Application of the Single-Cycle Optimization Approach to Aerodynamic Design

Magdi H. Rizk*
Flow Industries, Inc., Kent, Washington

A single-cycle approach has been developed to solve optimization problems in which the objective and constraint functions are dependent on the solution of a set of partial differential equations. This approach is very suitable for solving transonic aerodynamic design problems. The procedure simultaneously updates the solutions of the flow equations and the design parameters and, thus, presents an efficient alternative to the costly inner-outer iterative procedures currently used in transonic aerodynamic design. The procedure is applied here to examples in which the Euler equations are assumed to be the flow governing equations.

Nomenclature

c_1 = incrementing factor for optimization scheme, Eqs. (5b), (6b)
c_2 = decrementing factor for optimization scheme, Eqs. (5b), (6b)
c_3 = constant for the chord method, Eq. (7)
C_D = drag coefficient
C_L = lift coefficient
C_P = pressure coefficient
$\boldsymbol{e}_\ell$ = unit vector along the P_ℓ axis
E = objective function (or the performance index)
$\boldsymbol{f}$ = constraint vector
$\boldsymbol{g}$ = solution of flow governing equations
$\boldsymbol{i}_\ell$ = unit vector along the $\hat{P}_\ell$ axis
K = number of constraints
K_e = number of effective constraints
L = number of design parameters
M = Mach number
N = number of iterations required to achieve convergence
$\boldsymbol{P}$ = vector of design parameters (or the decision vector)
R = residual
T = transformation matrix
x = coordinate in the undisturbed flow direction normalized by the airfoil chord length
$\boldsymbol{y}_s$ = $[y_{ls}(x), y_{us}(x)]$
y_{ls}, y_{us} = vertical coordinate of airfoil lower and upper surfaces, respectively
$\boldsymbol{Y}^l$ = $[Y^l_{ls}(x), Y^l_{us}(x)]$
Y^l_{ls}, Y^l_{us} = vertical coordinate of the lower and upper surfaces of ℓ th shape function, respectively
α = angle of attack
ΔN = number of iterative steps after which the value of $\boldsymbol{P}$ is periodically updated
ΔP = small positive incremental value
$\delta \boldsymbol{P}$ = incremental vector used to update the vector of design parameters
δP_{max} = constant that sets an upper limit on the magnitude of the components of the incremental vector $\delta \boldsymbol{P}$
ϵ = small positive number for determining convergence
η = number of analysis iterative steps, which is equivalent to a design-analysis iterative step
μ = ν_e/η
ν_e = number of equivalent analyses
σ = defined in Eq. (4)
∇ = $(\partial/\partial P_1, \partial/\partial P_2, \ldots, \partial/\partial P_L)$

Subscripts

ℓ = element number in design parameter vector
∞ = undisturbed condition

Superscripts

A = analysis problem
D = design problem
n = iteration number
$(\,)'$ = transpose
$(\,)^*$ = optimum value
$(\,\hat{}\,)$ = rotated coordinate system

Introduction

NUMERICAL optimization is one of the tools being used in transonic aerodynamic design. The solution of the optimization problem attempts to determine the vector of design parameters $\boldsymbol{P}$ that minimizes the obective function $E(\boldsymbol{P};\boldsymbol{g})$ subject to the constraint $\boldsymbol{f}(\boldsymbol{P};\boldsymbol{g}) \leq 0$, where $\boldsymbol{g}$ is the solution of the flow equations. In airfoil and wing design problems, $\boldsymbol{P}$ contains the coefficients of the polynomials or the shape functions used to define the lifting surface, while E may be chosen to be the minimized drag, subject to the constraint of a minimum allowable lift value. The objective function E may also be chosen to be a measure of the difference between the pressure on the lifting surface and a desired pressure distribution.

Optimization methods (e.g., the steepest descent method, conjugate gradient method, and method of feasible directions) are iterative procedures that determine a sequence of solutions $\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots$, that converges to the optimum solution $\boldsymbol{P}^*$. These methods require the evaluation of the objective function many times before the optimum solution is determined. Since E is dependent on the solution $\boldsymbol{g}$, the flow governing equations must be solved each time E is evaluated. Therefore, any of the usual optimization schemes (e.g., the method of feasible directions) becomes a two-cycle (inner-outer) iterative procedure. The inner iterative cycle solves the analysis problem for $\boldsymbol{g}$ iteratively for a given iterative solution $\boldsymbol{P}$, while the outer iterative cycle determines the optimum $\boldsymbol{P}$ iteratively. In the inner-outer iterative approach[1-3] for solving the design problem, the usual procedure is to couple an existing analysis

Presented as Paper 84-2165 at the AIAA 2nd Applied Aerodynamics Conference, Seattle, Wash., Aug. 21-23, 1984; received Oct. 24, 1984; revision received Feb. 12, 1985.

*Senior Research Scientist, Research and Technology Division. Member AIAA.

code (which solves the flow equation iteratively for a given $\boldsymbol{P}$) to an optimization code (which finds the optimum $\boldsymbol{P}$ iteratively). The repetitive execution of time-consuming analysis codes is the source of the high cost of this approach. A reduction in the number of times the analysis problem must be solved before the optimum $\boldsymbol{P}$ is determined has been achieved recently by utilizing fast approximation methods.[4,5]

The purpose of this paper is to introduce the design engineer to a different optimization approach that is suitable for aerodynamic design. This single-cycle approach has recently been developed for solving optimization problems in which the objective and constraint functions are dependent on the solution of a set of partial differential equations. The scheme is obtained by modifying the iterative procedure for solving the partial differential equations (the analysis problem), so that the solutions of the differential equations $\boldsymbol{g}$ and the design parameters $\boldsymbol{P}$ are updated simultaneously. This results in the successively improved approximations $(\boldsymbol{g}^n,\boldsymbol{P}^n)$, where $n=1,2,\ldots$, that converge to the solution $(\boldsymbol{g}^*,\boldsymbol{P}^*)$, thus satisfying the optimization problem. With this approach, the need for the costly inner-outer iterative procedure is eliminated. A scheme using this approach was developed in Ref. 6 for application to unconstrained optimization problems. This scheme was extended in Ref. 7 so that it could be used to solve general constrained problems. Unlike presently used approaches in which constrained algorithms are much more inefficient than unconstrained ones,[3] the effect of the constraints is to increase efficiency in the new approach.[7] This approach has been applied to examples in which the potential equation is assumed to be governing equation.[6,7] In this paper, it is applied to examples in which the Euler equations are assumed to be the governing equations.

