

**AGARD FDP/VKI Special Course**

**on**

# **Optimum Design Methods for Aerodynamics**

**von Karman Institute for Fluid Dynamics  
Rhode-Saint-Genese, Belgium  
25-29 April 1994**

**AGARD Report 803**



# Airfoil Optimization by the *One-Shot* Method

G. Kuruvila\*

ViGYAN, Inc.  
Hampton, VA 23666-1325  
USA

Shlomo Ta'asan†

Institute for Computer Applications in Science and Engineering  
Hampton, VA 23681-0001  
USA

M. D. Salas‡

NASA Langley Research Center  
Hampton, VA 23681-0001  
USA

## SUMMARY

An efficient numerical approach for the design of optimal aerodynamic shapes is presented in this paper. The objective of any optimization problem is to find the optimum of a cost function subject to a certain state equation (Governing equation of the flow field) and certain side constraints. As in classical optimal control methods, the present approach introduces a costate variable (Lagrange multiplier) to evaluate the gradient of the cost function. High efficiency in reaching the optimum solution is achieved by using a multigrid technique and updating the shape in a hierarchical manner such that smooth (low-frequency) changes are done separately from high-frequency changes. Thus, the design variables are changed on a grid where their changes produce nonsmooth (high-frequency) perturbations that can be damped efficiently by the multigrid. The cost of solving the optimization problem is approximately two to three times the cost of the equivalent analysis problem.

## LIST OF SYMBOLS

$C_p$	pressure coefficient
$F$	cost function
$f_k$	$k$ th shape function
$i$	unit vector in $x$ -direction
$j$	unit vector in $y$ -direction
$M_\infty$	free stream Mach number
$n$	unit normal
$t$	unit tangent
$U_\infty$	free stream velocity
$y^{U,L}$	$y$ -coordinate of the upper and lower surface of the airfoil
$\alpha$	amplitude of shape functions (design variables)
$\tilde{\alpha}$	direction of change of $\alpha$
$\alpha_k^{U,L}$	components of $\alpha$ (upper and lower surface amplitudes of the $k$ th shape function)
$\Gamma$	circulation
$\gamma$	ratio of specific heats
$\varepsilon$	magnitude of change of $\alpha$
$\zeta$	angle of attack

$\Theta$	$\theta$ corrected for Mach number
$\theta$	angular position of a far-field location
$\lambda$	Lagrange multiplier
$\mu_k$	$k$ th component of the gradient of $F$
$\rho$	density
$\phi$	full velocity potential
$\phi_0$	target potential
$\Upsilon$	coefficient of the delta function

## 1. INTRODUCTION

Analysis of flow fields using computational fluid dynamics (CFD) has come a long way. Today, accurate computation of the flow field around realistic aircraft configurations using the Navier Stokes equations with turbulence modeling can be done at affordable cost and reasonable turnaround time. Design and optimization of aircraft configurations, on the other hand, is far from this level of maturity. In the last two decades, many different techniques have been developed to design aerodynamically better aircraft. These techniques can be classified into three broad categories, namely inverse design methods, loosely coupled optimization (LCO), and tightly coupled optimization (TCO).

The inverse design method,<sup>1-6</sup> pioneered by Lighthill, requires a priori knowledge of a desirable pressure or velocity distribution and some strategy for obtaining a shape that produces this distribution. The quality of the shape obtained from the inverse design method is strictly a function of the distribution it is required to match. Therefore, a weakness of this approach is its dependence on the experience and knowledge of the designer to establish desirable velocity or pressure distributions. In addition, the method does not lend itself to the imposition of constraints.

In the LCO approach, an analysis code interacts with a numerical optimization code to find a shape that meets some design objective (i.e., minimizes some cost function). To achieve this goal, the analysis problem is solved many times to find the best combination of perturbations to the design variables that both minimizes the cost function and satisfies the constraints. This process is repeated until the cost function cannot be further reduced. Examples of this approach are found in Refs. 7-10. The approach can be viewed as a two part process: an inner loop that finds both a direction and a step size to update the design variables and an outer loop that repeats the inner loop until the cost function reaches a minimum. If each inner-loop pass requires  $N$  solutions of

\* Research Engineer

† Senior Scientist

‡ Chief Scientist, Fluid Mechanics and Acoustics Division

the analysis problem (where  $N$  is proportional to the number of design variables) and the outer loop requires  $C$  iterations (where  $C$  depends on how far the initial conditions are from a minimum), then the cost of this approach is approximately  $N \times C$  times the cost of the analysis problem. The LCO method can be improved by analytically evaluating the sensitivity derivatives needed to update the design variables.<sup>11</sup> Usually this requires the inversion of a very large matrix. For three-dimensional problems, the size of this matrix can render the method impractical with current computer technology.

Even greater efficiency can be achieved through a TCO method. With this method, the optimization and analysis problems are attacked simultaneously. The TCO problem requires the solution of an adjoint problem equivalent in complexity to the analysis problem. This results in an overall cost that is proportional to  $2C$ . The factor of 2 results from doubling the number of equations that govern the problem. This approach has been discussed in Ref. 12. Even this procedure can become prohibitively expensive for practical aerodynamic design and optimization problems.

The *One-Shot* method<sup>13,14</sup> overcomes the unacceptable cost of the existing design and optimization procedures. It brings the cost of design and optimization to the same order as that of a single analysis. High performance is achieved by exploiting the property of the partial differential equations (associated with the scales (frequency) of the errors) which govern the physics of the flow and by the efficient damping out of high-frequency error components with multigrid. Consider the subsonic flow over an airfoil profile. The change in the shape of the profile of a given wavelength produces changes of the same wavelength in the solution. These changes penetrate into the flow field only up to a distance that is proportional to the wavelength of the perturbation. Thus, while the high-frequency changes in the shape of the airfoil produce changes in the solution that are of high frequency and remain local to the neighborhood of the airfoil, the smooth (low-frequency) changes in the shape produce smooth changes to the solution and are global in nature. Typically, any relaxation scheme quickly damps the high-frequency components of the error on a grid. Multigrid efficiently damps the whole spectrum of error components by relaxing the governing equations on a sequence of grids of varying resolution.

Therefore, the basic idea of the *One-Shot* method is to change the shape of the airfoil profile in a hierarchical manner such that smooth changes are made separately from high-frequency changes. Because each of these changes involves a different scale, the governing equation of the flow field can be solved efficiently on grids of appropriate resolution. Thus, the flow field due to smooth changes in the shape of the airfoil is solved on coarse grids, and the flow field due to increasingly high-frequency shape changes is solved on increasingly fine grids. This breaks the optimization procedure into a sequence of suboptimization problems, each of a given scale; therefore, the problem is well conditioned. The resulting optimization procedure is very efficient because the work on a particular scale is done on the appropriate grid. (Ill conditioning results from working on many scales simultaneously.) The *One-Shot* method is implemented within a full approximation scheme (FAS) full multigrid (FMG) algorithm. The solution process starts on the coarsest grid, where only the smooth component of the shape function is updated. This solution is interpolated to the next finest grid, where it serves as an initial approximation of the solution on that grid. This process is continued until the finest grid is reached. Thus, smooth (low-frequency) shapes are updated on coarse grids; high-frequency shapes

are updated on finer grids. The fine- to coarse-grid transfers are designed such that the optimization problem at each grid level is driven by the fine-grid residual. The resulting algorithm has an estimated overall cost that ranges from two to three times the cost of the analysis problem.

The successful application of the *One-Shot* method to the aerodynamic shape design problem was first reported in Ref. 14. The capability of the method was demonstrated by using the small-disturbance potential equation as the governing equation of the flow field. However, in that study, the issue of updating the grid was avoided. In the present study, the full potential equation is used as the governing equation; hence, the grid must be updated as the shape changes. In this work, the adjoint equation and the corresponding gradient of the cost function are derived. The solution procedure and some typical results are also presented.

## 2. CONSTRAINED MINIMIZATION PROBLEM

A general constrained minimization problem can be stated as

$$\min_{b, Q} F[b, Q(b)] \quad (2.1)$$

subject to

$$R[b, Q(b)] = 0 \quad (2.2)$$

and

$$C_n[b, Q(b)] \leq 0 \quad (n = 1, 2, \dots, N) \quad (2.3)$$

where  $F$  is the cost function;  $b$  the design variables; and  $Q$ , the state variables. The set of state equations is denoted by  $R$  and the side constraints are denoted by  $C_n$ ;  $C_n$  is referred to as a side constraint because the state equation is considered to be the primary constraint of the problem.

In aerodynamic minimization problems, the cost function is, for example, the drag coefficient  $C_d$  or the ratio of drag to lift  $C_d/C_l$ . The design variables are, typically, the shape parameters that define the shape of the body in the flow field. The state equations are the governing equations of the flow field and their boundary conditions. Depending on the level of fidelity of the mathematical model, the governing equations are the Navier-Stokes equations, the Euler equations, or the potential equations. The side constraints are either geometric constraints like the maximum thickness of the airfoil section, the volume of the wing, or aerodynamic constraints like maximum lift ( $\max C_l$ ).

### 2.1 The Necessary Conditions

The objective of the minimization problem is to find  $b^*$  and the corresponding  $Q^*$  such that  $F(b^*, Q^*)$  is a minimum and the state equations and the side constraints are satisfied. A necessary condition for  $b^*$  to be at a minimum is

$$\nabla_b F(b^*, Q^*) = 0 \quad (2.4)$$

where

$$\nabla_b F \equiv \frac{\partial F}{\partial b} + \left( \frac{dQ}{db} \right)^T \frac{\partial F}{\partial Q} \quad (2.5)$$

( $\nabla_b F$  will be referred to as the gradient of  $F$ ). This necessary condition can be proved by contradiction as follows. The

Taylor-series expansion of  $F$  in the neighborhood of  $b^*$  and  $Q^*$  can be written as

$$F(b^* + \varepsilon \tilde{b}, Q^* + \varepsilon \tilde{Q}) = F(b^*, Q^*) + \varepsilon \tilde{b}^T \nabla_b F(b^*, Q^*) + O(\varepsilon^2) \quad (2.6)$$

where  $\varepsilon$  is a positive scalar and  $\tilde{b}$  is a vector;  $\varepsilon \tilde{b}$  is the change in  $b^*$ , and  $\varepsilon \tilde{Q}$  is the corresponding change in  $Q^*$  that satisfies the state equations and the side constraints. If  $\nabla_b F(b^*, Q^*)$  is nonzero, then a vector  $\tilde{b}$  must exist for which

$$\tilde{b}^T \nabla_b F(b^*, Q^*) < 0 \quad (2.7)$$

(e.g.,  $\tilde{b} = -\nabla_b F(b^*, Q^*)$ ). A vector  $\tilde{b}$  that satisfies (2.7) is called a descent direction at  $b^*$ . Given any descent direction  $\tilde{b}$ , a positive scalar  $\bar{\varepsilon}$  exists such that for all positive  $\varepsilon$  that satisfy  $\varepsilon \leq \bar{\varepsilon}$ ,

$$\varepsilon \tilde{b}^T \nabla_b F(b^*, Q^*) + O(\varepsilon^2) < 0 \quad (2.8)$$

If we substitute (2.8) into (2.6), then

$$F(b^* + \varepsilon \tilde{b}, Q^* + \varepsilon \tilde{Q}) \leq F(b^*, Q^*) \quad (2.9)$$

for all such  $\varepsilon$ . Hence, unless  $\nabla_b F(b^*, Q^*) = 0$ , the neighborhood of  $b^*$  contains points with a lower function value than  $F(b^*, Q^*)$ . The other necessary conditions that must be satisfied at the minimum are the state equations and the side constraints.

## 2.2 The Minimization Process

At some initial  $b$ , any minimization process seeks to find a descent direction  $\tilde{b}$  and a step size  $\varepsilon$  in which to change  $b$  such that

$$F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q}) \leq F(b, Q) \quad (2.10)$$

where  $\varepsilon \tilde{Q}$  is the corresponding change in  $Q$  that satisfies the state equations and the side constraints. This process is repeated several times until a minimum is reached.

### The Descent Direction

A descent direction  $\tilde{b}$  can be determined as follows. The Taylor series expansion of  $F$  about  $b$  and  $Q$  can be written as

$$F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q}) = F(b, Q) + \varepsilon \tilde{b}^T \nabla_b F(b, Q) + O(\varepsilon^2) \quad (2.11)$$

where  $\nabla_b F$  is given by (2.5). Equation (2.11) clearly shows that if

$$\tilde{b} = -\frac{\nabla_b F(b, Q)}{|\nabla_b F(b, Q)|} \quad (2.12)$$

then (2.10) is satisfied. Equality occurs in (2.10) at the minimum when  $\nabla_b F(b^*, Q^*) = 0$ , where  $b^*$  is the optimum value of the design variables and  $Q^*$  is the corresponding value of the state variables that satisfies the state equations. Therefore, to obtain the descent direction, the gradient of  $F$  must be evaluated. The efficient and accurate evaluation of this gradient is one of the important but difficult steps in any minimization scheme. The formula for the gradient of  $F$ , given by (2.5), is not a very useful one because, in general,  $dQ/db$  is difficult to determine. However, by using the adjoint method, this difficulty can be easily overcome. This method is outlined in section 2.3.

