) f - \delta \left(\frac{\partial x}{\partial \eta} \right) g \right) \\ &\quad + \frac{\partial \psi^T}{\partial \eta} \left(-\delta \left(\frac{\partial y}{\partial \xi} \right) f + \delta \left(\frac{\partial x}{\partial \xi} \right) g \right) d\xi d\eta \\ &+ \int_C \left\{ \psi_2 \delta \left(-\frac{\partial y}{\partial \xi} \right) + \psi_3 \delta \left(\frac{\partial x}{\partial \xi} \right) \right\} p d\xi. \quad (19)\end{aligned}$$

If the flow is subsonic, this procedure should converge toward the desired pressure distribution since the solution will remain smooth, and no unbounded

derivatives will appear. If, however, the flow is transonic, one must allow for the appearance of shock waves in the trial solutions, even if p_d is smooth. In such instances $p - p_d$ is not differentiable. This difficulty can be circumvented by a more sophisticated choice of the cost function. Consider the choice

$$I = \frac{1}{2} \int_C \left(\lambda_1 Z^2 + \lambda_2 \left(\frac{dZ}{d\xi} \right)^2 \right) \frac{ds}{d\xi} d\xi,$$

where λ_1 and λ_2 are parameters, and the periodic function $Z(\xi)$ satisfies the equation

$$\lambda_1 Z - \lambda_2 \frac{d^2 Z}{d\xi^2} = p - p_d. \quad (20)$$

Then,

$$\begin{aligned} \delta I &= \int_C \left(\lambda_1 Z \delta Z + \lambda_2 \frac{dZ}{d\xi} \frac{d}{d\xi} \delta Z \right) \frac{ds}{d\xi} d\xi \\ &= \int_C Z \left(\lambda_1 \delta Z - \lambda_2 \frac{d^2}{d\xi^2} \delta Z \right) \frac{ds}{d\xi} d\xi \\ &= \int_C Z \delta q \frac{ds}{d\xi} d\xi. \end{aligned}$$

Thus, Z replaces $p - p_d$ with a corresponding modification to the boundary condition for the adjoint equations.

The following two sections develop alternative implementations of the general method discussed thus far. The differences between the two lie in the way in which the control modifies the mesh on which both the flow and the adjoint equations are solved. The first method relies on an analytic mapping of the mesh and directly uses the value of the mapping at the surface as the control. This implies that each surface mesh point becomes a design variable. The second method allows for an arbitrary numerically generated initial mesh. This is used as a template for all subsequent meshes, which are created by a grid perturbation technique. A set of design variables spanning a space of allowable airfoils is then chosen to assure smooth variations in the geometry. Thus these design variables become the control, and no dependence upon transformations is necessary.

Design Using an Analytic Mapping

A convenient way to treat an airfoil is to use a conformal mapping of the profile in the z plane to a near circle in the σ plane, followed by shearing of the radial coordinate to make the system boundary conforming. Polar coordinates are introduced in the mapped plane σ . When mapped back to the physical plane this gives a smooth, nearly orthogonal grid. We can now specialize our generalized design procedure to treat this grid system. Define the first conformal mapping from z to σ by letting

the derivative of the mapping function be

$$\frac{dz}{d\sigma} = h e^{i\beta}.$$

Now using polar coordinates r , and θ in the σ plane, the first transformation matrix is

$$K_1 = \begin{bmatrix} x_\theta & x_r \\ y_\theta & y_r \end{bmatrix} = h \begin{bmatrix} rs & c \\ -cs & s \end{bmatrix}$$

and we can define contravariant velocities

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} s & -c \\ c & s \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

where

$$s = \sin(\beta - \theta), \quad c = \cos(\beta - \theta).$$

The Euler equations can now be represented in the σ plane as

$$\frac{\partial(\rho h^2 W)}{\partial t} + \frac{\partial(hF)}{\partial \theta} + \frac{\partial(\rho h G)}{\partial r} = 0 \quad \text{in } D, \quad (21)$$

where

$$\begin{aligned} W &= \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \\ F &= \begin{bmatrix} \rho U \\ \rho U u + sp \\ \rho U v - cp \\ \rho U H \end{bmatrix}, \quad G = \begin{bmatrix} \rho V \\ \rho V u + cp \\ \rho V v + sp \\ \rho V H \end{bmatrix}. \end{aligned} \quad (22)$$

Now let the final computational coordinates be defined by a radial shearing transformation

$$\theta = \xi, \quad r = \eta + S(\xi)$$

and the transformation matrix

$$K_2 = \begin{bmatrix} \frac{\partial \theta}{\partial \xi} & \frac{\partial \theta}{\partial \eta} \\ \frac{\partial r}{\partial \xi} & \frac{\partial r}{\partial \eta} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{\partial S}{\partial \xi} & 1 \end{bmatrix}, \quad J_2 = 1.$$

Now we can identify the complete transformation matrix as

$$K = K_1 K_2 = h \begin{bmatrix} rs + S_\xi c & c \\ -rc + S_\xi s & s \end{bmatrix},$$

while the fluxes are

$$\begin{aligned} hF &= h(sf - cg) \\ &= y_\eta f - x_\eta g \end{aligned}$$

and

$$\begin{aligned} h[(\eta + S)G - S_\xi F] &= h[(\eta + S)c - S_\xi s]f \\ &\quad + h[(\eta + S)s + S_\xi c]g \\ &= x_\xi g - y_\xi f. \end{aligned}$$

Thus the Euler equations assume the form

$$\begin{aligned} &\frac{\partial}{\partial t}((\eta + S)h^2W) \\ &+ \frac{\partial}{\partial \xi}(hF) \\ &+ \frac{\partial}{\partial \eta}(h(\eta + S)G - hS_\xi F) = 0, \end{aligned}$$

while the surface tangency condition on the velocity becomes

$$x_\xi v - y_\xi u = h[(\eta + s)V - s_\xi U] = 0.$$

Now we take $S(\xi)$ as the control. It is also convenient to represent the inverse problem by the cost function

$$I = \frac{1}{2} \int_C (p - p_d)^2 d\theta = I = \frac{1}{2} \int_C (p - p_d)^2 d\xi.$$