Approach

A concise presentation of the single-cycle approach is given here. Further details are available in Refs. 6 and 7.

This approach modifies the iterative procedure of determining the solution $\boldsymbol{g}$ for the analysis problem, in which $\boldsymbol{P}$ is given, to a procedure that determines the solutions $\boldsymbol{g}$ and $\boldsymbol{P}$ for the optimization (design) problem. The iterative solutions for the analysis problem are given by

$$\boldsymbol{g}^{n+1}=\psi(\boldsymbol{g}^n;\boldsymbol{P}), \qquad n=0,1,2,\ldots \tag{1}$$

where $\psi(\boldsymbol{g}^n;\boldsymbol{P})$ denotes the solution obtained by applying the iterative scheme once, using $\boldsymbol{g}^n$ as the initial guess, while holding the value of $\boldsymbol{P}$ fixed. The iterative solutions for the design problems are given by

$$\boldsymbol{P}^{n+1}=\boldsymbol{P}^n+\delta\boldsymbol{P}^{n+1}, \qquad n=0,1,2,\ldots \tag{2}$$

$$\boldsymbol{g}^{n+1}=\psi(\boldsymbol{g}^n;\boldsymbol{P}^{n+1}), \qquad n=0,1,2,\ldots \tag{3}$$

where $\delta\boldsymbol{P}^{n+1}$ is set equal to zero for a number $(\Delta N-1)$ of consecutive iterative steps $[(n+1)/\Delta N\neq 1,2,3,\ldots]$ and is then determined through a special scheme in the following step $[(n+1)/\Delta N=1,2,3,\ldots]$. Therefore, two types of iterative steps are used to solve the design problem. The analysis iterative step updates the solution $\boldsymbol{g}$. It is equivalent to the iterative step used in the analysis scheme. With every ΔN iterations, the analysis iterative step is preceded by the design iterative step updating $\boldsymbol{P}$. The design iterative step is equivalent to $L+1$ analysis iterative steps. Therefore, on average, an iterative step of the single-cycle scheme used here is equivalent to σ analysis iterative steps, where

$$\sigma=(\Delta N+L+1)/\Delta N \tag{4}$$

The optimization iterative process continues until the convergence criterion

$$\max(|R^{n+1}|-\epsilon_e,\max_{\ell}|\delta P_\ell^{n+1}|-\epsilon_p)<0$$

is satisfied.

There are two options when applying the design iterative step. The unconstrained option is used when solving an unconstrained optimization problem. It is also used when solving a constrained optimization problem as long as the vector of design parameters $\boldsymbol{P}^n$ falls within the admissible region (i.e., no constraints are violated). However, if a constraint is effective, it becomes necessary to search for the optimum design parameter within the constraint boundary. In this case, the constrained option is used.

The Unconstrained Option

When this option is used in the $n+1$ iterative step, $\boldsymbol{P}$ is updated by Eq. (2). The sign of the incremental correction δP_ℓ^{n+1} (where δP_ℓ^{n+1} is the ℓ th component of the vector $\delta\boldsymbol{P}^{n+1}$) is chosen to be opposite to that of $\partial E/\partial P_\ell^n$ in order to reduce E further. The magnitude of the increment δP_ℓ^{n+1} is given by

$$|\delta P_\ell^{n+1}|=c\,|\delta P_\ell^{n+1-\Delta N}|$$

where $c>0$. If the signs of δP_ℓ^{n+1} and $\delta P_\ell^{n+1-\Delta N}$ are in agreement, it is an indication that the last two iterative solutions P_ℓ^n and $P_\ell^{n-\Delta N}$ fall to one side of the point along the P_ℓ direction at which E is a minimum. In such a case, c is set equal to the constant c_1, which is greater than 1. Increasing the magnitude of the step size in this manner will accelerate the approach toward the point along the P_ℓ direction at which E is a minimum. On the other hand, if the signs of δP_ℓ^{n+1} and $\delta P_\ell^{n+1-\Delta N}$ are not in agreement, it is an indication that P_ℓ^n and $P_\ell^{n-\Delta N}$ fall on opposite sides of the point along the P_ℓ direction at which E is a minimum. In this case, c is set equal to the constant c_2, which is less than 1. Decreasing the magnitude of the step size in this manner is necessary for convergence to the point along the P_ℓ direction at which E is a minimum.

The equation used to calculate δP_ℓ^{n+1} in the case of the unconstrained option is given by

$$\delta P_\ell^{n+1}=\min\left(1,\frac{\delta P_{\max}}{|\bar{\delta}P_\ell^{n+1}|}\right)\bar{\delta}P_\ell^{n+1}, \qquad \ell=1,2,\ldots,L \tag{5a}$$

where

$$\bar{\delta}P_\ell^{n+1}=\tfrac{1}{2}[c_1(\tau_\ell^{n+1}+1)+c_2(\tau_\ell^{n+1}-1)]\,\delta P_\ell^{n+1-\Delta N}$$

$$\tau_\ell^{n+1}=-\frac{\Gamma_\ell\delta P_\ell^{n+1-\Delta N}}{|\Gamma_\ell\delta P_\ell^{n+1-\Delta N}|}, \qquad \Gamma_\ell=\frac{\partial E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})}{\partial P_\ell}$$

$$\boldsymbol{G}^{0,n+1}=\psi(\boldsymbol{g}^n;\boldsymbol{P}^n), \qquad \boldsymbol{G}^{\ell,n+1}=\psi(\boldsymbol{g}^n;\boldsymbol{P}^n+\Delta P\boldsymbol{e}_\ell) \tag{5b}$$

and Γ_ℓ is evaluated by

$$\Gamma_\ell=\frac{E(\boldsymbol{P}^n+\Delta P\boldsymbol{e}_\ell;\boldsymbol{G}^{\ell,n+1})-E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})}{\Delta P}$$

Here Γ_ℓ is the ℓ th component of $\nabla E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})$.