### The Step size

Once the descent direction has been determined, the next step is to evaluate the step size  $\varepsilon$ . One approach is to do a line search. The objective of the line search is to find  $\varepsilon$  such that  $\|\nabla_b F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q})\|^2$  is a minimum. That is,

$$\frac{\partial \|\nabla_b F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q})\|^2}{\partial \varepsilon} = 0 \quad (2.13)$$

If we use a Taylor series expansion, then we can write

$$\begin{aligned} & \|\nabla_b F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q})\|^2 \\ &= \|\nabla_b F(b, Q) + \varepsilon \nabla_b^2 F(b, Q) \tilde{b} + \varepsilon^2 C + O(\varepsilon^3)\|^2 \\ &= [\nabla_b F(b, Q)]^T \nabla_b F(b, Q) + 2\varepsilon [\nabla_b F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b} \\ &+ \varepsilon^2 \left\{ \tilde{b}^T [\nabla_b^2 F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b} + 2[\nabla_b F(b, Q)]^T C \right\} \\ &+ O(\varepsilon^3) \end{aligned} \quad (2.14)$$

where  $C$  denotes the  $O(\varepsilon^2)$  term of the expansion. Note that  $\nabla_b^2 F$  includes the variation with respect to  $Q$ . If we set the derivative with respect to  $\varepsilon$  on the right-hand side of (2.14) equal to 0 and solve for  $\varepsilon$ , then

$$\varepsilon = -\frac{[\nabla_b F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b}}{\tilde{b}^T [\nabla_b^2 F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b} + 2[\nabla_b F(b, Q)]^T C} + O(\varepsilon^2) \quad (2.15)$$

Near the minimum, because  $\nabla_b F$  is small, the second term in the denominator is negligible in comparison with the first term; hence, it is dropped. Therefore, if we also neglect the  $O(\varepsilon^2)$  terms in (2.15), then the step size becomes

$$\varepsilon = -\frac{[\nabla_b F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b}}{\tilde{b}^T [\nabla_b^2 F(b, Q)]^T \nabla_b^2 F(b, Q) \tilde{b}} \quad (2.16)$$

where  $\nabla_b^2 F$  is a symmetric matrix and is often referred to as the Hessian. Computation of the Hessian is expensive; the cost is proportional to the number of design variables. However,  $\nabla_b^2 F \tilde{b}$  can be evaluated relatively easily with finite differences as follows:

$$\nabla_b^2 F(b, Q) \tilde{b} = \frac{\nabla_b F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q}) - \nabla_b F(b, Q)}{\varepsilon} \quad (2.17)$$

where  $\varepsilon$  is a trial perturbation. To find the step size, the design variables are perturbed with an arbitrarily small  $\varepsilon$ , and the new values of the state variables that satisfy the state equations and the side constraints are determined. Next, the new gradient  $\nabla_b F(b + \varepsilon \tilde{b}, Q + \varepsilon \tilde{Q})$  is evaluated, followed by  $\nabla_b^2 F(b, Q) \tilde{b}$ . Then, the step size is determined with (2.16).

### 2.3 The Adjoint Method

As stated earlier, the efficient and accurate evaluation the gradient of  $F$  is one of the important but difficult steps in any minimization scheme. The most elegant way of determining this gradient is to use the adjoint method. The adjoint equations, also referred to as the costate equations, can be derived as follows. In the following derivation and in the rest of the paper, we assume that no side constraints exist. For a small change  $\varepsilon \tilde{\mathbf{b}}$  in  $\mathbf{b}$  and a corresponding change  $\varepsilon \tilde{\mathbf{Q}}$  in  $\mathbf{Q}$  that satisfies the state equations (2.2), we can show that

$$\varepsilon \tilde{\mathbf{b}}^T \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T + \varepsilon \tilde{\mathbf{Q}}^T \left( \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^T + O(\varepsilon^2) = 0 \quad (2.18)$$

With (2.5), we can write

$$\varepsilon \tilde{\mathbf{b}}^T \nabla_b F = \varepsilon \tilde{\mathbf{b}}^T \frac{\partial F}{\partial \mathbf{b}} + \varepsilon \tilde{\mathbf{Q}}^T \frac{\partial F}{\partial \mathbf{Q}} \quad (2.19)$$

where

$$\tilde{\mathbf{Q}} = \frac{d\mathbf{Q}}{d\mathbf{b}} \tilde{\mathbf{b}} \quad (2.20)$$

If we add a term, which is the product of (2.18) and an arbitrary multiplier  $\lambda$ , to (2.19), then we get

$$\begin{aligned} \varepsilon \tilde{\mathbf{b}}^T \nabla_b F &= \varepsilon \tilde{\mathbf{b}}^T \frac{\partial F}{\partial \mathbf{b}} + \varepsilon \tilde{\mathbf{Q}}^T \frac{\partial F}{\partial \mathbf{Q}} \\ &+ \varepsilon \left[ \tilde{\mathbf{b}}^T \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T + \tilde{\mathbf{Q}}^T \left( \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^T \right] \lambda + O(\varepsilon^2) \end{aligned} \quad (2.21)$$

The arbitrary multiplier  $\lambda$  is often referred to as either the Lagrange multiplier, the costate variable, or the adjoint variable. If we rearrange (2.21), then we get

$$\begin{aligned} \varepsilon \tilde{\mathbf{b}}^T \nabla_b F &= \varepsilon \tilde{\mathbf{b}}^T \left[ \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{b}} \right] \\ &+ \varepsilon \tilde{\mathbf{Q}}^T \left[ \left( \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{Q}} \right] + O(\varepsilon^2) \end{aligned} \quad (2.22)$$

If we choose  $\lambda$  such that

$$\left( \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{Q}} = 0 \quad (2.23)$$

then (2.22) becomes

$$\varepsilon \tilde{\mathbf{b}}^T \nabla_b F = \varepsilon \tilde{\mathbf{b}}^T \left[ \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{b}} \right] + O(\varepsilon^2) \quad (2.24)$$

Equation (2.23) is the set adjoint equations or the costate equations. The adjoint equations are similar to the linearized form of the state equations. They include the adjoint boundary conditions that correspond to the boundary conditions of the state equations. If we neglect the  $O(\varepsilon^2)$  terms of (2.24), then the gradient of  $F$  can be written as

$$\nabla_b F = \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{b}} \quad (2.25)$$

The gradient of  $F$  given by (2.25) is much easier to evaluate than the one given by (2.5). By introducing the Lagrange multiplier, the need to evaluate  $\mathbf{Q}$  has been eliminated. The

adjoint equations form an additional set of necessary conditions that must be satisfied at the minimum. In summary, the necessary conditions that must be satisfied at the minimum are

$$\begin{aligned} \mathbf{R}[\mathbf{b}, \mathbf{Q}(\mathbf{b})] &= 0 \\ \left( \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{Q}} &= 0 \\ \nabla_b F &= \left( \frac{\partial \mathbf{R}}{\partial \mathbf{b}} \right)^T \lambda + \frac{\partial F}{\partial \mathbf{b}} = 0 \end{aligned} \quad (2.26)$$

The derivation presented above is for a general constrained minimization problem. In the following section, the adjoint equations and the gradient of the cost function are derived for a specific set of state equations and a cost function.

## 3. DESIGN OF OPTIMAL AIRFOIL SHAPES

The design of optimal airfoil shapes is a constrained minimization problem. The objective is to find the optimal shape of the airfoil that will minimize a cost function  $F$  subject to the state equation of the flow field and side constraints.

### 3.1 The State Equations

The analysis problem, defined by the state equation, consists of finding the flow over a specified shape for a given free-stream Mach number and angle of attack. In order to focus on the optimization procedure, the flow model considered is the subsonic potential flow over an airfoil profile.

Consider the steady irrotational flow past a two-dimensional airfoil.<sup>15,16</sup> The governing equation of the flow field, known as the full potential equation, is

$$\text{div}(\rho \nabla \phi) = 0 \quad (3.1)$$

The boundary condition on the airfoil is

$$\nabla \phi \cdot \mathbf{n} = 0 \quad (3.2)$$

At infinity the boundary condition is

$$\nabla \phi = \mathbf{U}_\infty \quad (3.3)$$

For the Kutta condition, the circulation  $\Gamma$  around the airfoil is such that

$$\text{the velocity at the trailing edge is finite and continuous} \quad (3.4)$$

In these equations,  $\phi = \phi(x, y)$  is the full velocity potential,  $\rho = \rho(\phi)$  is the density,  $\mathbf{n}$  is the unit normal, and  $\mathbf{U}_\infty$  is the free-stream velocity. The density  $\rho$  is given by

$$\rho = \left[ 1 - \frac{\gamma - 1}{2} M_\infty^2 (|\nabla \phi|^2 - 1) \right]^{\frac{1}{\gamma - 1}} \quad (3.5)$$

where  $M_\infty$  is the free-stream Mach number and  $\gamma$  is the ratio of specific heats. If  $\zeta$  is the angle of attack of the airfoil, then the free-stream velocity is given by

$$\mathbf{U}_\infty = U_\infty [\cos(\zeta) \mathbf{i} + \sin(\zeta) \mathbf{j}] \quad (3.6)$$

where  $\mathbf{i}$  and  $\mathbf{j}$  are the unit vectors in the  $x$  and  $y$  directions, respectively.

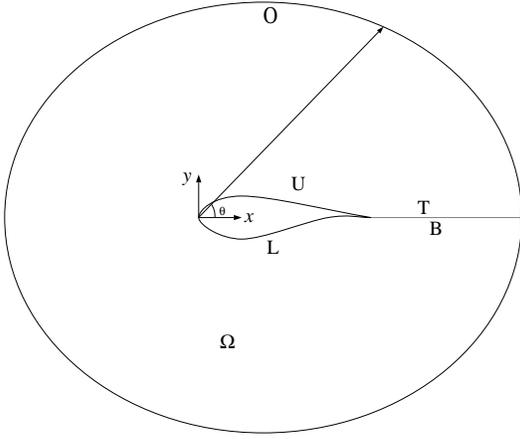


Figure 1. Computational domain.

### 3.2 The Computational Domain

The computational domain is shown in Figure 1. The interior of the flow field is denoted by  $\Omega$ ; the upper and lower surfaces of the airfoil are denoted by  $U$  and  $L$ , respectively. The far-field boundary, located at a finite distance from the airfoil (30 to 50 airfoil chord lengths) is denoted by  $O$ . To impose the Kutta condition around the airfoil, an artificial boundary or cut that begins at the airfoil and extends to the far field is introduced. A jump in potential that is equal to  $\Gamma$  is allowed across the cut. For convenience, this cut is chosen to emanate from the trailing edge of the airfoil. The top and bottom sides of the cut are denoted by  $T$  and  $B$ , respectively. The jump across the cut can be written as

$$\phi^T - \phi^B = \Gamma \quad (3.7)$$

The value of the  $\Gamma$  is determined by requiring that the velocity perpendicular to the trailing edge bisector be equal to 0 at the trailing edge. A good approximation for  $\Gamma$  is given by

$$\Gamma = \phi_{t.e.}^T - \phi_{t.e.}^B \quad (3.8)$$

where *t.e.* refers to the trailing edge of the airfoil (refer to Appendix C for details). To satisfy mass conservation across the cut, derivatives of the potential normal to the cut are required to be continuous.

At the far-field boundary, the circulation modifies the velocity as follows:

$$\nabla \phi \cdot \mathbf{n} = U_\infty \cdot \mathbf{n} + \frac{\Gamma}{2\pi} \nabla \Theta \cdot \mathbf{n} \quad (3.9)$$

where

$$\Theta = 2\pi - \tan^{-1} \left( \sqrt{1 - M_\infty^2} \tan \theta \right) \quad (3.10)$$

and  $\theta$  is the angular position of a far-field point. For convenience,  $\mathbf{n}$  is the unit normal on the boundary. The far-field boundary condition given by (3.9) is consistent with the infinity condition stated by (3.3).

### 3.3 The Design Variables

The airfoil is represented as follows:

$$y^U = \sum_{k=1}^K \alpha_k^U f_k(x) \quad (0 \leq x \leq 1) \quad (3.11)$$

$$y^L = \sum_{k=1}^K \alpha_k^L f_k(x)$$

where  $\alpha_k^U$  and  $\alpha_k^L$  are the amplitudes of the shape functions  $f_k$  on the upper and lower surfaces of the airfoil, respectively. The design variables  $\alpha_k$  must be determined to obtain the optimal shape of the airfoil. Let  $\alpha$  denote a vector whose elements are the design variables. That is,

$$\alpha = \left[ \alpha_1^U, \alpha_2^U, \dots, \alpha_K^U, \alpha_1^L, \alpha_2^L, \dots, \alpha_K^L \right]^T \quad (3.12)$$

The functionality of the shape functions will be presented later.

### 3.4 The Optimization Problem

The model problem chosen is the design of an airfoil shape that can match a given target potential. Given a target potential distribution  $\phi_0$  around an airfoil, the objective is to find  $\alpha$  that will minimize

$$F[\alpha, \phi(\alpha)] \equiv \int_{U+L} (\phi - \phi_0)^2 d\sigma \quad (3.13)$$

subject to the state equations, where  $d\sigma$ , which is an element of the airfoil, can be written as

$$d\sigma^2 = dx^2 + dy^2 \quad (3.14)$$

Note that the choice of this particular cost function does not make it an inverse-design problem. Unlike inverse-design problems, the minimization is done over a finite number of design variables. This approach also can be used, for example, to find the optimal shape of an airfoil that has the minimum  $D/L$  (Drag/Lift) subject to geometric and aerodynamic constraints.