This eliminates terms in $\delta \left(\frac{ds}{d\xi} \right)$ from the gradient. The variations in the fluxes become

$$\begin{aligned} \delta(hF) &= C_1 \delta w \\ \delta[h(\eta + S)G - hS_\xi F] &= C_2 \delta w + h\delta SG - h\delta S_\xi F \end{aligned}$$

where C_1 and C_2 are the Jacobian matrices defined in equation (14). Choosing ψ to satisfy the adjoint equation (17) with the boundary condition

$$x_\xi \psi_2 - y_\xi \psi_3 = h[(\eta + S)s + S_\xi c]\psi_\eta = p - p_d$$

the variation in the cost reduces to

$$\begin{aligned} \delta I &= \int_C (p - p_d) \delta p d\xi \\ &\quad + \int_D \psi^T \left(\frac{\partial}{\partial \xi} \delta F + \frac{\partial}{\partial \eta} \delta G \right) d\xi d\eta \\ &= \int_C \psi^T (\delta S h \bar{G} - \delta S_\xi h \bar{F}) d\xi \\ &\quad + \int_D \psi^T \frac{\partial}{\partial \xi} (\delta S h G + \delta S_\xi h F) d\xi d\eta \end{aligned}$$

where F and G are the fluxes defined in equation (22), and \bar{F} and \bar{G} are F and G with the pressure terms deleted. Define

$$\begin{aligned} P &= \psi^T h \bar{F} + \int \psi^T \frac{\partial}{\partial \eta} (hF) d\eta \\ Q &= \psi^T h \bar{G} + \int \psi^T \frac{\partial}{\partial \eta} (hG) d\eta. \end{aligned} \quad (23)$$

Then

$$\begin{aligned} \delta I &= \int_C (Q \delta S - P \delta S_\xi) d\xi \\ &= \int_C \mathcal{G} \delta S d\xi, \end{aligned}$$

where the gradient is

$$\mathcal{G} = Q + \frac{\partial P}{\partial \xi}. \quad (24)$$

Here, the gradient is defined on the surface of the computational mesh. In effect, the gradient with respect to a number of design variables equal to the number of surface points can be calculated in one shot.

Design Using an Arbitrary Mesh

In order to treat a more general mesh we revert to equations (13-19). The difficulty in using these general equations is that the variation of the metric terms in the equations needs to be obtained in order to construct δI in equation (19). One way to accomplish this is by using an existing grid generation algorithm and applying finite differences to calculate the necessary information. While this approach would still obviate the use of multiple flow solutions to determine the gradient, it would still require the mesh generator to be used repeatedly. The number of mesh solutions required would be proportional to the number of design variables. This may be acceptable, since the flow solution process is typically much more computationally expensive than grid generation. Such a method should then ensure a significant savings over using finite differences for both the grid generation and flow solution processes, and it was successfully applied by the authors to two-dimensional problems in potential flow [15]. For three-dimensional design, where both the number of design variables and the computational cost of grid generation can be high, the method is excessively expensive. Further, for complicated three-dimensional configurations, for which fully automatic grid generation is still not practical, it will not be feasible.

This motivates the need to find a method which bypasses these difficulties. In order to remove the cost of the successive grid generation from the gradient calculation a successive grid perturbation method is therefore used. In this approach, which was also used by Burgreen and Baysal [4], an initial structured curvilinear body fitted grid over the initial configuration is created by any grid generation process before optimization. Then the geometry as well as the grid become inputs to the optimization process. New grids, which conform to the surface as it is modified, can then be generated by shifting

the grid points along each grid index line projecting from the surface by an amount which is attenuated as the arc length from the surface increases. If the outer boundary of the grid domain is held constant the modification to the grid has the form

$$\begin{aligned} x^{new} &= x^{old} + \mathcal{R} (x_s^{new} - x_s^{old}) \\ y^{new} &= y^{old} + \mathcal{R} (y_s^{new} - y_s^{old}), \end{aligned}$$

where x and y represent the volume grid points and, x_s and y_s represent the surface grid points. \mathcal{R} represents the arclength along the surface projecting mesh line measured on the original grid, from the outer domain, and normalized so that $\mathcal{R} = 1$ at the inner surface. The required variations in the metric terms can then be obtained in terms of surface perturbations since it follows that,

$$\begin{aligned} \delta x &= \mathcal{R} \delta x_s, \\ \delta y &= \mathcal{R} \delta y_s. \end{aligned}$$

and

$$[K] = \mathcal{R} [K_s] \quad (25)$$

where K_s is the transformation matrix K defined at the surface.

Now it is convenient to rewrite (19) after integrating by parts as.

$$\begin{aligned} \delta I = & \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi \\ & - \int_D \psi^T \frac{\partial}{\partial \xi} \left(\delta \left(\frac{\partial y}{\partial \eta} \right) f - \delta \left(\frac{\partial x}{\partial \eta} \right) g \right) \\ & + \psi^T \frac{\partial}{\partial \eta} \left(\delta \left(-\frac{\partial y}{\partial \xi} \right) f + \delta \left(\frac{\partial x}{\partial \xi} \right) g \right) d\xi d\eta \\ & - \int_C \psi^T \left\{ -\delta \left(\frac{\partial y}{\partial \xi} \right) \bar{f} + \delta \left(\frac{\partial x}{\partial \xi} \right) \bar{g} \right\} d\xi. \end{aligned} \quad (26)$$

where \bar{f} and \bar{g} are again the flux components f and g with the pressure terms deleted from the momentum equation. Substituting the expressions defined by equation (25) into (26) allows us to integrate the second term along the index direction projecting from the configuration, η surface without any dependence on particular design variables since the metric variations are fully determined by the surface perturbations. Thus, the expression for the variation in the cost function can be reduced to surface integrals only

$$\begin{aligned} \delta I = & \frac{1}{2} \int_C (p - p_d)^2 \delta \left(\frac{ds}{d\xi} \right) d\xi \\ & - \int_C \delta \left(\frac{\partial y}{\partial \eta} \right)_s \mathcal{N}_{11} - \delta \left(\frac{\partial x}{\partial \eta} \right)_s \mathcal{N}_{12} \\ & - \delta \left(\frac{\partial y}{\partial \xi} \right)_s \mathcal{N}_{21} + \delta \left(\frac{\partial x}{\partial \xi} \right)_s \mathcal{N}_{22} d\xi \\ & - \int_C \psi^T \left\{ -\delta \left(\frac{\partial y}{\partial \xi} \right) \bar{f} + \delta \left(\frac{\partial x}{\partial \xi} \right) \bar{g} \right\} d\xi, \end{aligned} \quad (27)$$

where

$$\mathcal{N}_{ij} = \begin{bmatrix} \int \psi^T \frac{\partial}{\partial \xi} (\mathcal{R} f) d\eta & \int \psi^T \frac{\partial}{\partial \xi} (\mathcal{R} g) d\eta \\ \int \psi^T \frac{\partial}{\partial \eta} (\mathcal{R} f) d\eta & \int \psi^T \frac{\partial}{\partial \eta} (\mathcal{R} g) d\eta \end{bmatrix}. \quad (28)$$