The Constrained Option

At iterative step $n+1$, assumed here to be a design analysis iterative step $[(n+1)/\Delta N=1,2,3,\ldots]$, let the first K_e^n inequality constraints

$$f_k(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})\leq 0, \qquad k=1,2,\ldots,K_e^n$$

be effective where $K_e^n\leq K$. The kth constraint is effective if either of the following conditions is satisfied:

1) $f_k(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})\geq 0$.

2) The kth constraint was effective in the previous design iterative step and the following conditions are satisfied:

$$f_k(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})<0, \qquad \frac{\partial E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})}{\partial\beta_k^{n+1}}<0$$

where β_k^{n+1} is a coordinate in the $\nabla f_k(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})$ direction.

The search for the optimum solution must be restricted to the subspace that is the intersection of the constraint surfaces $f_k(\boldsymbol{P};\boldsymbol{G}^{0,n+1})=0$, $1\le k\le K_e^n$.

It is convenient to decompose the incremental correction $\delta\boldsymbol{P}^{n+1}$ into two vectors $\delta\boldsymbol{N}^{n+1}$ and $\delta\boldsymbol{S}^{n+1}$ that are orthogonal to each other. The hyperplane tangent to the hypersurface $f_k(\boldsymbol{P};\boldsymbol{G}^{0,n+1})=\text{const}$, $1\le k\le K_e^n$, at $\boldsymbol{P}=\boldsymbol{P}^n$ is an $(L\text{-}1)$-dimensional linear subspace. Let $C^{n+1}(1)$ define the first $(k=1)$ of these hyperplanes and let $C^{n+1}(k)$, $2\le k\le K_e^n$ define the $(L\text{-}k)$- dimensional linear subspace, which is the intersection of the first k hyperplanes. The vector $\delta\boldsymbol{N}^{n+1}$ is chosen to be orthogonal to $C^{n+1}(K_e^n)$. It is responsible for constraint corrections. The vector $\delta\boldsymbol{S}^{n+1}$ lies within $C^{n+1}(K_e^n)$. The purpose of this correction is to bring the updated design parameter vector $\boldsymbol{P}^{n+1}$ closer to the optimum design point within $C^{n+1}(K_e^n)$. The vector $\delta\boldsymbol{N}^{n+1}$ is most conveniently determined in terms of an orthonormal basis whose members are orthogonal to $C^{n+1}(K_e^n)$, while $\delta\boldsymbol{S}^{n+1}$ is most conveniently determined in terms of an orthonormal basis that spans $C^{n+1}(K_e^n)$.

The first orthonormal set $\boldsymbol{n}_k$, $1\le k\le K_e^n$ is defined such that $\boldsymbol{n}_1$ is orthogonal to the hypersurface $f_1(\boldsymbol{P};\boldsymbol{G}^{0,n+1})=\text{const}$ at $\boldsymbol{P}=\boldsymbol{P}^n$, while $\boldsymbol{n}_k$, $2\le k\le K_e^n$, is orthogonal to $C^{n+1}(k)$; i.e., it is orthogonal to the intersection of the hypersurface $f_k(\boldsymbol{P};\boldsymbol{G}^{0,n+1})=\text{const}$ and the subspace $C^{n+1}(k-1)$ at $\boldsymbol{P}=\boldsymbol{P}^n$. These vectors are given by

$$\boldsymbol{n}_1=\boldsymbol{N}_1/|\boldsymbol{N}_1|$$

$$\boldsymbol{n}_k=\boldsymbol{Q}_k/|\boldsymbol{Q}_k|,\qquad k=2,3,\ldots,K_e^n$$

$$\boldsymbol{Q}_k=\boldsymbol{N}_k-\sum_{r=1}^{k-1}(\boldsymbol{N}_k\cdot\boldsymbol{n}_r)\boldsymbol{n}_r,\qquad k=2,3,\ldots,K_e^n$$

$$\boldsymbol{N}_k=\nabla f_k(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})$$

where $\boldsymbol{N}_k$ is orthogonal to the hypersurface $f_k(\boldsymbol{P};\boldsymbol{G}^{0,n+1})=\text{const}$ at $\boldsymbol{P}=\boldsymbol{P}^n$, and $\boldsymbol{Q}_k$ is the projection of $\boldsymbol{N}_k$ on the subspace $C^{n+1}(k-1)$. The second set of orthogonal unit vectors $\boldsymbol{i}_k$, $K_e^n+1\le k\le L$, is constructed by taking linear combinations of the vectors $\boldsymbol{n}_k$, $1\le k\le K_e^n$ and $\boldsymbol{e}_k$, $K_e^n+1\le k\le L$. It is assumed that this set of L vectors is linearly independent. If this is not the case, then the vectors $\boldsymbol{e}_k$, $K_e^n+1\le k\le L$ are replaced by any $L-K_e^n$ vectors from the set $\boldsymbol{e}_k$, $1\le k\le L$, such that the set composed of these vectors and the vectors $\boldsymbol{n}_k$, $1\le k\le K_e^n$ is linearly independent. The following equations are used to construct the set $(\boldsymbol{i}_1^{n+1},\boldsymbol{i}_2^{n+1},\ldots,\boldsymbol{i}_L^{n+1})$:

$$\boldsymbol{i}_\ell=\boldsymbol{n}_\ell,\qquad \ell=1,2,\ldots,K_e^n$$

$$=\boldsymbol{I}_\ell/|\boldsymbol{I}_\ell|,\qquad \ell=K_e^n+1,\ldots,L$$

where

$$\boldsymbol{I}_\ell=\boldsymbol{e}_\ell-\sum_{k=1}^{\ell-1}(\boldsymbol{e}_\ell\cdot\boldsymbol{i}_k^{n+1})\boldsymbol{i}_k^{n+1}$$