To make the presentation of the derivation of the adjoint equations simple and easy to understand, the flow is assumed to be incompressible (i.e.,  $M_\infty = 0$ ); therefore,  $\rho = 1$ . In this case, the full potential equation reduces to the Laplace equation. Also, no side constraints are considered in this derivation. Therefore, the specific optimization problem considered here is

$$\min_{\alpha, \phi} \int_{U+L} (\phi - \phi_0)^2 d\sigma \quad (3.15)$$

subject to

$$\text{div}(\nabla \phi) = 0 \quad \text{in } \Omega \quad (3.16a)$$

$$\nabla \phi \cdot \mathbf{n} = 0 \quad \text{on the airfoil} \quad (3.16b)$$

$$\nabla \phi \cdot \mathbf{n} = U_\infty \cdot \mathbf{n} + \frac{\Gamma}{2\pi} \nabla \Theta \cdot \mathbf{n} \quad \text{in the far field} \quad (3.16c)$$

$$\phi^T - \phi^B = \Gamma \quad \text{along the cut} \quad (3.16d)$$

where  $\Gamma$  is given by (3.8).



Because  $\nabla\tilde{\phi}$  is continuous across the cut and  $\mathbf{n}$  points in opposite directions along the top and bottom boundaries of the cut, we can write

$$\int_{T+B} \lambda (\nabla\tilde{\phi} \cdot \mathbf{n}) d\tau = \int_{\text{Cut}} (\lambda^T - \lambda^B) (\nabla\tilde{\phi} \cdot \mathbf{n}) d\tau \quad (3.28)$$

If we assume that  $\nabla\lambda$  is continuous across the cut, then we can write

$$\int_{T+B} (\nabla\lambda \cdot \mathbf{n}) \tilde{\phi} d\tau = \tilde{\Gamma} \int_{\text{Cut}} \nabla\lambda \cdot \mathbf{n} d\tau \quad (3.29)$$

If we use (3.28), (3.29), and (3.22b-d), then equation (3.27) can be written as

$$\begin{aligned} \tilde{\alpha}^T \nabla_{\alpha} F &= \int_{\text{U}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad - \int_{\text{L}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad + \int_{\text{U+L}} (\phi - \phi_0)^2 \frac{y_x \tilde{y}_x}{1+y_x^2} d\sigma + \int_{\text{U+L}} 2(\phi - \phi_0) \tilde{\phi} d\sigma \\ &\quad + \iint_{\Omega} \text{div}(\nabla\lambda) \tilde{\phi} d\Omega \\ &\quad - \int_{\text{U+L}} \lambda \nabla(\tilde{y} \nabla\phi \cdot \mathbf{t}) \cdot \mathbf{i} d\sigma + \int_{\text{U+L}} (\nabla\lambda \cdot \mathbf{n}) \tilde{\phi} d\sigma \\ &\quad - \int_{\text{Cut}} (\lambda^T - \lambda^B) (\nabla\tilde{\phi} \cdot \mathbf{n}) d\tau + \tilde{\Gamma} \int_{\text{Cut}} \nabla\lambda \cdot \mathbf{n} d\tau \\ &\quad - \frac{\tilde{\Gamma}}{2\pi} \int_{\text{O}} \lambda (\nabla\Theta \cdot \mathbf{n}) d\tau + \int_{\text{O}} (\nabla\lambda \cdot \mathbf{n}) \tilde{\phi} d\tau \end{aligned} \quad (3.30)$$

If we substitute for  $\tilde{\Gamma}$  from (3.23) and rearrange, then (3.30) becomes

$$\begin{aligned} \tilde{\alpha}^T \nabla_{\alpha} F &= \int_{\text{U}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad - \int_{\text{L}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad + \int_{\text{U+L}} (\phi - \phi_0)^2 \frac{y_x \tilde{y}_x}{1+y_x^2} d\sigma - \int_{\text{U+L}} \lambda \nabla(\tilde{y} \nabla\phi \cdot \mathbf{t}) \cdot \mathbf{i} d\sigma \\ &\quad + \iint_{\Omega} \text{div}(\nabla\lambda) \tilde{\phi} d\Omega \\ &\quad + \int_{\text{U+L}} [\nabla\lambda \cdot \mathbf{n} + 2(\phi - \phi_0)] \tilde{\phi} d\sigma \\ &\quad + (\tilde{\phi}_{t.e.}^{\text{U}} - \tilde{\phi}_{t.e.}^{\text{L}}) \left( \int_{\text{Cut}} \nabla\lambda \cdot \mathbf{n} d\tau - \frac{1}{2\pi} \int_{\text{O}} \lambda \nabla\Theta \cdot \mathbf{n} d\tau \right) \\ &\quad + \int_{\text{O}} (\nabla\lambda \cdot \mathbf{n}) \tilde{\phi} d\tau - \int_{\text{Cut}} (\lambda^T - \lambda^B) (\nabla\tilde{\phi} \cdot \mathbf{n}) d\tau \end{aligned} \quad (3.31)$$

We choose  $\lambda$  such that

$$\begin{aligned} \text{div}(\nabla\lambda) &= 0 && \text{in } \Omega \\ \nabla\lambda \cdot \mathbf{n} + 2(\phi - \phi_0) - \Upsilon\delta(x - x_{t.e.}) &= 0 && \text{on L} \\ \nabla\lambda \cdot \mathbf{n} + 2(\phi - \phi_0) + \Upsilon\delta(x - x_{t.e.}) &= 0 && \text{on U} \\ \nabla\lambda \cdot \mathbf{n} &= 0 && \text{in the far field} \\ \lambda^T - \lambda^B &= 0 && \text{along the cut} \end{aligned} \quad (3.32)$$

where

$$\Upsilon = \int_{\text{Cut}} \nabla\lambda \cdot \mathbf{n} d\tau - \frac{1}{2\pi} \int_{\text{O}} \lambda \nabla\Theta \cdot \mathbf{n} d\tau \quad (3.33)$$

and  $\delta$  denotes the Dirac delta function (*t.e.* stands for trailing edge of the airfoil). Equations (3.32) are the adjoint equation and its boundary conditions (also called the costate equations). These equations are similar to the linearized state equations. The size of the system is the same as the size of the state equations and can be solved with the same technique used to solve the state equations.

Because  $\text{div}(\nabla\lambda) = 0$  in  $\Omega$ , we obtain the following from the divergence theorem:

$$\int_{\tau} \nabla\lambda \cdot \mathbf{n} d\tau = 0 \quad (3.34)$$

Therefore, for (3.32) to have a solution, we can show that

$$\int_{\text{U+L}} (\phi - \phi_0) d\sigma \equiv 0 \quad (3.35)$$

Equation (3.16) clearly shows that a constant can be added to  $\phi$ . We can choose this constant  $\phi_c$  such that

$$\int_{\text{U+L}} (\phi + \phi_c - \phi_0) d\sigma = 0 \quad (3.36)$$

Therefore,

$$\phi_c = -\frac{\int_{\text{U+L}} (\phi - \phi_0) d\sigma}{\int_{\text{U+L}} d\sigma} \quad (3.37)$$

### 3.6 The Gradient of $F$

If (3.32) is substituted into (3.31), then it reduces to

$$\begin{aligned} \tilde{\alpha}^T \nabla_{\alpha} F &= \int_{\text{U}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad - \int_{\text{L}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1+y_x^2}} \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad + \int_{\text{U+L}} (\phi - \phi_0)^2 \frac{y_x \tilde{y}_x}{1+y_x^2} d\sigma \\ &\quad - \int_{\text{U+L}} \lambda \nabla(\tilde{y} \nabla\phi \cdot \mathbf{t}) \cdot \mathbf{i} d\sigma \end{aligned} \quad (3.38)$$

If we integrate the last integral by parts, then we get

$$\begin{aligned} &- \int_{\text{U+L}} \lambda \nabla(\tilde{y} \nabla\phi \cdot \mathbf{t}) \cdot \mathbf{i} d\sigma \\ &= \int_{\text{U+L}} (\tilde{y} \nabla\phi \cdot \mathbf{t}) \nabla\lambda \cdot \mathbf{i} d\sigma + \int_{\text{U+L}} \frac{y_x y_{xx} \tilde{y}}{1+y_x^2} \lambda \nabla\phi \cdot \mathbf{t} d\sigma \\ &= \int_{\text{U}} \frac{\tilde{y}}{\sqrt{1+y_x^2}} (-y_x \nabla\lambda \cdot \mathbf{n} + \nabla\lambda \cdot \mathbf{t}) \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad + \int_{\text{L}} \frac{\tilde{y}}{\sqrt{1+y_x^2}} (y_x \nabla\lambda \cdot \mathbf{n} - \nabla\lambda \cdot \mathbf{t}) \nabla\phi \cdot \mathbf{t} d\sigma \\ &\quad + \int_{\text{U+L}} \frac{y_x y_{xx} \tilde{y}}{1+y_x^2} \lambda \nabla\phi \cdot \mathbf{t} d\sigma \end{aligned} \quad (3.39)$$

If (3.39) is substituted into (3.38) and rearranged, then we can write

$$\begin{aligned}
& \tilde{\alpha}^T \nabla_{\alpha} F \\
&= \int_{\bar{U}} [(\phi - \phi_0)^2 \tilde{y}_x + \lambda \nabla \phi \cdot \mathbf{t} y_{xx} \tilde{y}] \frac{y_x}{1 + y_x^2} d\sigma \\
&+ \int_{\bar{U}} [2(\phi - \phi_0) y_x - \nabla \lambda \cdot \mathbf{n} y_x + \nabla \lambda \cdot \mathbf{t}] \frac{\nabla \phi \cdot \mathbf{t} \tilde{y}}{\sqrt{1 + y_x^2}} d\sigma \\
&+ \int_{\bar{L}} [(\phi - \phi_0)^2 \tilde{y}_x + \lambda \nabla \phi \cdot \mathbf{t} y_{xx} \tilde{y}] \frac{y_x}{1 + y_x^2} d\sigma \\
&- \int_{\bar{L}} [2(\phi - \phi_0) y_x - \nabla \lambda \cdot \mathbf{n} y_x + \nabla \lambda \cdot \mathbf{t}] \frac{\nabla \phi \cdot \mathbf{t} \tilde{y}}{\sqrt{1 + y_x^2}} d\sigma
\end{aligned} \tag{3.40}$$

If we substitute for  $\tilde{y}$  from (3.11), then (3.40) can be written as

$$\tilde{\alpha}^T \nabla_{\alpha} F = \sum_{k=1}^K \tilde{\alpha}_k^U \mu_k^U + \sum_{k=1}^K \tilde{\alpha}_k^L \mu_k^L \tag{3.41}$$

where

$$\begin{aligned}
\mu_k^U &= \int_{\bar{U}} [(\phi - \phi_0)^2 (f_k)_x + \lambda \nabla \phi \cdot \mathbf{t} y_{xx} f_k] \frac{y_x}{1 + y_x^2} d\sigma \\
&+ \int_{\bar{U}} [2(\phi - \phi_0) y_x - \nabla \lambda \cdot \mathbf{n} y_x + \nabla \lambda \cdot \mathbf{t}] \frac{\nabla \phi \cdot \mathbf{t} f_k}{\sqrt{1 + y_x^2}} d\sigma
\end{aligned} \tag{3.42}$$

and

$$\begin{aligned}
\mu_k^L &= \int_{\bar{L}} [(\phi - \phi_0)^2 (f_k)_x + \lambda \nabla \phi \cdot \mathbf{t} y_{xx} f_k] \frac{y_x}{1 + y_x^2} d\sigma \\
&- \int_{\bar{L}} [2(\phi - \phi_0) y_x - \nabla \lambda \cdot \mathbf{n} y_x + \nabla \lambda \cdot \mathbf{t}] \frac{\nabla \phi \cdot \mathbf{t} f_k}{\sqrt{1 + y_x^2}} d\sigma
\end{aligned} \tag{3.43}$$

Equations (3.42) and (3.43) are the components of the gradient of  $F$ . When  $\phi$  satisfies the state equations (3.16) and  $\lambda$  satisfies the costate equations (3.32), then the components of the gradient of  $F$  can be evaluated with (3.42) and (3.43). Because  $\nabla_{\alpha} F = 0$  at the minimum, we can clearly see that

$$\left. \begin{aligned} \mu_k^U &= 0 \\ \mu_k^L &= 0 \end{aligned} \right\} \text{ for } k = 1, 2, \dots, K \tag{3.44}$$

### 3.7 A Design Strategy

Figure 3 shows a typical design strategy. In this process, at some initial conditions the state and adjoint equations are solved, and the gradient of  $F$  is computed. If the gradient is equal to 0, then a minimum has been reached and the iteration is terminated; otherwise, the new descent direction  $\tilde{\alpha}$  and the step size  $\varepsilon$  are computed, and the design variables are updated. The iteration is repeated until the gradient vanishes. The cost of this strategy can be estimated as follows. Let the cost of solving the state equations be equal to  $K$ . The cost of solving the adjoint equation is at most equal to  $K$ . Let the number of design iterations required be  $N$ . Therefore, the total cost of doing the optimal design is approximately  $2KN$  with  $N$ , at best, equal to the number of design variables. In practice, especially for nonlinear problems,  $N$  is many times the number of design variables. A factor of 100 is not unrealistic. One way to bring the total design cost down is to reduce the magnitude of  $K$ . One of the most practical and proven ways of achieving this is by using multigrid. Here, a multigrid scheme

is used to relax the state and adjoint equations. At the end of one or several multigrid cycles, the optimizer is called and the design variables are updated. In this process, the design variables are updated only on the finest grid. A schematic of this strategy is shown in Figure 4.