While this type of grid perturbation method does not guarantee that grid lines will not eventually cross if the perturbations are large, this point is irrelevant for gradient calculations since only analytic grid derivative information is needed. However, since we employ a numerical optimization algorithm with line searches along a descent direction, true re-gridding is also necessary. For these line search calculations the grid perturbation algorithm is used so long as negative volumes are not created. If singularities begin to develop in the grid, the original grid generator can be used to create a new grid and the process restarted. In practice, the grid perturbation method has proven to be robust, and no failures due to singularities have occurred during optimization of typical convex geometries.

Equation (27) has reduced the expression for δI into line integrals along the surface where the only remaining unknowns are the grid metrics. These surface grid metrics can be easily determined for any modification in the surface by direct evaluation. This suggests choosing a set of design variables which smoothly modifies the original shape, say b_i . The gradient can then be defined with respect to these design variables as

$$\mathcal{G}(b_i) = \frac{\delta I}{\delta b_i}, \quad (29)$$

where δI is calculated by (27) with each term b_i being independently perturbed by a finite step.

Therefore, to construct \mathcal{G} , an independent basis space of perturbation functions b_i , $i = 1, 2, \dots, n$ (n = number of design variables) is chosen that allows for the needed freedom of the design space. Design variables are chosen with the following form, suggested by Hicks and Henne [6, 7]:

$$\begin{aligned} b(x) &= \sin \left(\pi x^{\frac{\log_{10}(0.5)}{\log_{10}(t_1)}} \right)^{t_2} \\ b(x) &= x^{t_1} (1 - x) e^{-t_2 x}, \end{aligned}$$

where t_1 and t_2 control the center and width of the perturbation and x is the normalized chord length. When distributed over the entire chord on both upper and lower surfaces these analytic perturbation functions admit a large possible design space. They have the advantage of being space based functions, as opposed to frequency based functions, and thus they allow for local control of the design. They can be chosen such that symmetry, thickness, or volume can be explicitly constrained. Further, particular

choices of these variables will concentrate the design effort in regions where refinement is needed, while leaving the rest of the airfoil section virtually undisturbed. The disadvantage of these functions is that they do not form a complete basis space, nor are they orthogonal. Thus, they do not guarantee that a solution, for example, of the inverse problem for a realizable target pressure distribution will necessarily be attained. Here, they are employed due to their ease of use and ability to produce a wide variation of shapes with a limited number of design variables.

Implementation of the Euler based design methods

Both of the design methods have been successfully implemented. The two techniques share many common features such as the flow and adjoint solution algorithms. The procedures can be summarized as follows.

1. Solve the flow equations (8–12) for ρ , u , v , p , E , H , U , and V .
2. Smooth the cost function if necessary by (20).
3. Solve the adjoint equations (17) for ψ subject to the boundary condition (18).
4. Either calculate P and Q , by (23), from the variation in the control $S(\xi)$, or evaluate \mathcal{N}_{ij} from equation (28).
5. Evaluate \mathcal{G} by equation (24) or (29).
6. Project \mathcal{G} into a feasible direction subject to any geometric constraints to obtain $\tilde{\mathcal{G}}$.
7. Then either correct the mapping $S(\xi)$ or update the design variables b_i based on the direction from steepest decent

$$\delta S(\xi) = -\lambda \tilde{\mathcal{G}} \quad \text{or} \quad \delta b_i = -\lambda \tilde{\mathcal{G}}$$

or as an alternative a quasi-Newton method.

8. Return to 1.

In practice these methods resemble those used by Hicks, Reuther et al. [16, 8, 17] with control theory replacing the brute force, finite difference based, gradient calculation. In their most recent applications, single point design is accomplished on realistic wing body configurations with full nonlinear modeling of supersonic flow. Unlike these procedures, the current methods' computational cost do not hinge upon the number of design variables, which in these cases is either the number of surface mesh points used to represent $S(\xi)$, or the number of perturbation functions b_i . Thus it can be hoped that with the implementation of this method in three-dimensions,

non-linear aerodynamic design of complete aircraft can be brought into the realm of computational feasibility. The method also has the advantage of being quite general in that arbitrary choices for both the design variables and optimization technique are admitted, as is demonstrated by the two implementations.

The practical implementation of the design methods rely heavily upon fast and accurate solvers for both the state (w) and co-state (ψ) fields. Further, to improve the speed and realizability of the methods, a robust choice of the optimization algorithm must be made. In this work, Jameson's FLO82 computer program has been used to solve the Euler equations. This program uses a multistage time stepping scheme with multigrid acceleration to obtain very rapid steady state solutions, typically in 25 steps. The adjoint equations are solved by a similar method, in which the flux calculations for the Euler equations are replaced by the corresponding formulas for the adjoint equation.

In the implementation with analytic mapping a simplified gradient procedure is used as the optimization process. To preserve the smoothness of the profile the gradient is smoothed at each step. Thus the change in the shape function $S(\xi)$ is defined by solving

$$\delta S - \frac{\partial}{\partial \xi} \beta \frac{\partial}{\partial \xi} \delta S = -\Lambda \mathcal{G},$$

where β is a smoothing parameter. Then, to first order, the variation in the cost is

$$\begin{aligned} \delta I &= \int \mathcal{G} \delta S d\xi \\ &= - \int \frac{1}{\Lambda} \left[\delta S - \delta S \frac{\partial}{\partial \xi} \beta \frac{\partial}{\partial \xi} \delta S \right] d\xi \\ &= - \int \frac{1}{\Lambda} \left[\delta S^2 + \beta \left(\frac{\partial}{\partial \xi} \delta S \right)^2 \right] d\xi \\ &< 0. \end{aligned}$$

For the implementation on arbitrary meshes a quasi-Newton optimization method is employed. For this purpose the QNMDIF program developed by Gill, Murray and Wright [5] is used.