Along these vectors a new set of coordinate axes $\hat{P}_1^{n+1}$, $\hat{P}_2^{n+1},\ldots,\hat{P}_L^{n+1}$ is obtained by a rotation of the original coordinate axes $P_1, P_2,\ldots, P_L$.

While the components of a vector $\boldsymbol{u}$ relative to the original axes are given by

$$u_\ell=\boldsymbol{u}\cdot\boldsymbol{e}_\ell,\qquad \ell=1,2,\ldots,L$$

the components of $\boldsymbol{u}$ relative to the rotated axes are given by

$$\hat{u}_\ell^{n+1}=\boldsymbol{u}\cdot\boldsymbol{i}_\ell^{n+1},\qquad \ell=1,2,\ldots,L$$

The vectors $\boldsymbol{u}$, $\boldsymbol{u}=[u_1u_2\ldots u_L]'$ and $\hat{\boldsymbol{u}}^{n+1}$, $\hat{\boldsymbol{u}}^{n+1}=[\hat{u}_1^{n+1}\hat{u}_2^{n+1}\ldots\hat{u}_L^{n+1}]'$, are related by the transformation equation

$$\boldsymbol{u}=\boldsymbol{T}^{n+1}\hat{\boldsymbol{u}}^{n+1}$$

or

$$\hat{\boldsymbol{u}}^{n+1}=(\boldsymbol{T}^{n+1})'\boldsymbol{u}$$

where the orthogonal transformation matrix $\boldsymbol{T}^{n+1}$ is given by

$$\boldsymbol{T}^{n+1}=[\boldsymbol{i}_1^{n+1}\boldsymbol{i}_2^{n+1}\ldots\boldsymbol{i}_L^{n+1}]$$

When the constrained option is used in the $n+1$ iterative step, $\boldsymbol{P}$ is updated by the equation

$$\boldsymbol{P}^{n+1}=\boldsymbol{P}^n+\delta\boldsymbol{P}^{n+1}$$

where

$$\delta\boldsymbol{P}^{n+1}=\boldsymbol{T}^{n+1}\delta\hat{\boldsymbol{P}}^{n+1}$$

$$\delta\hat{\boldsymbol{P}}^{n+1}=\delta\hat{\boldsymbol{S}}^{n+1}+\delta\hat{\boldsymbol{N}}^{n+1}$$

$$\delta\hat{\boldsymbol{S}}^{n+1}=\begin{bmatrix}0\\0\\\cdot\\\cdot\\\cdot\\0\\\delta\hat{S}_{\kappa+1}^{n+1}\\\cdot\\\cdot\\\cdot\\\delta\hat{S}_L^{n+1}\end{bmatrix}\qquad \delta\hat{\boldsymbol{N}}^{n+1}=\begin{bmatrix}\delta\hat{N}_1^{n+1}\\0\\\cdot\\\cdot\\\cdot\\\delta\hat{N}_\kappa^{n+1}\\0\\\cdot\\\cdot\\\cdot\\0\end{bmatrix}\qquad \kappa=K_e^n$$

The components $\delta\hat{S}_\ell^{n+1}$, $K_e^n+1\le\ell\le L$ are calculated using the unconstrained option. Therefore,

$$\delta\hat{S}_\ell^{n+1}=\min\left(1,\frac{\delta P_{\max}}{|\delta\bar{\hat{S}}_\ell^{n+1}|}\right)\delta\bar{\hat{S}}_\ell^{n+1},\qquad \ell=K_e^n+1,\ldots,L \tag{6a}$$

where

$$\delta\bar{\hat{S}}_\ell^{n+1}=\tfrac{1}{2}[c_1(t_\ell^{n+1}+1)+c_2(t_\ell^{n+1}-1)](\boldsymbol{i}_\ell^{n+1}\cdot\delta\boldsymbol{P}^{n+1-\Delta N})$$

$$\ell=K_e^n+1,\ldots,L$$

$$t_\ell^{n+1}=-\frac{\hat{\Gamma}_\ell(\boldsymbol{i}_\ell^{n+1}\cdot\delta\boldsymbol{P}^{n+1-\Delta N})}{|\hat{\Gamma}_\ell(\boldsymbol{i}_\ell^{n+1}\cdot\delta\boldsymbol{P}^{n+1-\Delta N})|}$$

$$\hat{\Gamma}_\ell=\frac{\partial E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1})}{\partial\hat{P}_\ell^{n+1}}=\boldsymbol{i}_\ell^{n+1}\cdot\nabla E(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1}) \tag{6b}$$

The components $\delta\hat{N}_\ell^{n+1}$, $1\le\ell\le K_e^n$ are calculated by using the chord method. Therefore,

$$\delta\hat{N}_\ell^{n+1}=-c_3f_\ell(\boldsymbol{P}^n;\boldsymbol{G}^{0,n+1}),\qquad \ell=1,2,\ldots,K_e^n \tag{7}$$

However, other iterative procedures may be used. In Ref. 7, the constraint was satisfied by using Newton's method instead of the chord method. Another possible alternative to Eq. (7) for determining $\delta\hat{N}_\ell^{n+1}$ is to use the minimization scheme of Eqs. (6a) and (6b) by replacing $\delta\hat{S}_\ell^{n+1}$ and E in these equations with $\delta\hat{N}_\ell^{n+1}$ and $|f_\ell|$, respectively.