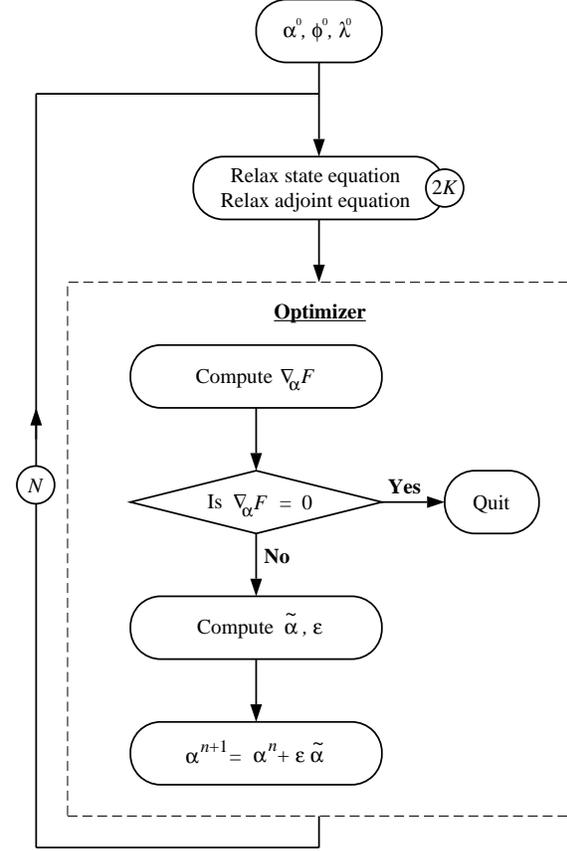


Figure 3. A design strategy flowchart.

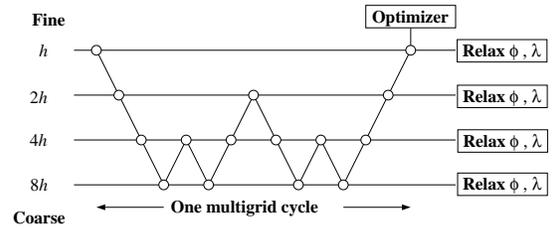


Figure 4. A multigrid strategy.

## 4. THE ONE-SHOT METHOD

The *One-Shot* method goes one step further by embedding the design process within the multigrid cycles. This method essentially makes  $N = 1$ . Thus, the cost of optimal design is approximately equal to  $2K$ . In this method, high efficiency is obtained by exploiting two key phenomena. The first one is the ability of multigrid to efficiently reduce high-frequency components of the error due to a perturbation, and the second one is the nature of propagation of perturbations in a flow field. These phenomena are explained below.

#### 4.1 Multigrid Efficiency

In any relaxation (smoothing) process, the high-frequency error components of the space discretization operator of the differential equation under consideration are generally damped in a few iterations. The low-frequency components are the slowest to be damped. Consider a one-dimensional domain of length  $L$  discretized into  $N$  cells of uniform grid spacing  $h = L/N$ , where the grid index ranges from 0 to  $N$ . This grid will be referred to as the  $h$  grid. If we assume periodic boundary conditions, then the error at the  $n$ th grid point can be written in Fourier series as

$$\epsilon_n = \sum_{j=-N}^N A_j e^{i\theta_j n} \quad (4.1)$$

where  $A_j$  is the amplitude of the  $j$ th harmonic and  $i = \sqrt{-1}$ . The phase angle  $\theta$  can be written as

$$\theta_j = \frac{j\pi}{N} \quad (4.2)$$

The phase angle covers the domain  $(-\pi, \pi)$  in increments of  $\pi/N$ . The value  $|\theta| = \pi$  corresponds to the highest frequency that is visible on this grid, namely the frequency of wavelength  $2h$ . If a coarse grid ( $H$  grid) is constructed by removing every other grid point of the  $h$  grid, then the highest frequency that is visible on this grid corresponds to  $|\theta| = \pi/2$  (i.e., the frequency of wavelength  $4h \equiv 2H$ ). Therefore, the frequencies that correspond to  $\pi/2 < |\theta| \leq \pi$  and are visible on the  $h$  grid cannot be represented on the  $H$  grid. These frequencies are considered to be high frequencies on the  $h$  grid and the relaxation scheme can damp these frequencies in a few iterations. The remaining frequencies in the spectrum, which correspond to  $0 \leq |\theta| \leq \pi/2$  and are well represented on the  $H$  grid, are referred to as low frequencies on the  $h$  grid. The frequencies that are visible on the  $H$  grid can also be separated into high and low frequencies, based on how well they are represented by the next coarsest grid. The high frequencies that correspond to the  $H$  grid can be damped quickly by a few iterations of the relaxation scheme on this grid.

In the multigrid method,<sup>17,16</sup> high efficiency is obtained by relaxing the discretized equation on successively coarser grids, where the high-frequency error components that correspond to each grid are damped efficiently. In the design process, high efficiency is obtained by changing only those design variables that produce high-frequency perturbations in the flow field on any grid. Therefore, the basic premise of the *One Shot* method, on any grid, is to *make changes in the design variables that produce high-frequency perturbations* in the flow field.

#### 4.2 The Effect of Airfoil Perturbation on the Flow Field

The other phenomenon that is exploited by the *One-Shot* method has to do with the way in which a disturbance is propagated in a flow field. In a subsonic flow, for example, a smooth perturbation is propagated through the entire flow field and a high-frequency perturbation is felt only in a small neighborhood around the source of the perturbation. That is, high-frequency components of the perturbation decay rapidly away from the source. This phenomenon is illustrated in the following analysis.

Consider the small-disturbance potential equation in the half-space  $0 \leq y < \infty$ ,  $-\infty < x < \infty$ . If the flow is incompressible, the governing equation is

$$\nabla^2 \phi = 0 \quad (4.3)$$

and the boundary condition applied at  $y = 0$  is

$$\frac{\partial \phi}{\partial y} = \frac{\partial f}{\partial x} \quad (4.4)$$

where  $f(x)$  is the shape of the boundary over which the flow must be determined. If  $\phi + \tilde{\phi}$  is the potential due to a change in shape to  $f + \tilde{f}$ , the governing equation for change in potential  $\tilde{\phi}$  is

$$\nabla^2 \tilde{\phi} = 0 \quad (4.5)$$

and the boundary condition at  $y = 0$  is

$$\frac{\partial \tilde{\phi}}{\partial y} = \frac{\partial \tilde{f}}{\partial x} \quad (4.6)$$

Let

$$\frac{\partial \tilde{f}}{\partial x} = e^{i\omega x} \quad (4.7)$$

where  $\omega$  is the frequency of the perturbation. A solution to the governing equation (4.5) that satisfies the boundary condition is

$$\tilde{\phi} = e^{-|\omega|y} e^{i\omega x} \quad (4.8)$$

The magnitude of  $\tilde{\phi}$  is

$$|\tilde{\phi}| = e^{-|\omega|y} \quad (4.9)$$

Figure 5, which is the plot of (4.9) for a few select frequencies,

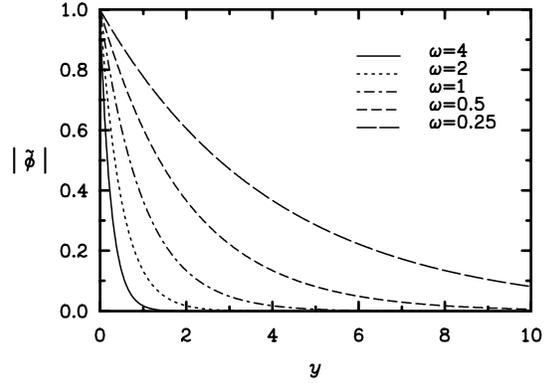


Figure 5.  $|\tilde{\phi}|$  versus  $y$ .

shows that the region where  $\tilde{\phi}$  is large becomes thinner as the frequency increases. Let  $y^*$  be a location where  $\tilde{\phi}$  is less than some small  $\epsilon$ . That is,

$$|\tilde{\phi}(\omega, y^*)| < \epsilon = e^{-\beta} \quad (4.10)$$

If we substitute for  $\tilde{\phi}$ , then

$$e^{-|\omega|y^*} = e^{-\beta} \quad (4.11)$$

Therefore,

$$y^* = \frac{-\ln(\varepsilon)}{|\omega|} \quad (4.12)$$

Equation (4.12) clearly shows that as the frequency of the perturbation  $\omega$  increases  $y^*$  decreases. Table 1 shows  $y^*$  for a few select frequencies when  $\varepsilon = 10^{-4}$ . For the discrete

Table 1.  $y^*$  versus  $\omega$

$\omega$	1/4	1/2	1	2	4
$y^*$	27.6	13.8	6.9	3.5	1.7

problem, (4.9) can be written as

$$|\tilde{\phi}| = e^{-(|\omega|h)(y/h)} = e^{-|\theta|(j-1)} \quad (j = 1, 2, \dots, J+1) \quad (4.13)$$

where  $\pi/J \leq \theta \leq \pi$  is the frequency scaled to the grid spacing  $h$ . Figure 6 shows the response to different frequencies for the discrete problem. Table 2 shows the grid location  $j^*$  beyond which  $|\tilde{\phi}| \leq 10^{-4}$ . It shows that the high frequency perturbations are significantly damped by about the fifth grid point ( $j = 0$  is the first grid point).

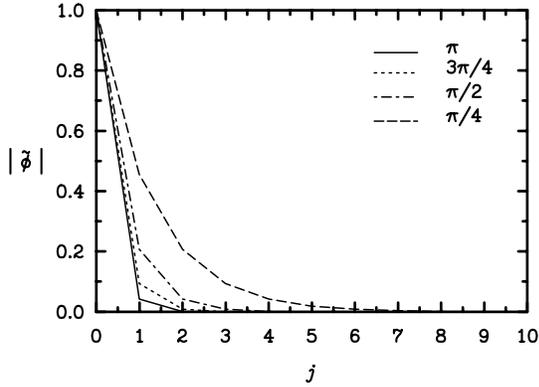


Figure 6.  $|\tilde{\phi}|$  versus  $j$ .

Table 2.  $j^*$  versus  $\theta$

$\theta$	$\pi/4$	$\pi/2$	$3\pi/4$	$\pi$
$j^*$	8.8	4.4	2.9	2.2

In the *One-Shot* method, a shape function is perturbed on a grid where it produces high-frequency error components. As described above, these errors penetrate only a small distance into the flow field. Hence, they can be quickly damped by a few relaxations of the discrete equations in a small neighborhood around the airfoil.

### 4.3 The Shape Functions

As presented earlier (section 3.3), the airfoil is represented as follows:

$$y^{U,L} = \sum_{k=1}^K \alpha_k^{U,L} f_k(x) \quad (4.14)$$

where  $\alpha_k^U$  and  $\alpha_k^L$  are the design variables and  $f_k$  are the shape functions. As explained in the previous two sections, to obtain high design efficiency, the changes in the design variables on a grid should produce nonsmooth (high-frequency) perturbations in the flow field. This is achieved by using a set of orthonormal functions as shape functions. Orthonormal functions are increasingly oscillatory. Each of them is assigned to a grid where a change in the amplitudes causes nonsmooth perturbations in the flow field. Often, basis functions that correspond to some known airfoil shape must be used. If these functions are not orthonormal, the corresponding orthonormal functions can then be determined by a Gram-Schmidt process.<sup>18</sup> A Gram-Schmidt procedure for orthonormalization can be developed with the property of orthonormal functions, namely,

$$\int_0^1 f_m(x) f_n(x) dx = 0 \quad (m \neq n) \quad (4.15)$$

$$\int_0^1 f_m^2(x) dx = 1$$

Let  $g_k(x)$  be the functions that are not orthonormal. First, the orthogonal set  $\bar{f}_k(x)$  is found from the following relations:

$$\begin{aligned} \bar{f}_1(x) &= g_1(x) \\ \bar{f}_2(x) &= g_2(x) + a_{21} \bar{f}_1(x) \\ &\vdots \\ \bar{f}_k(x) &= g_k(x) + \sum_{m=1}^{k-1} a_{km} \bar{f}_m(x) \end{aligned} \quad (4.16)$$

where

$$a_{km} = -\frac{\int_0^1 g_k(x) \bar{f}_m(x) dx}{\int_0^1 \bar{f}_m^2(x) dx} \quad (4.17)$$

Finally, the orthonormal functions are found by normalizing  $\bar{f}_k(x)$  as follows:

$$f_k = \frac{\bar{f}_k(x)}{\sqrt{\int_0^1 \bar{f}_k^2(x) dx}} \quad (4.18)$$

The Gram-Schmidt process described above can be programmed in symbolic language to find the expressions for  $f_k$ , or it can be implemented by numerical integration, in which case the shape functions are defined as an array of numbers.

As an example, consider the NACA 0012 airfoil, defined by

$$y^U = \sum_{k=1}^4 \beta_k g_k(x) \quad (0 \leq x \leq 1) \quad (4.19)$$

$$y^L = -y^U$$

where  $\beta_k$  and  $g_k$  are given in Table 3. The NACA 0012 shape has been slightly modified to ensure that it closes at the trailing edge. The same shape can be expressed in terms of the orthonormal functions as

$$y^U = \sum_{k=1}^4 \alpha_k f_k(x) \quad (0 \leq x \leq 1) \quad (4.20)$$

$$y^L = -y^U$$

where the orthonormal functions  $f_k$  of the basis functions and their corresponding amplitudes  $\alpha_k$  are given in Table 4. The orthonormal shape functions are shown in Fig. 7. Note that the number of zeros of  $f_k$  is equal to  $k + 1$ .