The option to minimize the pressure drag coefficient is realized in both methods by redefining the cost function as

$$I = C_d = \frac{1}{\frac{1}{2} \rho_\infty q_\infty^2 \bar{c}} \int_C p \frac{\partial y}{\partial \xi} d\xi,$$

where \bar{c} is the chord length. To prevent the procedure from trying to reduce drag by reducing the profile to a non-lifting flat plate a target pressure distribution can be retained in the cost function, which becomes

$$I = \frac{1}{2} \Omega_1 \int_C (p - p_d)^2 d\xi + \Omega_2 C_d,$$

where Ω_1 and Ω_2 are weighting parameters.

Numerical tests of the design method Using an Analytic Mapping

Three preliminary test cases are presented for the design algorithm using an analytic mapping. The first two address the problem of attaining a desired pressure distribution for lifting airfoils. The most convenient method of obtaining such solutions with the present design method is to determine the lift coefficient associated with the target pressure distribution, and match this lift with the initial airfoil. Therefore the angle of attack is adjusted after every 5 iterations of the flow calculation to drive the solution toward the target lift coefficient. The first example using this technique, shown in Figure 1, drives the Korn airfoil toward the target pressure distribution for the NACA 64A410 airfoil at $M_\infty = 0.75$, $\alpha = 0^\circ$, and $C_l = 0.66$. This case requires a shift in the shock location and a significant change in the profile shape. The final solution almost exactly recovers the pressure distribution and the airfoil shape. In the next example, Figure 2, the process is reversed to modify a NACA 64A410 airfoil to recover the Korn pressure distribution at Mach 0.75 as a target. Again the target is almost attained, except at the trailing edge, where a perfect cusp is not easy to reproduce. The last test case introduces drag as the cost function. Again the design process is carried out in the fixed lift mode. In Figure 3, a NACA 64A410 is again used as a starting airfoil. The design is at $M_\infty = 0.75$ and $C_l = 0.68$ where a strong shock causes considerable wave drag in the initial airfoil. The objective is to reduce the drag with the smallest possible change in the airfoil. Therefore the pressure distribution of the initial airfoil is retained as the target pressure distribution, and the cost function is a blend of the drag coefficient and the deviation from the target pressure. The final design has a reduction in drag from $C_d = 0.0127$ to $C_d = 0.0016$.

Numerical tests of the design method Using a General Mesh

Several test cases are conducted with the design method using the general mesh implementation. This technique more closely resembles brute force optimization methods. First, perturbations are made to individual design variables, and second, the calculated gradient vector is fed into a quasi-Newton optimization method. This similarity allows for easy comparisons of efficiency and accuracy between the brute force method and the control theory method. A fair test of the efficiency is conducted by calculating the gradient using both methods while adjusting

the level of convergence of the flowfield and adjoint solvers to obtain the optimum efficiency for each method. For the brute force calculation, the flow solver is restarted from a previously converged solution and thus requires only a few additional multi-grid iterations for each component of the gradient. Nevertheless, a dramatic benefit in efficiency is seen by using control theory to calculate the gradient. Using 50 design variables, 96.5 seconds of Cray C90 time is required to calculate the brute force gradient, compared to only 14.1 seconds required for the corresponding calculation using control theory. With the present implementation, the control theory produces a less accurate estimate of the gradient than the brute force method. Despite this difficulty the estimate of the gradient is still more than adequate to drive the numerical optimization procedure toward convergence.

The first test case for the design method with an arbitrary mesh is shown in Figure 4. The NACA 0012 airfoil is modified to achieve the pressure distribution of a NACA 64A410 airfoil at Mach 0.735, $\alpha = 0^\circ$ and $C_l = 0.6623$. 50 Hicks-Henne design variables distributed around the airfoil are used to modify the shape with the final airfoil and pressure distribution matching the target very closely. The second test case also uses 50 design variables and the NACA 0012 as a starting condition, as illustrated in Figure 5. A target pressure distribution of the RAE 2822 airfoil at $\alpha = 1^\circ$, Mach 0.75 and $C_l = 0.6982$ is specified. Again the design method converges to the target almost exactly matching even the shock position and strength. A third test case of the method in inverse mode is displayed in Figure 6. This time, the NACA 0012 airfoil is driven towards the target pressure distribution of the GAW 72 airfoil operating at Mach 0.70, $\alpha = 1^\circ$ and $C_l = 0.8158$. Under these conditions the target airfoil displays a very strong shock, yet the design method is able to converge to the desired shape without visible discrepancies using 50 design variables. The fourth test case represents a greater challenge. The Korn airfoil at Mach 0.75, $\alpha = 0^\circ$ and $C_l = 0.6249$ is chosen as the target pressure. The challenge is presented by the fact that the Korn airfoil at this condition has shock free supercritical flow with no wave drag. Figure 7 shows the NACA 0012 airfoil being driven towards the desired target, with the design method employing 52 design variables. As the design is approached there is a tendency to produce a double shock pattern instead of a smooth recompression. In effect, the design space for this problem is more nonlinear than in the previous test cases. The final design is very close, but does show some discrepancies from the desired pressure distribution. In the fifth test case the RAE 2822 airfoil is revisited for a target pressure distribution. However, a potential flow solver was used to provide the target pressure distribution. Thus this pressure

distribution is not realizable by the Euler equations because the shock wave of the target is modeled as an isentropic jump. The final solution with 50 design variables, shown in Figure 8, very closely approximates the desired pressure distribution, but of course does not match it exactly. An examination of the final airfoil reveals a striking difference between it and the expected airfoil. It can be seen that for such strong shock cases, the potential flow equation can give quite incorrect results. The sixth test case introduces drag as the objective function to be minimized. The NACA 64A410 airfoil at Mach 0.75, $\alpha = 1^\circ$ $C_l = 0.700$ and $C_d = 0.0162$ is used as a starting condition. Figure 9 illustrates by choosing 18 design variables that modify the camber, the optimization procedure is able to reduce the drag to just 7 counts in 3 design cycles. This is accomplished while the lift coefficient, Mach number and thickness distribution remain unchanged. In the final test case drag is again used as the objective function except that it is augmented by a lower limit on minimum allowable C_p . This constraint becomes an additional term in the boundary condition of the adjoint equation and demonstrates the versatility of the method. With this constraint added in the way of a penalty function, Figure 10 shows that the NACA 0012 airfoil operating at Mach 0.75 and $C_l = 0.6$ can be substantially improved. Twenty design variables are chosen which preserve symmetry in the airfoil. With a limit of -1.2 set for C_p , the design procedure is able to reduce the drag from 266 counts to 54 counts in 4 design cycles. The final pressure distribution is observed to display a flat top at $C_p = -1.2$ as would be expected.