Design Examples and Discussion

The single-cycle approach is applied here to airfoil design examples. In the examples, $\boldsymbol{P}$ contains the coefficients of the

shape functions used to define the airfoil. The upper and lower airfoil surfaces are defined by

$$y_s(x) = \sum_{\ell=1}^{L} P_\ell Y^\ell(x)$$

where Y defines the shape functions, which are profiles of given airfoils. The shape functions used in the examples are depicted in Fig. 1. The first shape function is the BGK1 airfoil. The second shape function is a Karman-Trefftz airfoil given by

$$z = x_p + iy_p = mc\frac{1+H^m}{1-H^m}$$

where

$$i=\sqrt{-1}, \quad H=(\xi-c)/(\xi+c)$$

$$\xi=\xi_0+re^{i\theta} \quad (0\le\theta\le 2\pi), \quad r=\sqrt{\eta_0^2+(c-\xi_0)^2}$$

$$c=1.0, \quad m=1.90, \quad \xi_0=-0.08, \quad \eta_0=0.08$$

and (x_p,y_p) are the coordinates of points on the airfoil. The third shape function is a NACA 0012 airfoil.

The flow about the airfoil is assumed to be governed by the Euler equations. Reference 8 solves the Euler equations by using the Runge-Kutta time-stepping scheme. The improvements to the Euler code developed by Jameson et al.[8] have been introduced by Salas et al.[9] This improved code has been modified to allow the use of the single-cycle scheme in airfoil design problems.

In the following examples, the designed airfoil is assumed to be at an angle of attack $\alpha=2.5$ deg and the freestream Mach number is assumed to be $M_\infty=0.7$. The initial iterative guess for the solution $\boldsymbol{g}$ is given by the freestream conditions. Therefore,

$$\boldsymbol{g}^0=\boldsymbol{g}_\infty$$

The initial incremental values δP_ℓ are given by

$$\delta P_\ell^0=0.03, \qquad \ell=1,2,\ldots,L$$

Unless otherwise stated, the following values are used for the computational parameters c_1, c_2, c_3, ΔP, $\delta P_{\max}$, and ΔN:

$$c_1=1.02 \qquad c_2=0.6$$

$$c_3=3.0 \qquad \Delta N=10$$

$$\Delta P=0.001 \qquad \delta P_{\max}=0.03$$

In the first example, an airfoil is designed to maximize the lift subject to a constraint defining a maximum allowable

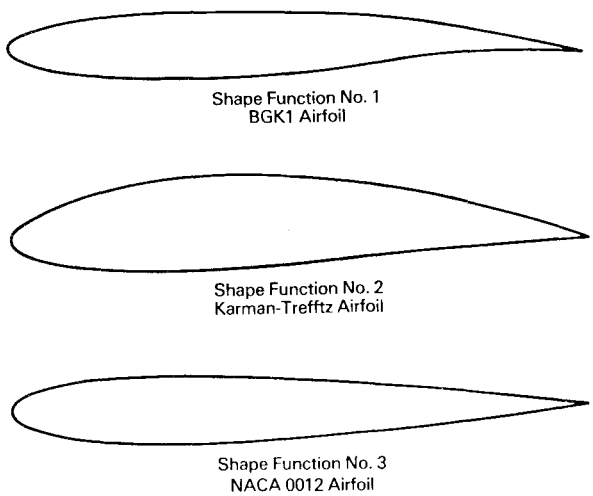

Fig. 1 Shape functions.

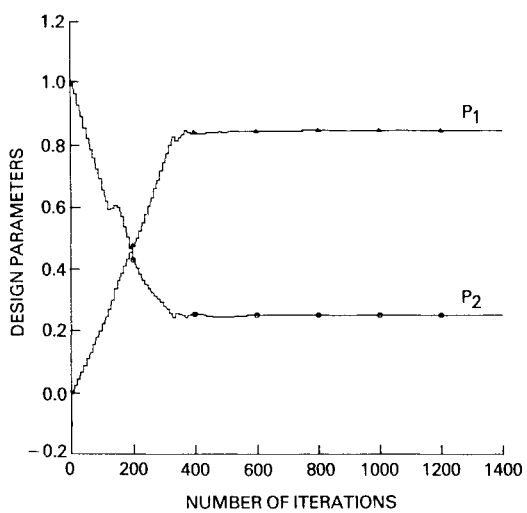

Fig. 2 Design parameters evolution histories (example 1).

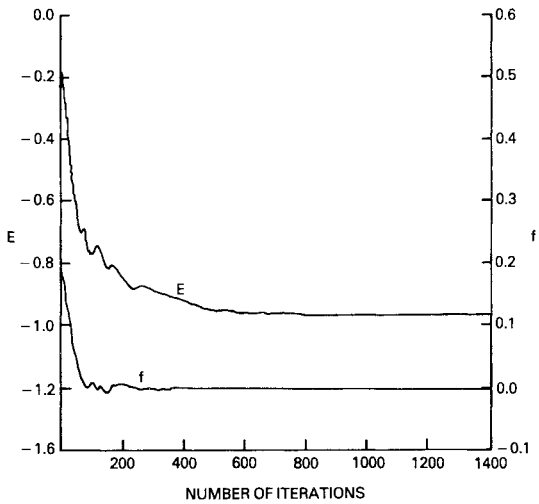

Fig. 3 Objective function and constraint function evolution histories (example 1).