Table 3. Shape Functions and Amplitudes of NACA 0012

$k$	$\beta_k$	$g_k$
1	0.17814	$\sqrt{x - x}$
2	0.10128	$x(1 - x)$
3	-0.10968	$x^2(1 - x)$
4	0.06090	$x^3(1 - x)$

Table 4. Orthonormal shape functions and amplitudes of NACA 0012

$k$	$\alpha_k \times 10^4$	$f_k$
1	439.474	$5.47723g_1$
2	28.2339	$14.7573(g_2 - .928571g_1)$
3	-5.85699	$54.7884(g_3 - .901236g_2 + .432099g_1)$
4	2.85283	$213.472(g_4 - 1.27406g_3 + .504011g_2 - .164439g_1)$

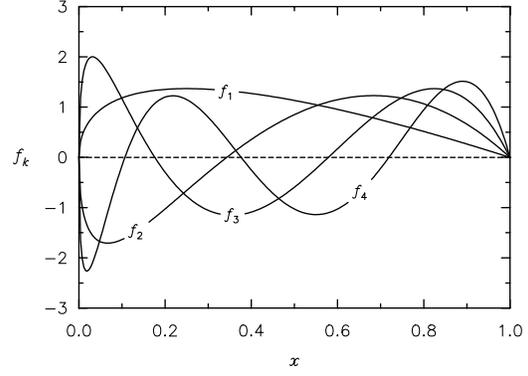


Figure 7. Orthonormal shape functions of NACA 0012 airfoil.

#### 4.4 The One-Shot Design Strategy

In the *One-Shot* method, the optimizer is embedded within the multigrid cycle as shown in Figure 8. The design variables are updated on a level where the corresponding shape functions produce high-frequency error components. In general, the low-frequency shape functions are updated on coarse levels, and higher frequency functions are updated on finer grids. For example, the design variables  $\alpha_1^U$  and  $\alpha_1^L$  are updated on the coarsest grid  $8h$ ;  $\alpha_1^U, \alpha_2^U, \alpha_3^U, \alpha_4^U, \alpha_1^L, \alpha_2^L, \alpha_3^L, \alpha_4^L$  are updated on the next finest grid  $4h$ . Some overlap of the design variables is permitted. Thus,  $\alpha_1^U$  and  $\alpha_1^L$  are updated on grid  $4h$  also. None of the design variables are updated on the finest grid  $h$ . The cost of solving the state or the adjoint equations on a coarse grid is only one-fourth of the cost of solving them on the next finest grid. Because the shape functions are perturbed only on levels where they generate high-frequency errors, a local relaxation around the airfoil is sufficient to damp out the errors, which reduces computing costs. Therefore, the overall cost of the design is dominated by the cost required to solve the state and adjoint equations on the finest grid. The total cost of the design process is approximately two to three times that of one analysis.

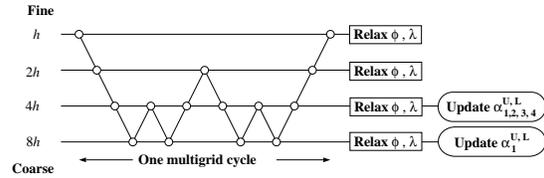


Figure 8. The *One-Shot* strategy.

#### 4.5 The Discretization and Solution Procedure

##### The State Equations

The computational domain is discretized with an O type of grid. The governing equation and its boundary conditions cast in curvilinear coordinates are discretized with the finite-volume approach. The Gauss-Seidel line-relaxation scheme is used to form the tridiagonal systems of equations in both curvilinear coordinate directions. These systems are solved with the Thomas algorithm. Note that the tridiagonal system is periodic in the direction that is around the airfoil. A FAS multigrid scheme is used to accelerate the convergence rate of the solution. The FMG process is used to obtain a good initial solution on the finest grid. The details of the discretization, the relaxation, and the multigrid acceleration are given in Appendix B.

##### The Adjoint Equations

The adjoint equations are discretized and solved in the same manner as the state equations. As in the case of the state equations, a FAS multigrid scheme and the FMG process are used to accelerate the convergence rate of the solution.

##### The Gradient of $F$

The gradient of the cost function involves only quantities on the airfoil. These quantities are discretized in a manner that is consistent with the discretization of the state and adjoint equations. The gradient is transferred to the coarse grid in a FAS manner.

##### Updating the Grid

During the design process, the grid is updated by moving only the grid points close to the airfoil and linearly decaying the change at the airfoil in this neighborhood. The outer boundary of this region is determined as follows. Let

$$y_{\max} = \eta \max(\varepsilon \tilde{y}) \quad (4.23)$$

where  $\eta$  is an arbitrary constant;  $\eta = 10$  in this study. Among the grid lines that go around the airfoil, the one that is nearest to the  $y_{\max}$  location is taken to be the outer boundary of the region within which the grids are changed. The entire grid is regenerated at the beginning of each FMG stage also. With this approach, by the time the FMG process reaches the finest grid, only a few lines around the airfoil must be moved.

## 5. THE RESULTS

### Test Case 1

As our first test problem, we recover the NACA 0012 airfoil shape using the potential distribution obtained from the analysis of NACA 0012 at an angle of attack of  $0^\circ$  and  $M_\infty = 0$  as the target potential  $\phi_0$ . Figure 9 shows the computed  $C_p$  distribution obtained from the analysis run. A five-level W-cycle multigrid with  $128 \times 64$  cells on the finest grid was used. The FMG process was used to obtain a good initial approximation for the finest grid. The analysis converged to machine zero ( $< 10^{-10}$ ) in 10 multigrid cycles.

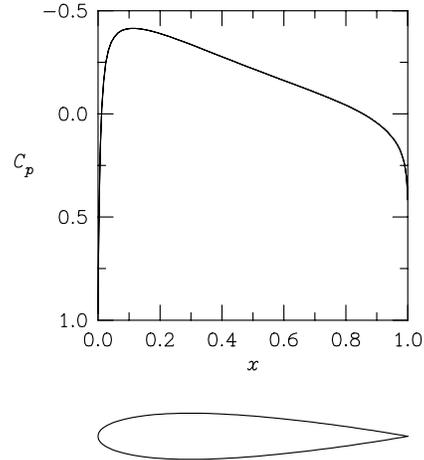


Figure 9. Computed  $C_p$  distribution for NACA 0012.

The design run was similar to the analysis run. During the design process, both the state and costate equations were relaxed at any multigrid level. The shape functions used were the orthonormal functions based on the NACA 0012 shape functions. The design variables were distributed such that on the coarsest level ( $8 \times 4$ ) only  $\alpha_1^U$  and  $\alpha_1^L$  were updated. On the next finest level ( $16 \times 8$ ), all the design variables ( $\alpha_{1,2,3,4}^{U,L}$ ) were updated. None of the design variables were updated on the next three levels, including the finest level. Thus, most of the design overhead was limited to the two coarsest grids. The FMG process was used to obtain a good initial approximation of the shape for the finest grid. Figure 10 shows the results of this run. The residuals of the state and costate equations and the gradient of the cost function reached machine zero in 12 multigrid cycles. The cost function at convergence was equal to  $3 \times 10^{-13}$ , which indicates that NACA 0012 was indeed recovered.

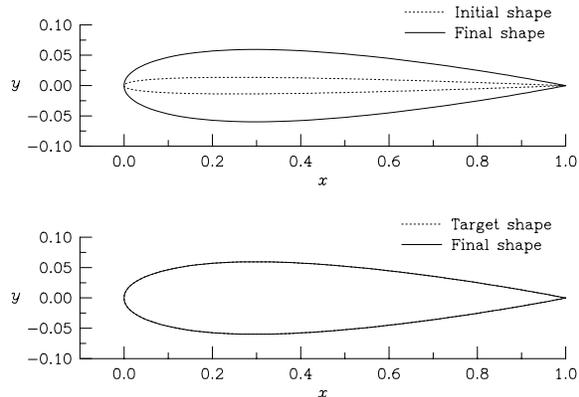


Figure 10. Test case 1.

**Test Case 2**

For test case 2, we selected the airfoil FX 60–126/1, a cambered airfoil whose coordinates are tabulated in Ref. 19. Figure 11 shows the  $C_p$  distribution for this airfoil at an angle of attack of  $0^\circ$  and  $M_\infty = 0$ . This airfoil is not smooth, which is reflected in the computed  $C_p$  distribution. Using this solution as the target, we tried to recover the shape with the NACA 0012 shape functions. Figure 12 shows the resulting shape. Although the designed shape did not fall

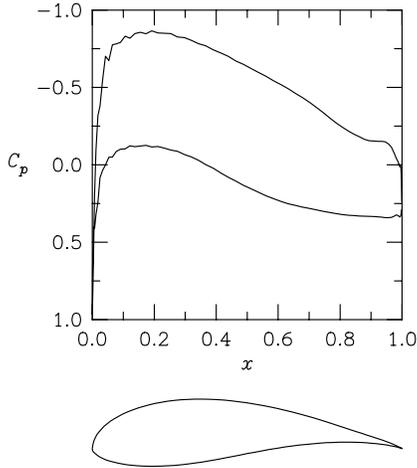


Figure 11. Computed  $C_p$  distribution for FX 60–126/1.

right on top of the target shape, the residuals of the state and costate equations and the gradient of the cost function reached machine zero, which indicates that a minimum was reached. The cost function reached a value of  $6 \times 10^{-9}$ .

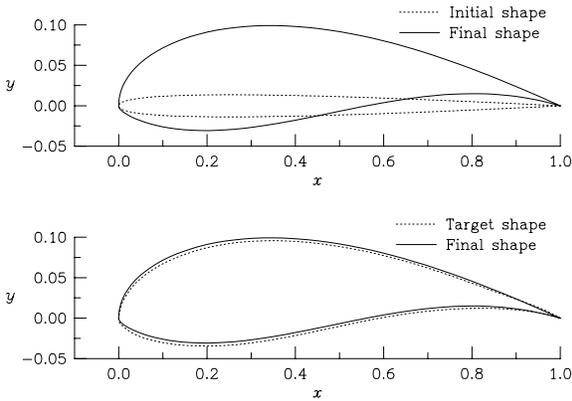


Figure 12. Test case 2.

Next, an experiment was done to see how well the FX 60–126/1 airfoil can be represented with the NACA 0012 shape function. Figure 13 shows the result. The NACA 0012 shape functions clearly do a good job everywhere except near the trailing edge. The reason why the optimum shape in the previous experiment does not correspond to the shape

obtained from the shape fitting is not clear; one reason may be the poor quality of the grid because the airfoil is not smooth.

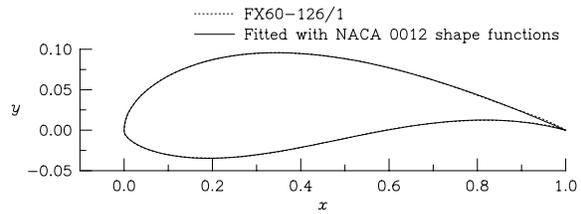


Figure 13. Shape fitting with NACA 0012 shape functions.

**Test Case 3**

A third test was done; this time the fitted airfoil was used to generate the target potential. This shape is very close to the FX 60–126/1 airfoil and is smooth because it is based on smooth shape functions. The result of the design is shown in figure 14. As expected, the final shape fell on top of the target shape. The residuals of the state and costate equations and the gradient of the cost function are shown in figure 15.

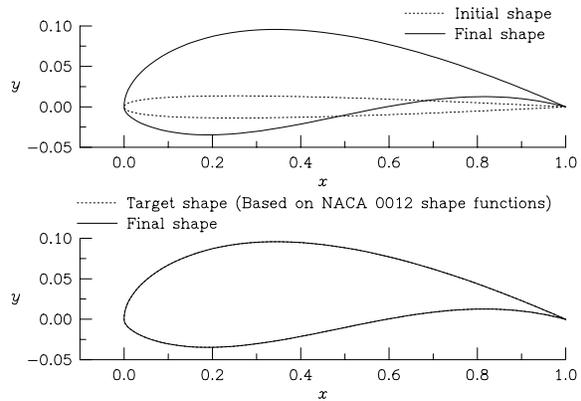


Figure 14. Test case 3.

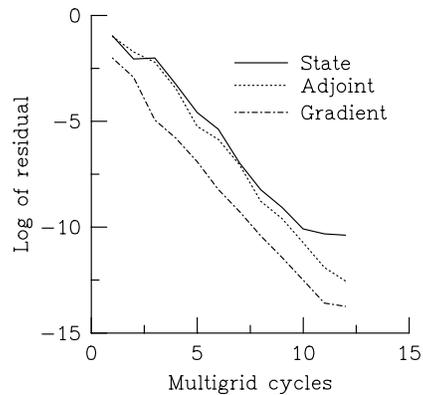


Figure 15. Convergence history.

## The Efficiency of One-Shot Method

Finally, the performance of the *One-Shot* method with respect to pure analysis is presented. The efficiency of a design method is defined as the ratio of the central processing unit (CPU) time that is required for the complete design process  $t_D$  to the CPU time that is required to do one analysis  $t_A$ . Figure 16 shows this ratio  $t_D/t_A$  plotted against the number of grid cells for the last test case. The figure shows that as the grid becomes finer the cost of design drops in comparison with the cost of one analysis. For the finest grid considered here, this ratio dropped below 4. The efficiencies for the other cases were similar.