Conclusions and Recommendations

We have developed two control theory based airfoil design methods for the Euler equations. The difference in the two methods lie in the manner in which the variations with respect to the position of the mesh points are treated. The first relies on an analytic mapping, while the second permits an arbitrary mesh. The methods represent an extension of our previous work on the design of airfoils with the potential flow equation. The new methods are both efficient and robust, combining the versatility of numerical optimization methods with the efficiency of inverse design methods. A demonstrated seven fold improvement in efficiency was realized over the brute force optimization method. In three-dimensional problems for which the number of design variables must be much larger, the improvement in efficiency will be even more significant. The direct extendibility of this method to three-dimensions is perhaps its most appealing aspect, with the final goal being to create practical aerodynamic shape design methods for complete aircraft configurations.

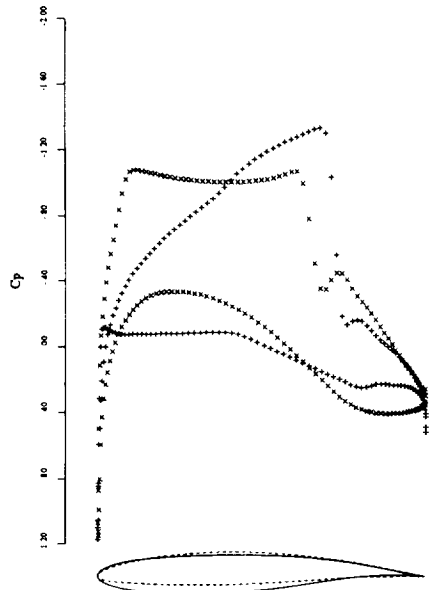
Acknowledgment

This work has benefited from the generous support of AFOSR under Grant No. AFOSR-91-0391 and also the support of the Aerodynamics Division of NASA Ames Research Center.

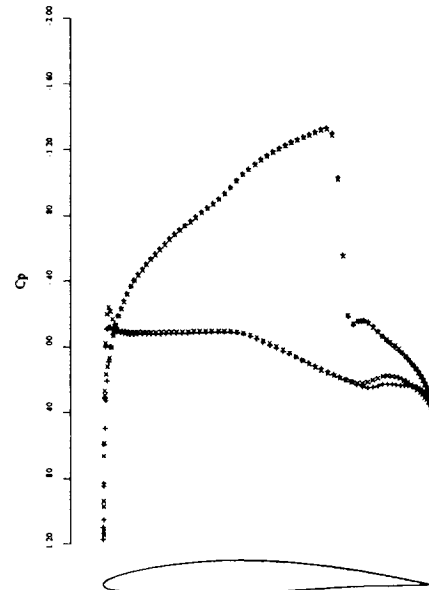
References

- [1] O. Baysal and M. E. Eleshaky. Aerodynamic sensitivity analysis methods for the compressible Euler equations. *Journal of Fluids Engineering*, 113(4):681-688, 1991.
- [2] O. Baysal and M. E. Eleshaky. Aerodynamic design optimization using sensitivity analysis and computational fluid dynamics. *AIAA Journal*, 30(3):718-725, 1992.
- [3] O. Baysal and M. E. Eleshaky. Airfoil shape optimization using sensitivity analysis on viscous flow equations. *Journal of Fluids Engineering*, 115:75-84, 1993.
- [4] G. W. Burgreen and O. Baysal. Three-dimensional aerodynamic shape optimization of wings using sensitivity analysis. *AIAA paper 94-0094*, 32th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1994.
- [5] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, 1981.
- [6] R. M. Hicks and P. A. Henne. Wing design by numerical optimization. *Journal of Aircraft*, 15:407-412, 1978.
- [7] R. M. Hicks and P. A. Henne. Wing design by numerical optimization. *AIAA paper 79-0080*, 1979.
- [8] 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.
- [9] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3:233-260, 1988.
- [10] 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.
- [11] V. M. Korivi, A. C. Taylor III, P. A. Newman, G. W. Hou, and H. E. Jones. An incremental strategy for calculating consistent discrete CFD sensitivity derivatives. *NASA TM 104207*, Langley Research Center, Hampton, VA, February 1992.

- [12] J. L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer-Verlag, New York, 1971. Translated by S.K. Mitter.
- [13] P. A. Newman, G. W. Hou, H. E. Jones, A. C. Taylor III, and V. M. Korivi. Observations on computational methodologies for use in large-scale gradient-based, multidisciplinary design incorporating advanced CFD codes. *NASA TM 104206*, Langley Research Center, Hampton, VA, February 1992.
- [14] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, New York, 1984.
- [15] J. Reuther and A. Jameson. Control theory based airfoil design for potential flow and a finite volume discretization. *AIAA paper 91-499*, 32th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1994.
- [16] 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.
- [17] J. J. Reuther, S. E. Cliff, R. M. Hicks, and C. P. van Dam. Practical design optimization of wing/body configurations using the Euler equations. *AIAA paper 92-2633*, 1992.
- [18] G. R. Shubin. Obtaining cheap optimization gradients from computational aerodynamics codes. *Internal paper AMS-TR-164*, Boeing Computer Services, June 1991.
- [19] G. R. Shubin and P. D. Frank. A comparison of the implicit gradient approach and the variational approach to aerodynamic design optimization. *internal paper AMS-TR-164*, Boeing Computer Services, April 1991.
- [20] S. Ta'asan, G. Kuruvila, and M. D. Salas. Aerodynamic design and optimization in one shot. *AIAA paper 91-005*, 30th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1992.