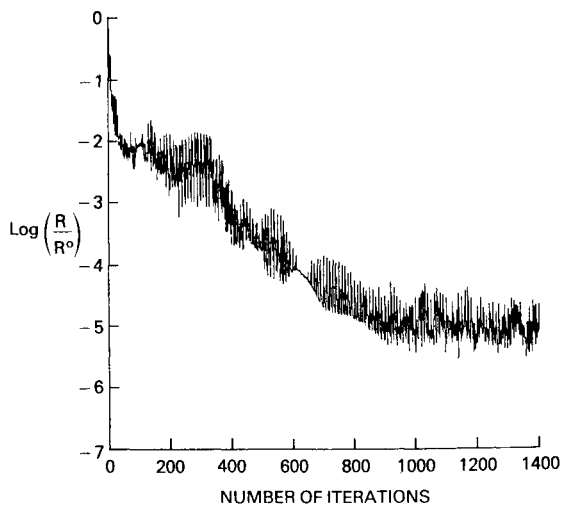

Fig. 4 Residual evolution history for the design problem (example 1).

drag. The objective function E, which is to be minimized, is therefore defined by $E=-C_L$, while the constraint function is given by

$$f=C_D-C_{DO}\leq 0$$

where $C_{DO}=0.02$.

A second constraint, which puts limits on the allowable airfoil area, is given by

$$\sum_{\ell=1}^{3} P_\ell = 1$$

Using this relation, it is possible to express one of the design parameters in terms of the other two. Therefore, the problem is redefined to be a two-design-parameter problem that seeks to determine P_1 and P_2 so that the airfoil defined by

$$y_s(x)=P_1[Y^1(x)-Y^3(x)]+P_2[Y^2(x)-Y^3(x)]+Y^3(x)$$

satisfies the design conditions. The initial values of the design parameters are given by

$$\boldsymbol{P}^0=\begin{bmatrix}0\\1\end{bmatrix}$$

The evolution histories of the design parameters are shown in Fig. 2. The converged solution is given by $P_1=0.846$, $P_2=0.251$, $P_3=-0.097$. The evolution histories of the objective and constraint functions are shown in Fig. 3. From Figs. 2 and 3, two distinct stages in the convergence process of the design parameters may be identified. In the first stage, relatively rapid changes in the values of $\boldsymbol{P}^n$, E^n, and f^n occur as they approach the converged values of the solutions. At the end of this stage, these parameters are close to their final values. In the second stage, minor adjustments take place as the parameter values converge to their final values. In this example, the first stage spans approximately the first 400 iterations.

The maximum residual R^n, which is a measure of the convergence of the flowfield solution $\boldsymbol{g}^n$, is shown in Fig. 4. The high-frequency oscillations apparent in the figure are of period ΔN iterations and are due to introducing a perturbation in the boundary conditions as $\boldsymbol{P}^n$ is updated. The corresponding R^n curve for the analysis problem, with $\boldsymbol{P}=\boldsymbol{P}^*$, is shown in Fig. 5. As expected, a comparison of the two figures indicates a faster rate of convergence for the analysis problem, which is allowed to converge without introducing the boundary condition perturbations. The high-frequency oscillations appearing in Fig. 5 after approximately 700 iterations are due to rounding errors. In order to avoid this error, the convergence criterion for the residual R^n was set at $\epsilon_e=0.5\times10^{-3}$ and the convergence criterion for the design parameters at $\epsilon_p=10^{-5}$. On this basis, it is found that the solution of the

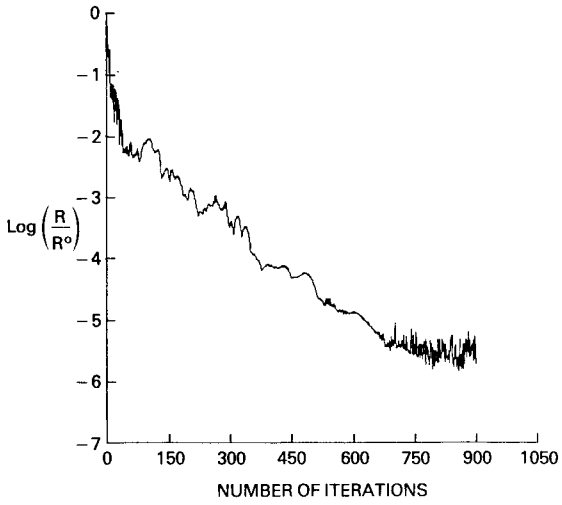

Fig. 5 Residual evolution history for the analysis problem (example 1).

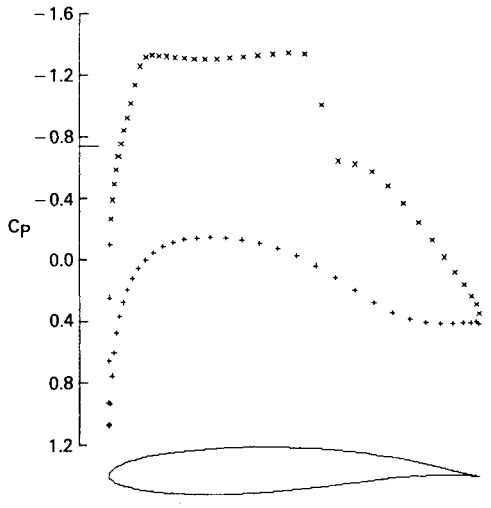

Fig. 6 Pressure distribution on surface of designed airfoil.

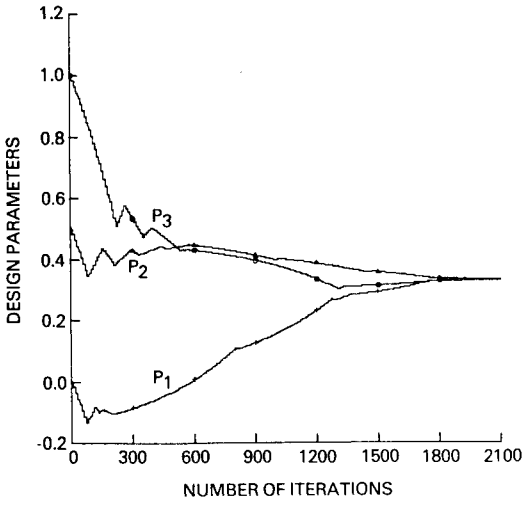

Fig. 7 Design parameters evolution histories ($c_l=1.02$).