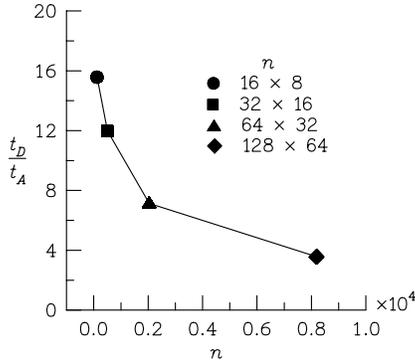


Figure 16. Efficiency of the *One-Shot* Method.

## 6. CONCLUDING REMARKS

An efficient method for the design of optimal airfoil shapes has been presented in this paper. This method brings the cost of the design process to the same order as that of the analysis problem. It offers great potential in designing optimal aircraft configurations efficiently at a reasonable computer cost. However, much work is still required before practical use can be made of this method.

## ACKNOWLEDGMENTS

The authors would like to thank Jonay Campbell, MHSI, for her help in editing the manuscript. The first and second authors were supported by the National Aeronautics and Space Administration (NASA) under contract nos. NAS1-19672 and NAS1-19480, respectively.

## REFERENCES

- 1 M. J. Lighthill. A New Method of Two Dimensional Aerodynamic Design. *ARC Rand M 2112*, 1945.
- 2 F. Bauer, P. Garabedian, and D. Korn. *A Theory of Supercritical Wing Sections with Computer Programs and Examples*. Springer-Verlag, 1972.
- 3 J. L. Steger and J. M. Klineberg. A Finite Difference Method for Transonic Airfoil Design. *AIAA J.*, Vol. 11, No. 5, May 1973, pp. 628-635.
- 4 L. A. Carlson, R. R. Ratcliff, T. A. Gally, and R. L. Campbell. Inverse Wing Design in Transonic Flow Including Viscous Interaction. In *Transonic Symposium: Theory, Application, and Experiment.*, NASA CP 3020, Vol. I, Part 2, 1989, pp. 497-520.
- 5 G. Volpe and R. E. Melnik. The Design of Transonic Aerofoils by a Well Posed Inverse Method. *Int. J.*

- 6 *Numerical Methods in Engineering*, Vol. 22, 1986, pp. 341-361.
- 7 M. Giles, M. Drela, and W. T. Thompkins. Newton Solution of Direct and Inverse Transonic Euler Equations. In *AIAA 7th Computational Fluid Dynamics Conference.*, 1985, pp. 394-402.
- 8 M. R. Hicks, M. E. Murman, and G. N. Vanderplaats. An Assessment of Airfoil Design by Numerical Optimization. *NASA TM X 3092*, July 1974.
- 9 G. A. Wrenn. An Indirect Method for Numerical Optimization Using the Kreisselmeier-Steinhauser Function. *NASA CR 4220*, March 1989.
- 10 G. B. Cosentino and T. L. Holst. Numerical Optimization Design of Advanced Transonic Wing Configurations. *NASA TM 85959*, May 1984.
- 11 K. D. Lee and S. Eyi. Transonic Airfoil Design by Constrained Optimization. *AIAA 9th Applied Aerodynamics Conference*, September, 1991, Baltimore, MD. AIAA Paper 91-3287.
- 12 A. C. Taylor, G. W. Hou, and V. M. Korivi. Sensitivity Analysis, Approximate Analysis, and Design Optimization for Internal and External Viscous Flows. *AIAA Aircraft Design Systems and Operations Meeting*, September, 1991, Baltimore, MD. AIAA Paper 91-3083.
- 13 A. Jameson. Automatic Design of Transonic Airfoils to Reduce the Shock Induced Pressure Drag. In *31st Israel Annual Conference on Aviation and Aeronautics*, February 21-22, 1990.
- 14 S. Ta'asan. *One Shot Methods for Optimal Control of Distributed Parameters Systems I: Finite Dimensional Control*. *ICASE Report No. 91-2*, January 1991.
- 15 Shlomo Ta'asan, G. Kuruvila, and M. D. Salas. Aerodynamic Design and Optimization in *One Shot*. *AIAA 30th Aerospace Sciences Meeting & Exhibit*, January, 1992, Reno, NV. AIAA Paper 92-0025.
- 16 K. Karamcheti. *Principles of Ideal-Fluid Aerodynamics*. Robert E. Krieger Publishing Company, 1980.
- 17 C. Hirsch. *Numerical Computation of Internal and External Flows. Vol. 1*. John Wiley & Sons, 1990.
- 18 A. Brandt. Multilevel Adaptive Solutions to Boundary-Value Problems. *Mathematics of Computation*, Vol. 31, April 1977, pp. 333-390.
- 19 R. Courant and D. Hilbert. *Methods of Mathematical Physics, Vol. I*. Interscience Publisher, Inc., N.Y., 7th Printing, 1966, pp. 48-54.
- 20 D. Althaus and F. X. Wortmann. *Stuttgarter Profilkatalog I*. Friedr. Vieweg & Sohn, Braunschweig, West Germany, 1981, pp. 96-97.

## Appendix A

### The Normal and the Tangent

Let the upper surface of the airfoil be

$$y = f(x) = 0 \quad (\text{A.1})$$

If the unit normal  $\mathbf{n}$  and the unit tangent  $\mathbf{t}$  are chosen such that the normal points into the flow field and  $\mathbf{t} \times \mathbf{n}$  points out of the paper, then

$$\begin{aligned} \mathbf{n} &= \frac{-f_x \mathbf{i} + \mathbf{j}}{\sqrt{1 + f_x^2}} \\ \mathbf{t} &= \frac{\mathbf{i} + f_x \mathbf{j}}{\sqrt{1 + f_x^2}} \end{aligned} \quad (\text{A.2})$$

where  $\mathbf{i}$  and  $\mathbf{j}$  are the unit vectors in the  $x$ - and  $y$ -directions, respectively, and  $f_x = df/dx$ . From (A.2) the following inverse relations can be written

$$\begin{aligned} \mathbf{i} &= (-f_x \mathbf{n} + \mathbf{t})/\sqrt{1+f_x^2} \\ \mathbf{j} &= (\mathbf{n} + f_x \mathbf{t})/\sqrt{1+f_x^2} \end{aligned} \quad (\text{A.3})$$

Let the upper surface be perturbed in the  $y$ -direction such that the new shape is

$$\bar{y} = f(x) + \varepsilon \tilde{f}(x) \quad (\text{A.4})$$

The new normal  $\bar{\mathbf{n}}$  and the new tangent  $\bar{\mathbf{t}}$  can be expressed as

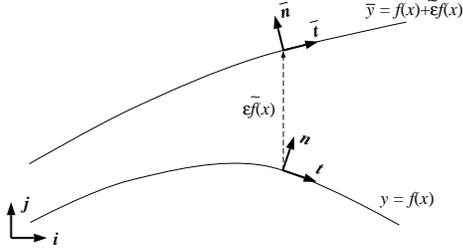


Figure 17. Perturbation of upper surface.

$$\begin{aligned} \bar{\mathbf{n}} &= \frac{-(f_x + \varepsilon \tilde{f}_x) \mathbf{i} + \mathbf{j}}{\sqrt{1 + (f_x + \varepsilon \tilde{f}_x)^2}} \\ \bar{\mathbf{t}} &= \frac{\mathbf{i} + (f_x + \varepsilon \tilde{f}_x) \mathbf{j}}{\sqrt{1 + (f_x + \varepsilon \tilde{f}_x)^2}} \end{aligned} \quad (\text{A.5})$$

With some algebraic manipulation we can show that

$$\begin{aligned} \bar{\mathbf{n}} &= \mathbf{n} - \varepsilon \frac{\tilde{f}_x}{\sqrt{1+f_x^2}} \mathbf{t} + O(\varepsilon^2) \\ \bar{\mathbf{t}} &= \mathbf{t} + \varepsilon \frac{\tilde{f}_x}{\sqrt{1+f_x^2}} \mathbf{n} + O(\varepsilon^2) \end{aligned} \quad (\text{A.6})$$

If the lower surface of the airfoil is given by

$$y - f(x) = 0 \quad (\text{A.7})$$

and the normal and the tangent are such that  $\mathbf{n}$  points into the flow and  $\mathbf{t} \times \mathbf{n}$  points out of the paper, then

$$\begin{aligned} \mathbf{n} &= \frac{f_x \mathbf{i} - \mathbf{j}}{\sqrt{1+f_x^2}} \\ \mathbf{t} &= \frac{-\mathbf{i} - f_x \mathbf{j}}{\sqrt{1+f_x^2}} \end{aligned} \quad (\text{A.8})$$

From (A.8), the inverse relation can be written as

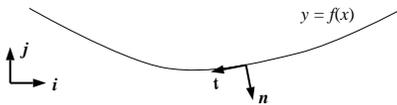


Figure 18. Normal and tangent on lower surface.

$$\begin{aligned} \mathbf{i} &= (f_x \mathbf{n} - \mathbf{t})/\sqrt{1+f_x^2} \\ \mathbf{j} &= (-\mathbf{n} - f_x \mathbf{t})/\sqrt{1+f_x^2} \end{aligned} \quad (\text{A.9})$$

The new normal and the tangent on the lower surface (perturbed in the  $y$ -direction) are also given by (A.6).

### The Infinitesimal Segment $d\sigma$

The infinitesimal segment  $d\sigma$  on the original airfoil can be written as

$$\begin{aligned} d\sigma^2 &= dx^2 + dy^2 \\ &= dx^2 + \left(\frac{df}{dx}\right)^2 dx^2 + h.o.t. \\ &= dx^2 (1 + f_x^2) + h.o.t. \end{aligned} \quad (\text{A.10})$$

The corresponding infinitesimal segment  $d\bar{\sigma}$  on the new airfoil can be expressed as

$$\begin{aligned} d\bar{\sigma}^2 &= dx^2 + d\bar{y}^2 \\ &= dx^2 \left[ 1 + (f_x + \varepsilon \tilde{f}_x)^2 \right] + h.o.t. \end{aligned} \quad (\text{A.11})$$

From (A.11) we can show that

$$d\bar{\sigma} = d\sigma \left( 1 + \varepsilon \frac{f_x \tilde{f}_x}{1 + f_x^2} \right) + O(\varepsilon^2) \quad (\text{A.12})$$

### A.1 The Gradient of the Cost Function

Let the change in the design variable be such that

$$\alpha \rightarrow \alpha + \varepsilon \tilde{\alpha} \quad (\text{A.13})$$

The resulting airfoil shape  $\bar{y}^{\bar{U}, \bar{L}}$  (Fig. 19) and the corresponding potential  $\bar{\phi}$  that satisfies the governing equation and its boundary conditions in the new domain  $\bar{\Omega}$  can be written as

$$\bar{y}^{\bar{U}, \bar{L}}(x) = y^{U, L}(x) + \varepsilon \tilde{y}^{U, L}(x) \quad (\text{A.14})$$

$$\bar{\phi} = \phi + \varepsilon \tilde{\phi} \quad (\text{A.15})$$

The cost function on the original airfoil is

$$F[\alpha, \phi(\alpha)] \equiv \int_{U+L} (\phi - \phi_0)^2 d\sigma \quad (\text{A.16})$$

On the new airfoil, the cost function can be written as

$$F(\alpha + \varepsilon \tilde{\alpha}, \bar{\phi}) = \int_{\bar{U} + \bar{L}} (\bar{\phi} - \phi_0)^2 d\bar{\sigma} \quad (\text{A.17})$$

If we use a Taylor series expansion, the potential on the new airfoil can be shifted to the original airfoil as follows:

$$\begin{aligned} \bar{\phi}^{\bar{U}, \bar{L}} &= \bar{\phi}^{U, L} + \varepsilon \tilde{y} \left( \frac{\partial \bar{\phi}}{\partial y} \right)^{U, L} + O(\varepsilon^2) \\ &= \phi^{U, L} + \varepsilon \tilde{\phi}^{U, L} + \varepsilon \tilde{y} (\nabla \phi \cdot \mathbf{j})^{U, L} + O(\varepsilon^2) \end{aligned} \quad (\text{A.18})$$

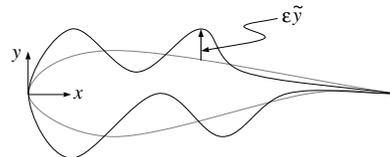


Figure 19. Perturbed airfoil shape.