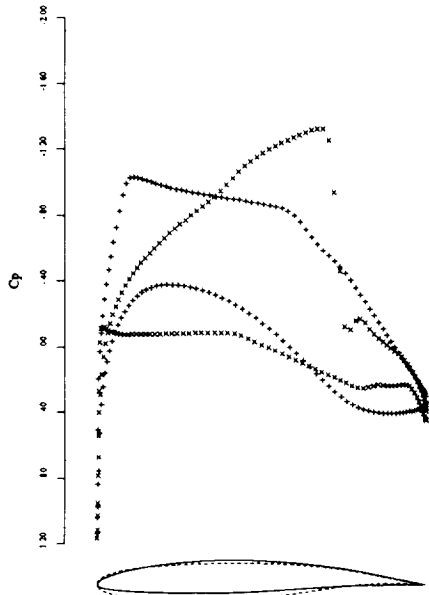
1a: Initial Condition
 $C_l = 0.7019$, $C_d = 0.0015$, $\alpha = 0.266^\circ$



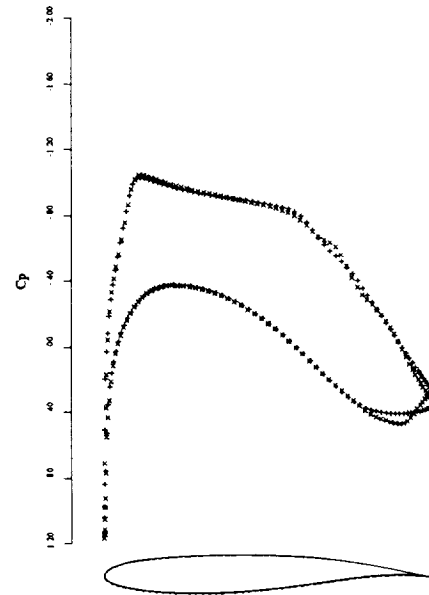
1b: 50 Design Iterations
 $C_l = 0.6610$, $C_d = 0.0136$, $\alpha = -0.039^\circ$

Figure 1: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.

—, × Initial Airfoil: Korn.
 - - -, + Target C_p : NACA 64A410, $M = 0.75$.
 Inverse Design



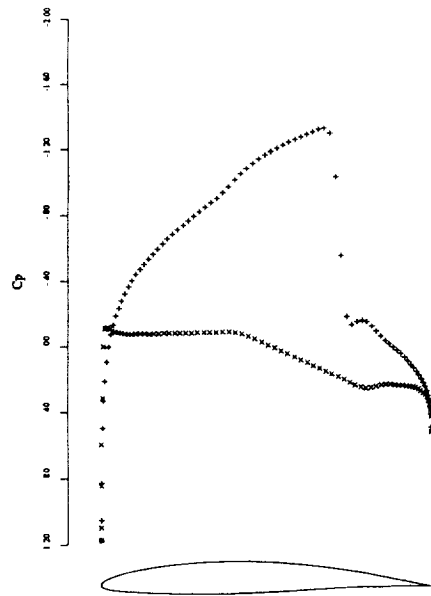
2a: Initial Condition
 $C_l = 0.6854$, $C_d = 0.0144$, $\alpha = -0.073^\circ$



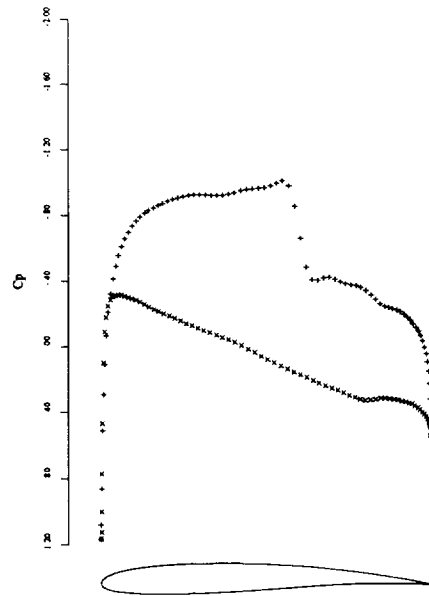
2b: 100 Design Iterations
 $C_l = 0.6284$, $C_d = 0.0003$, $\alpha = -0.073^\circ$

Figure 2: Lifting Design Case, $M = 0.75$ Fixed Lift Mode

—, × Initial Airfoil: NACA 64A410.
 - - -, + Target C_p : Korn, $M = 0.75$.
 Inverse Design

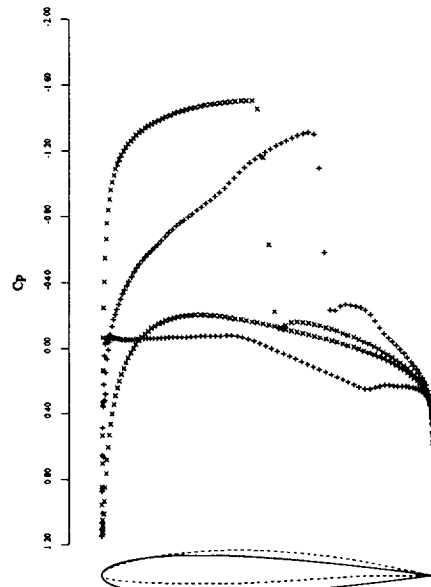


3a: Initial Condition
 $C_l = 0.6774$, $C_d = 0.0144$, $\alpha = -0.093^\circ$

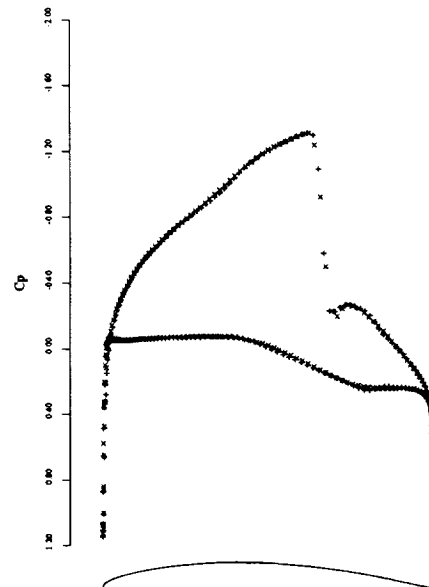


3b: 25 Design Iterations
 $C_l = 0.6855$, $C_d = 0.0010$, $\alpha = -0.722^\circ$

Figure 3: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.
 —, Initial Airfoil: NACA 64A410.
 Drag Reduction