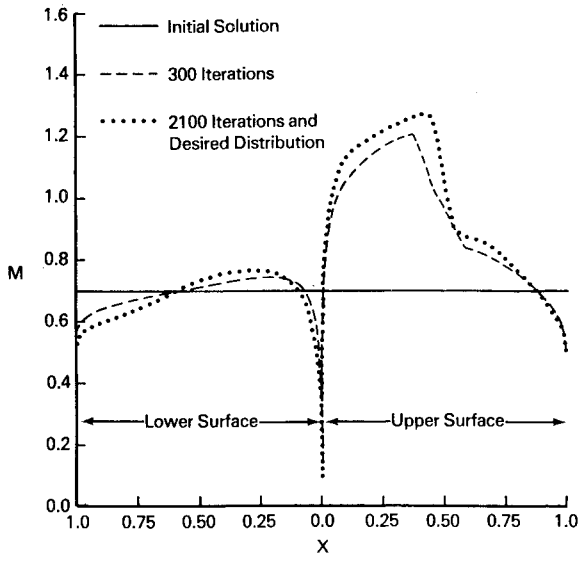

Fig. 8 Mach number distribution along airfoil surface.

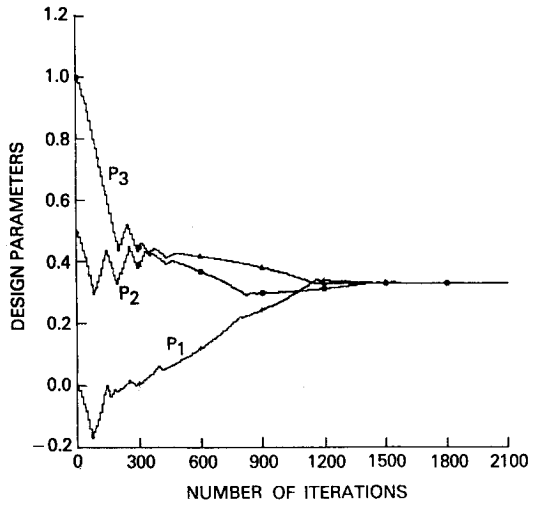

Fig. 9 Design parameters evolution histories ($c_1 = 1.1$, $n \leq 400$, $c_1 = 1.02$, $n > 400$).

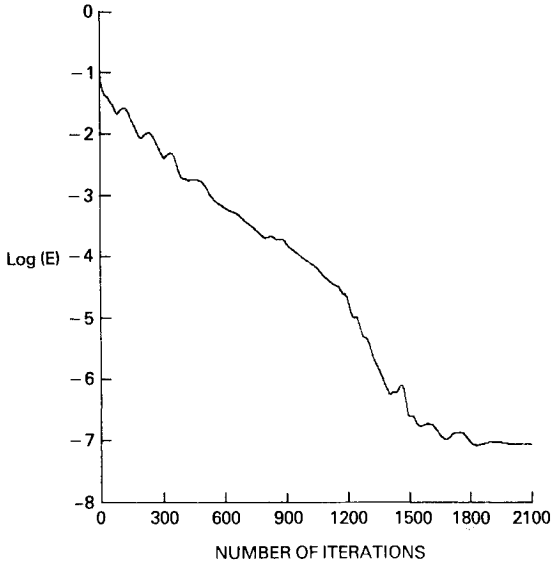

Fig. 10 Objective function evolution history (example 2).

design problem attains convergence after 1098 iterations, while the solution of the analysis problem attains convergence after 678 iterations.

For the design problem, the equivalent number of analysis iterative steps required to achieve convergence is given by $\sigma N^D = 1427$. Therefore, the number of equivalent analyses required to solve the design problem is

$$\nu_e = \sigma N^D / N^A = 2.1$$

The factor μ, which defines the efficiency of the single-cycle scheme, is then

$$\mu = \nu_e / \eta = \nu_e / (L+2) = 0.525$$

The efficiency is increased as the value of μ is reduced.

Finally, the pressure distribution along the designed airfoil is shown in Fig. 6.

In the second example, an airfoil is designed such that the objective function E, which is to be minimized, is a measure of the difference between the airfoil Mach number distribution and a desired Mach number distribution given by

$$E = \int (M_d - M)^2 ds / \int M_d^2 ds$$

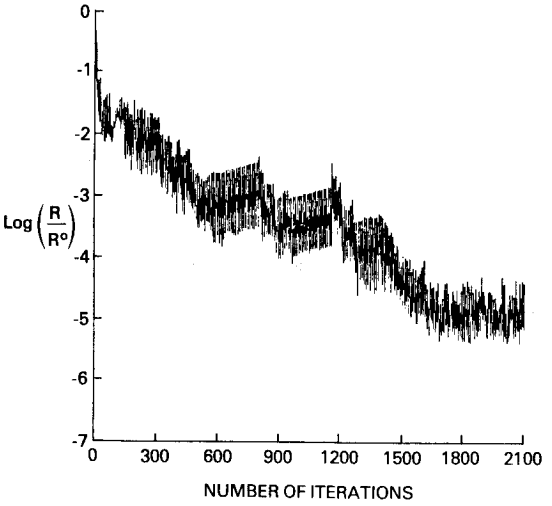

Fig. 11 Residual evolution history for the design problem (example 2).