If we substitute for  $d\bar{\sigma}$  from (A.12) and for  $\bar{\phi}$  from (A.18) and use (A.3) and (A.9), (A.17) can be written as

$$\begin{aligned}
F(\boldsymbol{\alpha} + \varepsilon \tilde{\boldsymbol{\alpha}}, \bar{\phi}) &= F(\boldsymbol{\alpha}, \phi) \\
&+ \varepsilon \int_{\bar{U}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1 + y_x^2}} \nabla \phi \cdot \mathbf{t} d\sigma \\
&- \varepsilon \int_{\bar{L}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1 + y_x^2}} \nabla \phi \cdot \mathbf{t} d\sigma \\
&+ \varepsilon \int_{\bar{U} + \bar{L}} (\phi - \phi_0)^2 \frac{y_x \tilde{y}_x}{1 + y_x^2} d\sigma \\
&+ \varepsilon \int_{\bar{U} + \bar{L}} 2(\phi - \phi_0) \tilde{\phi} d\sigma + O(\varepsilon^2)
\end{aligned} \tag{A.19}$$

The left-hand side of (A.19) can be expressed as

$$F(\boldsymbol{\alpha} + \varepsilon \tilde{\boldsymbol{\alpha}}, \bar{\phi}) = F(\boldsymbol{\alpha}, \phi) + \varepsilon \tilde{\boldsymbol{\alpha}}^T \nabla_{\boldsymbol{\alpha}} F + O(\varepsilon^2) \tag{A.20}$$

where

$$\nabla_{\boldsymbol{\alpha}} F \equiv \frac{\partial F}{\partial \boldsymbol{\alpha}} + \left( \frac{d\phi}{d\boldsymbol{\alpha}} \right)^T \frac{\partial F}{\partial \phi} \tag{A.21}$$

and  $\nabla_{\boldsymbol{\alpha}} F$  are the components of the gradient of the cost function. If we compare (A.19) and (A.20), we obtain

$$\begin{aligned}
\tilde{\boldsymbol{\alpha}}^T \nabla_{\boldsymbol{\alpha}} F &= \int_{\bar{U}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1 + y_x^2}} \nabla \phi \cdot \mathbf{t} d\sigma \\
&- \int_{\bar{L}} 2(\phi - \phi_0) \frac{y_x \tilde{y}}{\sqrt{1 + y_x^2}} \nabla \phi \cdot \mathbf{t} d\sigma \\
&+ \int_{\bar{U} + \bar{L}} (\phi - \phi_0)^2 \frac{y_x \tilde{y}_x}{1 + y_x^2} d\sigma \\
&+ \int_{\bar{U} + \bar{L}} 2(\phi - \phi_0) \tilde{\phi} d\sigma
\end{aligned} \tag{A.22}$$

## A.2 The State Equations

### The Governing Equation

For incompressible flow, the governing equation in the domain  $\Omega$  is

$$\operatorname{div}(\nabla \phi) = 0 \tag{A.23}$$

After the airfoil is perturbed, the governing equation in the new domain  $\bar{\Omega}$  is

$$\operatorname{div}(\nabla \bar{\phi}) = 0 \tag{A.24}$$

where

$$\bar{\phi} = \phi + \varepsilon \tilde{\phi} \tag{A.25}$$

In the region that is the intersection of both domains, we can write

$$\operatorname{div}(\nabla \bar{\phi}) - \operatorname{div}(\nabla \phi) = 0 \tag{A.26}$$

From (A.26) we can show that

$$\operatorname{div}(\nabla \tilde{\phi}) = 0 \quad \text{in } \Omega \cap \bar{\Omega} \tag{A.27}$$

Therefore, in the limit as  $\varepsilon \rightarrow 0$ , we can write

$$\operatorname{div}(\nabla \tilde{\phi}) = 0 \quad \text{in } \Omega \tag{A.28}$$

### The Airfoil Boundary Condition

The boundary condition on the airfoil is

$$\nabla \phi \cdot \mathbf{n} = 0 \tag{A.29}$$

where  $\mathbf{n}$  is the unit normal on the airfoil. If  $\bar{\mathbf{n}}$  is the unit normal on the new airfoil, then the boundary condition on the new airfoil can be written as

$$\nabla(\phi + \varepsilon \tilde{\phi}) \cdot \bar{\mathbf{n}} = 0 \tag{A.30}$$

With (A.6) and (A.18), the boundary condition on the new airfoil, shifted to the original airfoil, can be written as

$$\nabla(\phi + \varepsilon \tilde{\phi} + \varepsilon \tilde{y} \nabla \phi \cdot \mathbf{j}) \cdot \left( \mathbf{n} - \varepsilon \frac{\tilde{y}_x}{\sqrt{1 + y_x^2}} \mathbf{t} \right) + O(\varepsilon^2) = 0 \tag{A.31}$$

If we expand (A.31), substitute (A.29), and neglect the high-order terms, we can write

$$\nabla \tilde{\phi} \cdot \mathbf{n} = -\nabla(\tilde{y} \nabla \phi \cdot \mathbf{j}) \cdot \mathbf{n} + \frac{\tilde{y}_x}{\sqrt{1 + y_x^2}} \nabla \phi \cdot \mathbf{t} \tag{A.32}$$

Note that (A.32) is true on the original airfoil.

With (A.2), (A.3), (A.8), (A.9) and the boundary condition (A.29) we can show that

$$\begin{aligned}
\nabla \tilde{\phi} \cdot \mathbf{n} &= \nabla(\tilde{y} \nabla \phi \cdot \mathbf{t}) \cdot \mathbf{i} - \frac{\tilde{y}}{\sqrt{1 + y_x^2}} \nabla^2 \phi \quad \text{on } \bar{U} \\
\nabla \tilde{\phi} \cdot \mathbf{n} &= \nabla(\tilde{y} \nabla \phi \cdot \mathbf{t}) \cdot \mathbf{i} + \frac{\tilde{y}}{\sqrt{1 + y_x^2}} \nabla^2 \phi \quad \text{on } \bar{L}
\end{aligned} \tag{A.33}$$

Because  $\nabla^2 \phi = 0$  (the governing equation), we can write

$$\nabla \tilde{\phi} \cdot \mathbf{n} = \nabla(\tilde{y} \nabla \phi \cdot \mathbf{t}) \cdot \mathbf{i} \quad \text{on the airfoil} \tag{A.34}$$

### The Far-Field Boundary Condition

At the far field,

$$\nabla \phi \cdot \mathbf{n} = \mathbf{U}_{\infty} \cdot \mathbf{n} + \frac{\Gamma}{2\pi} \nabla \Theta \cdot \mathbf{n} \tag{A.35}$$

where  $\mathbf{n}$  is the unit normal on the far-field boundary. After the perturbation, we can write

$$\nabla(\phi + \varepsilon \tilde{\phi}) \cdot \mathbf{n} = \mathbf{U}_{\infty} \cdot \mathbf{n} + \frac{\Gamma + \varepsilon \tilde{\Gamma}}{2\pi} \nabla \Theta \cdot \mathbf{n} \tag{A.36}$$

If we subtract (A.35) from (A.36), we obtain

$$\nabla \tilde{\phi} \cdot \mathbf{n} = \frac{\tilde{\Gamma}}{2\pi} \nabla \Theta \cdot \mathbf{n} \tag{A.37}$$

### The Cut

Along the cut

$$\phi^T - \phi^B = \Gamma \tag{A.38}$$

where

$$\Gamma = \phi_{t,e}^T - \phi_{t,e}^B \tag{A.39}$$

After the perturbation,

$$(\phi + \varepsilon \tilde{\phi})^T - (\phi + \varepsilon \tilde{\phi})^B = \Gamma + \varepsilon \tilde{\Gamma} \tag{A.40}$$

From (A.38) through (A.40), we obtain

$$\tilde{\phi}^T - \tilde{\phi}^B = \tilde{\Gamma} \tag{A.41}$$

where

$$\tilde{\Gamma} = \tilde{\phi}_{t,e}^T - \tilde{\phi}_{t,e}^B \tag{A.42}$$



outermost cell in the  $\eta$ -direction. The velocity components  $\phi_x$  and  $\phi_y$  can be expressed as

$$\begin{aligned}\phi_x &= \phi_\xi \xi_x + \phi_\eta \eta_x \\ \phi_y &= \phi_\xi \xi_y + \phi_\eta \eta_y\end{aligned}\quad (\text{B.5})$$

If the inverse of the transformation  $x = x(\xi, \eta)$  and  $y = y(\xi, \eta)$  is known, the metric coefficients and the Jacobian can be expressed as

$$\begin{aligned}\xi_x &= J y_\eta \\ \xi_y &= -J x_\eta \\ \eta_x &= -J y_\xi \\ \eta_y &= J x_\xi\end{aligned}\quad (\text{B.6})$$

$$J = 1/(x_\xi y_\eta - x_\eta y_\xi) \quad (\text{B.7})$$

Figure 21 shows a typical cell in the flow field. The coor-

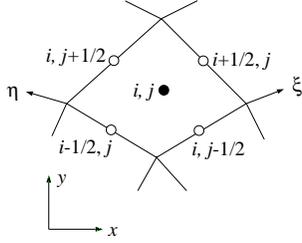


Figure 21. A cell in flow field.

ordinates of the vertices of the cell are known from the grid generation. That is,

$$\begin{aligned}x &= x\left(i \pm \frac{1}{2}, j \pm \frac{1}{2}\right) \\ y &= y\left(i \pm \frac{1}{2}, j \pm \frac{1}{2}\right)\end{aligned}\quad (\text{B.8})$$

By choosing  $\delta\xi = \delta\eta = 1$ , a finite-volume discretization of the governing equation (B.2) for the cell  $(i, j)$  can be written as

$$\left(\frac{\rho U}{J}\right)_{i+\frac{1}{2},j} - \left(\frac{\rho U}{J}\right)_{i-\frac{1}{2},j} + \left(\frac{\rho V}{J}\right)_{i,j+\frac{1}{2}} - \left(\frac{\rho V}{J}\right)_{i,j-\frac{1}{2}} = 0 \quad (\text{B.9})$$

Equation (B.9) is a consistent approximation to the integral form of the full potential equation.

Consider the first term in (B.9). If we substitute from (B.4) and (B.6), then we can write

$$\left(\frac{\rho U}{J}\right)_{i+\frac{1}{2},j} = \rho_{i+\frac{1}{2},j} (y_\eta \phi_x - x_\eta \phi_y)_{i+\frac{1}{2},j} \quad (\text{B.10})$$

where

$$(x_\eta)_{i+\frac{1}{2},j} = x_{i+\frac{1}{2},j+\frac{1}{2}} - x_{i+\frac{1}{2},j-\frac{1}{2}} \quad (\text{B.11})$$

The evaluation of  $y_\eta$  is similar. If we use (B.6), then the velocity components given by (B.5) can be written as

$$\begin{aligned}\phi_x &= J \bar{y}_\eta \phi_\xi - J \bar{y}_\xi \phi_\eta \\ \phi_y &= -J \bar{x}_\eta \phi_\xi + J \bar{x}_\xi \phi_\eta\end{aligned}\quad (\text{B.12})$$

The bars over the metric coefficients and the Jacobian indicate that they are evaluated with some mean values of  $x$  and  $y$ .

To ensure that the numerical discretization satisfies a uniform flow field identically, (B.12) is discretized as

$$\begin{aligned}(\phi_\xi)_{i+\frac{1}{2},j} &= \phi_{i+1,j} - \phi_{i,j} \\ (\phi_\eta)_{i+\frac{1}{2},j} &= \frac{1}{4}(\phi_{i+1,j+1} - \phi_{i+1,j-1} + \phi_{i,j+1} - \phi_{i,j-1})\end{aligned}\quad (\text{B.13})$$

and

$$\begin{aligned}(\bar{x}_\xi)_{i+\frac{1}{2},j} &= x_{i+1,j} - x_{i,j} \\ (\bar{x}_\eta)_{i+\frac{1}{2},j} &= \frac{1}{4}(x_{i+1,j+1} - x_{i+1,j-1} + x_{i,j+1} - x_{i,j-1})\end{aligned}\quad (\text{B.14})$$

where

$$x_{i,j} = \frac{1}{4}\left(x_{i-\frac{1}{2},j-\frac{1}{2}} + x_{i+\frac{1}{2},j-\frac{1}{2}} + x_{i+\frac{1}{2},j+\frac{1}{2}} + x_{i-\frac{1}{2},j+\frac{1}{2}}\right) \quad (\text{B.15})$$

Evaluation of  $\bar{y}_\xi$  and  $\bar{y}_\eta$  are similar.  $\bar{J}$  is evaluated as

$$\bar{J} = 1/(\bar{x}_\xi \bar{y}_\eta - \bar{x}_\eta \bar{y}_\xi) \quad (\text{B.16})$$

Similarly, we can write

$$\left(\frac{\rho V}{J}\right)_{i,j+\frac{1}{2}} = \rho_{i,j+\frac{1}{2}} (-y_\xi \phi_x + x_\xi \phi_y)_{i,j+\frac{1}{2}} \quad (\text{B.17})$$

where

$$(x_\xi)_{i,j+\frac{1}{2}} = x_{i+\frac{1}{2},j+\frac{1}{2}} - x_{i-\frac{1}{2},j+\frac{1}{2}} \quad (\text{B.18})$$

and  $y_\xi$  is evaluated similarly. The various pieces of the velocity components given by (B.12) are discretized as

$$\begin{aligned}(\phi_\xi)_{i,j+\frac{1}{2}} &= \frac{1}{4}(\phi_{i+1,j+1} - \phi_{i-1,j+1} + \phi_{i+1,j} - \phi_{i-1,j}) \\ (\phi_\eta)_{i,j+\frac{1}{2}} &= \phi_{i,j+1} - \phi_{i,j}\end{aligned}\quad (\text{B.19})$$

and

$$\begin{aligned}(\bar{x}_\xi)_{i,j+\frac{1}{2}} &= \frac{1}{4}(x_{i+1,j+1} - x_{i-1,j+1} + x_{i+1,j} - x_{i-1,j}) \\ (\bar{x}_\eta)_{i,j+\frac{1}{2}} &= x_{i,j+1} - x_{i,j}\end{aligned}\quad (\text{B.20})$$

where  $x_{i,j}$  is given by (B.15). Similarly,  $\bar{y}_\xi$  and  $\bar{y}_\eta$  are evaluated.