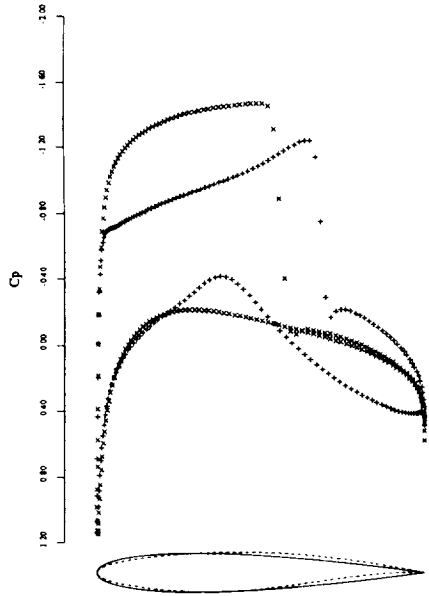


4a: Initial Condition
 $C_l = 0.6623$, $C_d = 0.0281$, $\alpha = 2.853^\circ$

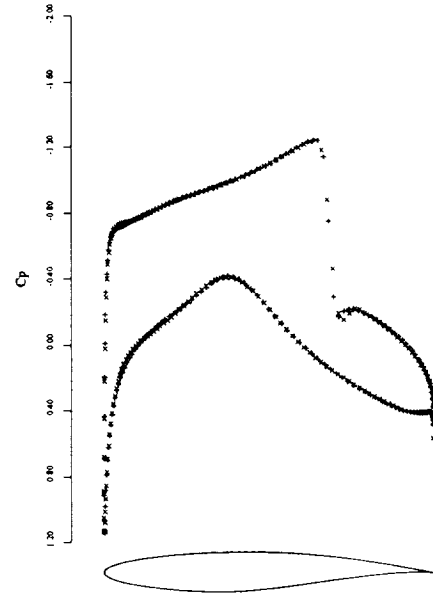


4b: 22 Design Iterations
 $C_l = 0.6623$, $C_d = 0.0079$, $\alpha = 0.059^\circ$

Figure 4: Lifting Design Case, $M = 0.735$, Fixed Lift Mode.
 —, \times Initial Airfoil: NACA 0012.
 - - -, + Target C_p : NACA 64A410, $M = 0.735$.
 Inverse Design



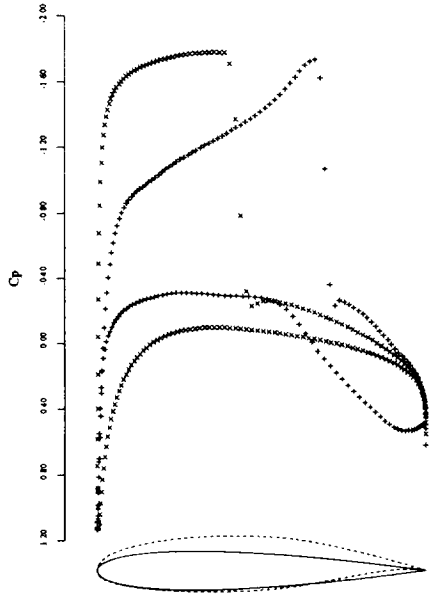
5a: Initial Condition
 $C_l = 0.6982$, $C_d = 0.0379$, $\alpha = 2.593^\circ$



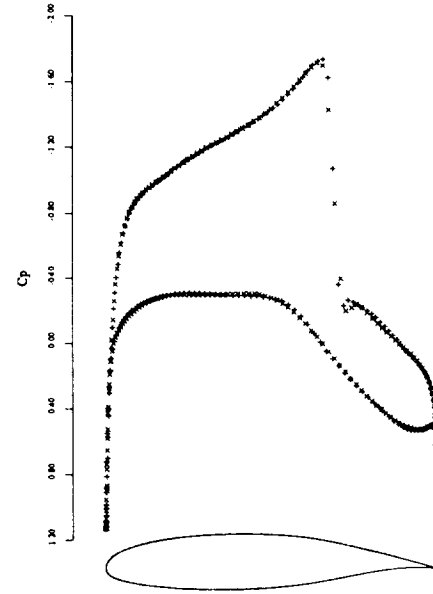
5b: 24 Design Iterations
 $C_l = 0.6982$, $C_d = 0.0074$, $\alpha = 0.962^\circ$

Figure 5: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.

—, × Initial Airfoil: NACA 0012.
 ---, + Target C_p : RAE 2822, $M = 0.75$.
 Inverse Design



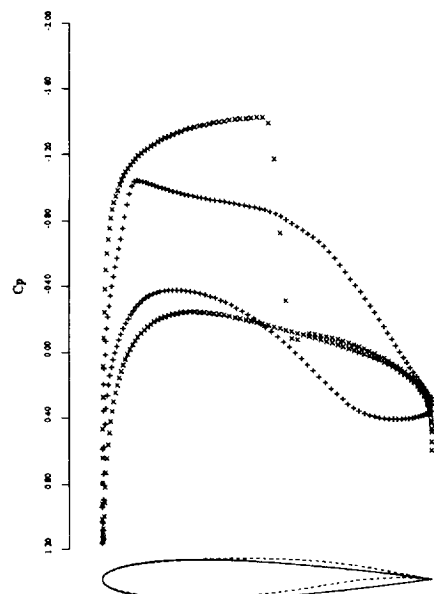
6a: Initial Condition
 $C_l = 0.8158$, $C_d = 0.0348$, $\alpha = 3.835^\circ$



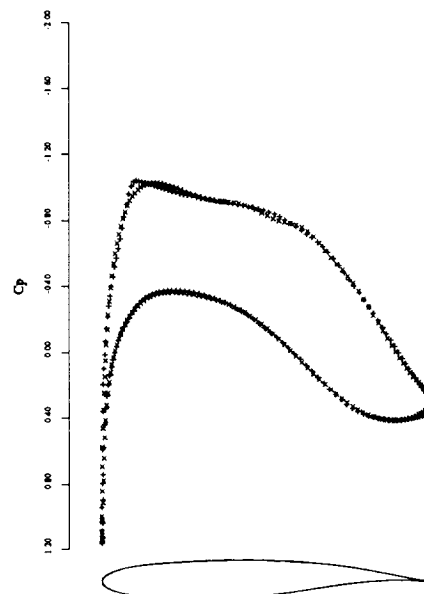
6b: 32 Design Iterations
 $C_l = 0.8158$, $C_d = 0.0159$, $\alpha = 0.094^\circ$

Figure 6: Lifting Design Case, $M = 0.70$, Fixed Lift Mode.

—, × Initial Airfoil: NACA 0012.
 ---, + Target C_p : GAW 72, $M = 0.70$.
 Inverse Design



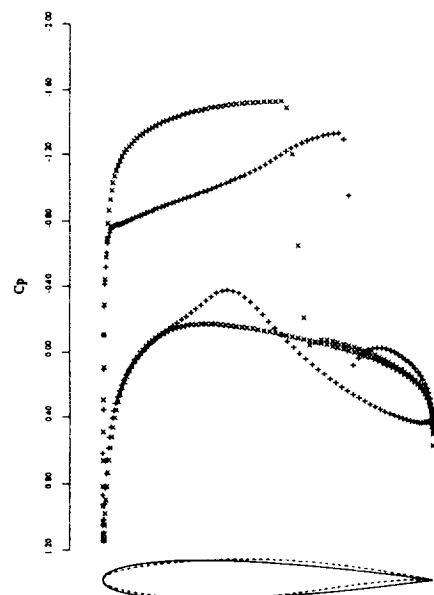
7a: Initial Condition
 $C_l = 0.6249$, $C_d = 0.0294$, $\alpha = 1.958^\circ$



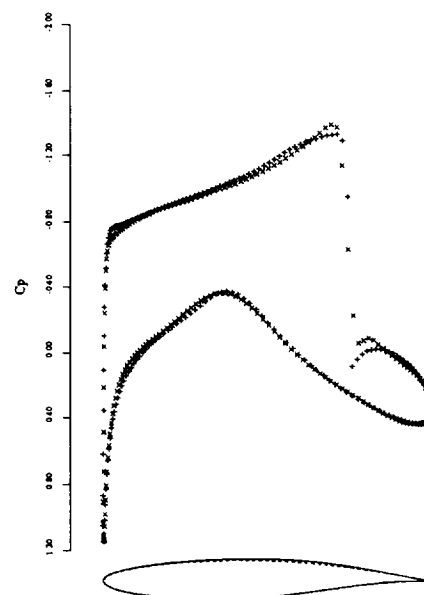
7b: 21 Design Iterations
 $C_l = 0.6249$, $C_d = 0.0004$, $\alpha = 0.248^\circ$

Figure 7: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.

—, x Initial Airfoil: NACA 0012.
 - - -, + Target C_p : Korn, $M = 0.75$.
 Inverse Design



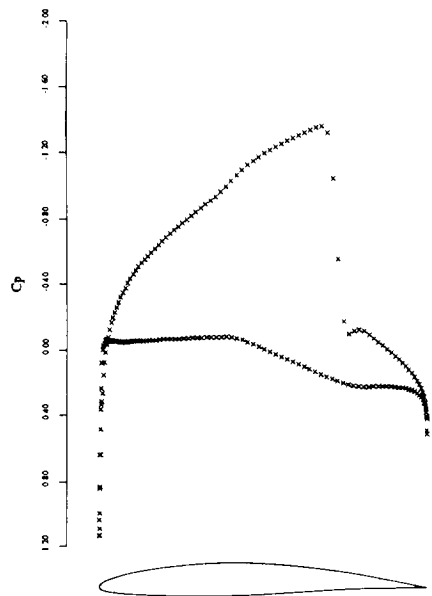
8a: Initial Condition
 $C_l = 0.7970$, $C_d = 0.0516$, $\alpha = 3.065^\circ$



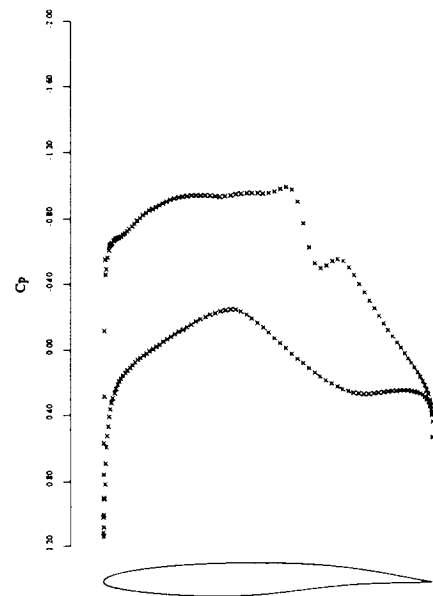
8b: 21 Design Iterations
 $C_l = 0.7970$, $C_d = 0.0147$, $\alpha = 1.126^\circ$

Figure 8: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.

—, x Initial Airfoil: NACA 0012.
 - - -, + Target C_p : RAE 2822 (Potential Flow), $M = 0.75$.
 Inverse Design

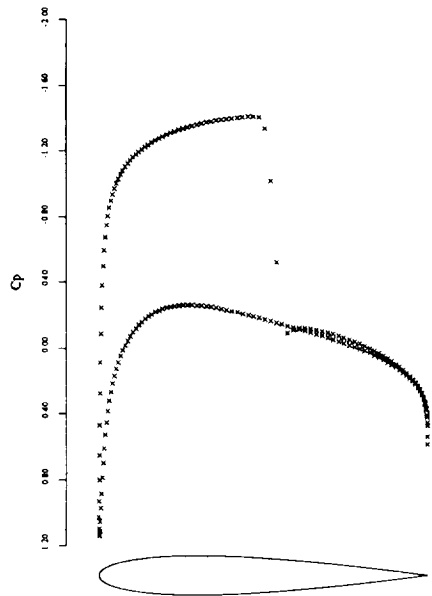


9a: Initial Condition
 $C_l = 0.7000$, $C_d = 0.0162$, $\alpha = 0.361^\circ$

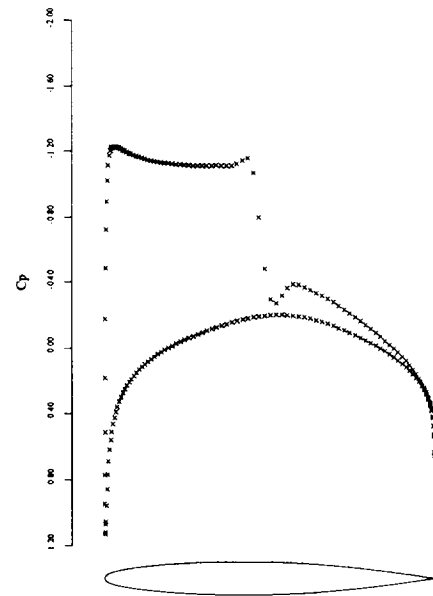


9b: 3 Design Iterations
 $C_l = 0.7000$, $C_d = 0.0007$, $\alpha = 0.964^\circ$

Figure 9: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.
 —, Initial Airfoil: NACA 64A410.
 Drag Reduction



10a: Initial Condition
 $C_l = 0.6000$, $C_d = 0.0266$, $\alpha = 2.629^\circ$



10b: 4 Design Iterations
 $C_l = 0.6000$, $C_d = 0.0054$, $\alpha = 2.713^\circ$

Figure 10: Lifting Design Case, $M = 0.75$, Fixed Lift Mode.
 —, Initial Airfoil: NACA 0012.
 Drag Reduction