### B.3 The Boundary Conditions

The boundary conditions are imposed with one set of ghost cells around the computational domain. For the cells adjacent to the airfoil  $(i, 1)$ , the metric coefficients except  $(\bar{x}_\xi)_{i+1/2,1}$  are computed as

$$\begin{aligned}(\bar{x}_\eta)_{i+\frac{1}{2},1} &= x_{i+\frac{1}{2},\frac{3}{2}} - x_{i+\frac{1}{2},\frac{1}{2}} \\ (\bar{x}_\xi)_{i,\frac{1}{2}} &= x_{i+\frac{1}{2},\frac{1}{2}} - x_{i-\frac{1}{2},\frac{1}{2}} \\ (\bar{x}_\eta)_{i,\frac{1}{2}} &= \frac{1}{2}\left(x_{i+\frac{1}{2},\frac{3}{2}} - x_{i+\frac{1}{2},\frac{1}{2}} + x_{i-\frac{1}{2},\frac{3}{2}} - x_{i-\frac{1}{2},\frac{1}{2}}\right)\end{aligned}\quad (\text{B.21})$$

Similarly, the corresponding metric coefficients that are functions of  $y$  are also evaluated.

### The Airfoil

If we use (3.2), (B.4), and (B.6), then we can write that (on the airfoil)

$$(\nabla \phi \cdot \mathbf{n}) \sqrt{(x_\xi^2 + y_\xi^2)} = \frac{V}{J} = 0 \quad (\text{B.22})$$

Figure 22 shows a ghost cell adjacent to the airfoil. The

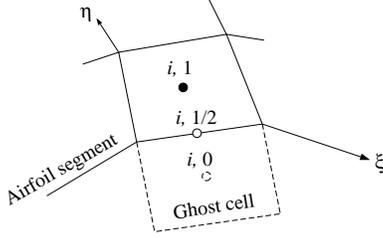


Figure 22. Ghost cell adjacent to airfoil.

value of  $\phi_{i,0}$  is set such that

$$\left(\frac{V}{J}\right)_{i, \frac{1}{2}} = 0 \quad (\text{B.23})$$

### The Far Field

Figure 23 shows a ghost cell adjacent to the far-field boundary. Similar to (B.22), we can show from (3.9) that at the far-field boundary

$$\frac{V}{J} = \left( \mathbf{U}_\infty \cdot \mathbf{n} + \frac{\Gamma}{2\pi} \nabla \Theta \cdot \mathbf{n} \right) \sqrt{(x_\xi^2 + y_\xi^2)} \quad (\text{B.24})$$

The value of  $\phi_{i,J+1}$  is set such that

$$\begin{aligned} & \left(\frac{V}{J}\right)_{i, J+\frac{1}{2}} \\ &= \left( \mathbf{U}_\infty \cdot \mathbf{n} + \frac{\Gamma}{2\pi} \nabla \Theta \cdot \mathbf{n} \right)_{i, J+\frac{1}{2}} \sqrt{(x_\xi^2 + y_\xi^2)_{i, J+\frac{1}{2}}} \end{aligned} \quad (\text{B.25})$$

The value of the circulation  $\Gamma$  is given by (refer to Appendix C for details)

$$\Gamma = \phi_{I,1} - \phi_{1,1} \quad (\text{B.26})$$

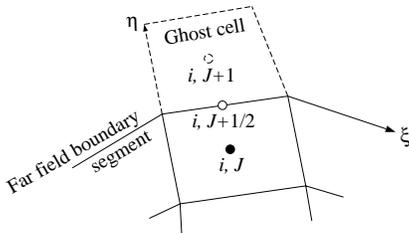


Figure 23. Ghost cell adjacent to far-field boundary.

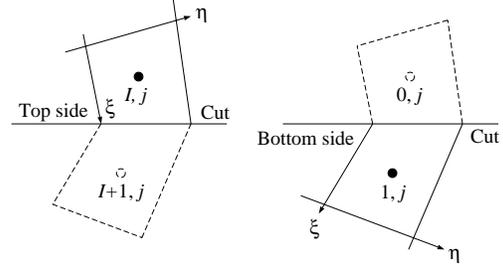
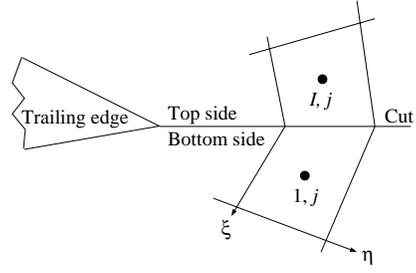


Figure 24. Ghost cells along cut.

### The Cut

Figure 24 shows the ghost cells along the top and bottom sides of the cut. The potential along the cut has a jump prescribed by (B.26). However, because the gradient of the potential normal to the cut must be continuous, the potential in these ghost cells is set as follows:

$$\begin{aligned} \phi_{0,j} &= \phi_{I,j} - \Gamma \\ \phi_{I+1,j} &= \phi_{1,j} + \Gamma \end{aligned} \quad (\text{B.27})$$

### B.4 The Solution Procedure

The discrete equations are solved with a Gauss-Seidel line-relaxation scheme. The nonlinearity introduced by the density  $\rho$  is handled by lagging its value by one iteration. Two systems of tridiagonal equations, one implicit in the  $\xi$ -direction and the other implicit in the  $\eta$ -direction, are solved sequentially with the Thomas algorithm. Note that the tridiagonal system implicit in the  $\xi$ -direction is periodic. A full approximation scheme (FAS) multigrid is used to accelerate the rate of convergence. Line relaxation is used to avoid the degradation in the performance of the multigrid scheme because of the presence of grid cells with large aspect ratios.

### B.5 The Multigrid Acceleration

A multigrid scheme is used to accelerate the convergence rate of the governing equations. In the multigrid process, starting with the fine grid, the problem is solved on a succession of increasingly coarser grids, and the corrections  $\delta\phi$  from the coarser grids are successively transferred back to the fine grid to obtain a new approximation to the solution. In this process, the component of the error that appears as a high frequency on a grid is damped very quickly by the iteration on that grid. Thus, low-frequency components of the error are damped on coarser grids, and the high-frequency components are damped on finer grids. This property of the multigrid is exploited by the *One-Shot* method during the design

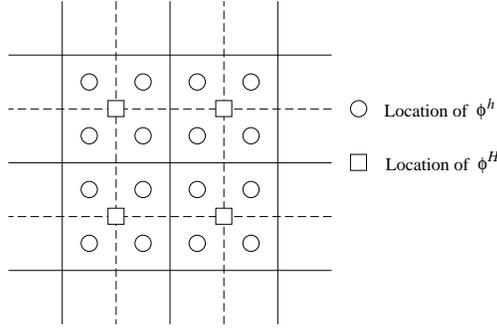


Figure 25. Location of  $\phi$  in fine and coarse grids.

process. Because the full potential equations are nonlinear, a FAS multigrid is used.

A two grid FAS multigrid algorithm is presented below. Let the fine grid on which solution is sought be represented by  $h$  and the coarse grid be represented by  $H$ . Also, let  $h$  and  $H$  represent grid sizes, where  $H > h$ . A coarse grid can be built by removing every second grid point from a fine grid. This makes  $H = 2h$ . For the cell-centered scheme, this method of coarsening combines four fine grid cells to form a coarse grid cell. Figure 25 shows the location of  $\phi^h$  and  $\phi^H$  on a fine and a coarse grid, respectively. Now, consider the following problem on grid  $h$ :

$$\mathcal{L}^h(\phi^h) = \mathcal{R}^h \quad (\text{B.28})$$

where  $\mathcal{L}^h(\phi^h)$  is a nonlinear equation and  $\mathcal{R}^h$  is its right-hand side. Equation (B.28) represents the discretized full potential equation or any of the boundary conditions. After a few relaxations of (B.28) on grid  $h$ , if we assume that the remaining error is smooth enough to be approximated on a coarse grid, then  $\phi$  and its residuals are transferred to the coarse grid  $H$  and an equivalent coarse grid problem is solved on this grid. The equivalent coarse grid problem can be written as

$$\mathcal{L}^H(\phi^H) = \mathcal{R}^H \quad (\text{B.29})$$

where

$$\mathcal{R}^H = \bar{I}_h^H [\mathcal{R}^h - \mathcal{L}^h(\phi^h)] + \mathcal{L}^H(\bar{I}_h^H \phi^h) \quad (\text{B.30})$$

and  $\bar{I}_h^H$  and  $I_h^H$  are the restriction operators that transfer  $\phi^h$  and its residuals to the coarse grid. Equation (B.29) is solved on the coarse grid, and the corrections  $\delta\phi^H$  are transferred back to the fine grid  $h$  to update  $\phi^h$  as follows:

$$\phi_{\text{New}}^h = \phi_{\text{Old}}^h + I_H^h \delta\phi^H \quad (\text{B.31})$$

where

$$\delta\phi^H = \phi^H - I_h^H \phi^h \quad (\text{B.32})$$

and  $I_H^h$  is the interpolation operator that transfers the corrections to the fine grid. This process is repeated until the residual of (B.28) reaches machine zero.

In the two-grid algorithm described above, we assume that the solution to (B.29) is accurate. In the multigrid algorithm, the solution on grid  $H$  is obtained by another two-grid iteration, where  $H$  is the fine grid and  $2H = 4h$  is the coarse grid. If this process is repeated this process on grid  $2H$  and so on, the “exact” solution is obtained on a very coarse grid. The sequence in which the transfer and relaxation are

performed between successive grids is done in various ways. Two of the more popular methods, V cycle and W cycle, are shown in Fig. 26. The V cycle consists of a sequence of relaxation and transfer to coarser grids with the “exact” resolution on the coarsest grid, followed by a sequence of relaxation and transfer back to the finest grid. In the Coarser levels are visited more often in the W cycle than in the V cycle. The W cycle, although 50 percent more expensive than V cycle, is more robust.

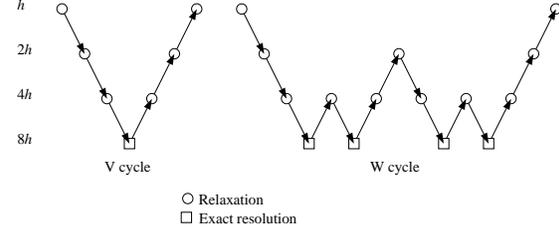


Figure 26. Multigrid cycling strategies.

### Appendix C

The Kutta condition states that the circulation  $\Gamma$  around the airfoil should be such that

*the velocity is finite and continuous at the trailing edge.*

The value of  $\Gamma$  is determined by requiring that the velocity that is perpendicular to the trailing edge bisector be equal to 0 at the trailing edge. In Fig. 27, let  $u_t$  and  $u_n$  be the

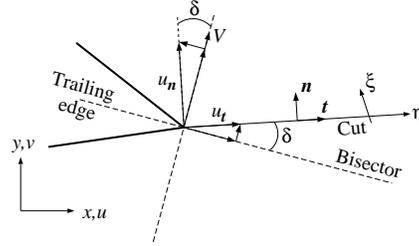


Figure 27. Velocity at trailing edge.

velocity components along the cut and perpendicular to the cut, respectively. Let the unit vectors in the corresponding directions be  $\mathbf{t}$  and  $\mathbf{n}$ , respectively. Let  $\delta$  be the angle between the trailing edge bisector and the cut. The velocity  $V$  perpendicular to the trailing-edge bisector, can be written as

$$V = u_n \cos \delta + u_t \sin \delta \quad (\text{C.1})$$

where

$$u_n = (u, v) \cdot \mathbf{n} \quad (\text{C.2})$$

$$u_t = (u, v) \cdot \mathbf{t}$$

As shown in Appendix B, the Cartesian velocity components  $u$  and  $v$  can be expressed as

$$\begin{aligned} u &= \phi_x = +J(y_\eta \phi_\xi - y_\xi \phi_\eta) \\ v &= \phi_y = -J(x_\eta \phi_\xi - x_\xi \phi_\eta) \end{aligned} \quad (\text{C.3})$$

and the unit vectors can be expressed as

$$\begin{aligned} \mathbf{n} &= (y_\eta/r, -x_\eta/r) \\ \mathbf{t} &= (x_\eta/r, y_\eta/r) \end{aligned} \quad (\text{C.4})$$

where

$$r = \sqrt{x_\eta^2 + y_\eta^2} \quad (\text{C.5})$$

If we substitute (C.2) through (C.5) into (C.1), it can be written as

$$V = [J(x_\eta^2 + y_\eta^2)\phi_\xi - J(x_\xi x_\eta + y_\xi y_\eta)\phi_\eta] \frac{\cos \delta}{r} + \phi_\eta \frac{\sin \delta}{r} \quad (\text{C.6})$$

To satisfy the Kutta condition, we require that  $V = 0$ . That is,

$$\phi_\xi + \left[ \frac{-J(x_\xi x_\eta + y_\xi y_\eta) + \tan \delta}{J(x_\eta^2 + y_\eta^2)} \right] \phi_\eta = 0 \quad (\text{C.7})$$

If the cut is aligned with the trailing edge bisector, then  $\delta = 0$ . If the grid is orthogonal, then  $x_\xi x_\eta + y_\xi y_\eta = 0$ . Therefore, if

the grid is orthogonal and the trailing-edge bisector is aligned with the cut, then (C.7) reduces to

$$\phi_\xi = 0 \quad (\text{C.8})$$

or

$$\phi_{t.e.}^T - (\phi_{t.e.}^B + \Gamma) = 0 \quad (\text{C.9})$$

where T and B refer to the top and bottom sides of the cut, respectively (see Fig. 1) and *t.e.* stands for the trailing edge. The value of  $\Gamma$  is easily obtained from (C.9).

In practice, particularly while designing an airfoil, the grid is not orthogonal nor is the trailing edge bisector aligned with the cut. However, numerical experiments have shown that the effect of the second term in (C.7) is of high order. Hence, a good approximation for the value of  $\Gamma$  can be obtained from (C.9).