where the integrals are taken along the airfoil surface. In general, the exact solution for the optimum design vector $\boldsymbol{P}^*$ is not known. However, in order to validate the accuracy of the optimization scheme, a special Mach number distribution that corresponds to a known $\boldsymbol{P}^*$ solution is chosen. The desired Mach number distribution M_d is chosen to be that produced by the airfoil defined by

$$y_s(x) = \sum_{\ell=1}^{3} \kappa_\ell Y^\ell(x)$$

for α and M_∞ values given by $\alpha = 2.5$ deg and $M_\infty = 0.7$, where $\kappa_1 = \kappa_2 = \kappa_3 = \frac{1}{3}$. The initial values of the design parameters are given by

$$\boldsymbol{P}^0 = \begin{bmatrix} 0.0 \\ 0.5 \\ 1.0 \end{bmatrix}$$

Figure 7 depicts the evolution histories of the design parameters. Note that the first stage of the convergence process is relatively long in comparison to the first example. Moreover, at the end of 2100 iterations convergence has not been attained.

$$R^n = 10^{-3} > 0.5 \times 10^{-3}$$

$$\max_\ell \left| \delta P_\ell^{n+1} \right| = 1.4 \times 10^{-4} > 10^{-5}$$

The relatively poor performance in the second example as compared to the first is due to several factors. The conditions in the first example resulted in incremental corrections δP_1^n, which were mostly positive throughout the first stage, and incremental corrections δP_2^n, which were mostly negative throughout the first stage. Therefore, the magnitudes of these corrections remained at their maximum allowable values $\delta P_{\max}$ throughout most of the first stage, thereby allowing a rapid approach toward the final solution. However, the conditions in the second example resulted in incremental corrections δP_ℓ^n with more frequent sign changes. These sign changes caused a reduction in the magnitude of the incremental corrections, thereby delaying the convergence process. To avoid this problem, one should choose a relatively large (1.1-1.5) incremeting factor c_1 during the first stage and a relatively small value (1.01-1.05) during the second stage. Since the end of the first stage is not known in advance, one may simply define a certain number of initial iterations where the large value for c_1 is to be used. The effect of using a small c_1 value in the first

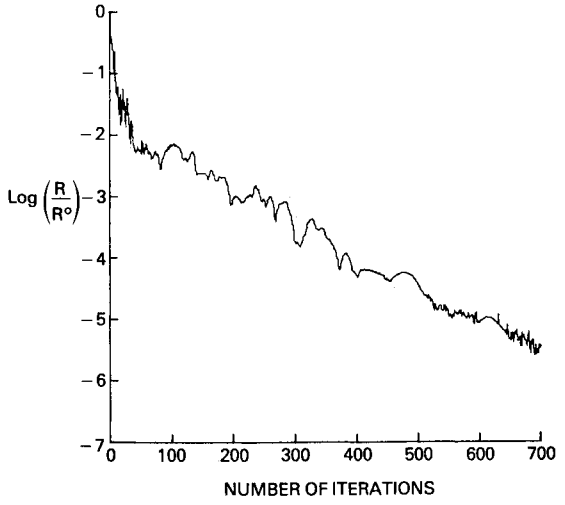

Fig. 12 Residual evolution history for the analysis problem (example 2).

stage has been demonstrated. However, the effect of using a large c_1 value in the second stage is for the iterative solutions $P_1^n,\ldots,P_L^n$ to oscillate about their converged solutions. Thus, it is possible to introduce a test for detecting such oscillations and to reduce the value of c_1 as they are detected. The second example is recalculated using the simpler of the two approaches. The values for c_1 used are 1.1 for $n \leq 400$ and 1.02 for $n > 400$.

Figure 8 compares the Mach number distribution along the airfoil surface at different stages of the iterative process to the desired Mach number distribution. The initial iterative solution is simply the free-air condition. The iterative solutions approach the desired solution as the iterative process continues. The curves showing the desired Mach number distribution and the Mach number distribution along the designed airfoil cannot be distinguished from one another. The evolution histories of the design parameters are shown in Fig. 9. The converged solutions are given by $P_1 = 0.333$, $P_2 = 0.333$, and $P_3 = 0.335$. The evolution history of the objective function is shown in Fig. 10 and the maximum residual R^n in Fig. 11. The corresponding R^n curve for the analysis problem is shown in Fig. 12.

The solution of the design problem attains convergence after 1870 iterations, while the solution for the analysis problem attains convergence after 656 iterations. The values for v_e and μ in this problem are therefore given by $v_e = 4.0$ and $\mu = 0.8$.

The value of μ for the examples solved here fall within the same range $(0.5 < \mu < 1.0)$ as those calculated for potential flows.[6,7] The values chosen here for the computational parameters c_1, c_2, c_3, ΔN, and $\delta P_{\max}$ seem to produce acceptable results; however, no attempt has been made to optimize these parameters. An optimum choice of the computational parameters will lead to the simultaneous convergence of g and P. If one of the solutions converges at a much higher rate than the other, this is an indication that further improvement in the overall convergence is obtainable by making a better choice of the computational parameters.

Conclusions

The single-cycle scheme has been applied to transonic aerodynamic design examples. The efficiency of the scheme seems to be comparable for both flows governed by the Euler equations and the potential equation. The higher efficiency for the constrained problem observed here has been previously observed. In contrast to current procedures whose efficiency decreases for constrained problems, for a single-cycle scheme the constraint being enforced while solving the problem may be viewed as an additional piece of information about the final solution. This additional information results in an accelerated convergence.

Acknowledgments

This work was supported in part under a Flow Industries Research and Technology Division internal research and development project.

References

[1]Hicks, R.M., Murman, E.M., and Vanderplaats, G.N., "An Assessment of Airfoil Design by Numerical Optimization," NASA TM X-3092, July 1974.

[2]Hicks, R.M. and Henne, P.A., "Wing Design by Numerical Optimization," *Journal of Aircraft,* Vol. 15, July 1978, pp. 407-412.

[3]Hicks, R.M., "Transonic Wing Design Using Potential-Flow Codes – Successes and Failures," SAE Paper 810565, April 1981.

[4]Vanderplaats, G.N., "Approximation Concepts for Numerical Airfoil Optimization," NASA TP 1370, March 1979.

[5]Stahara, S.S., "The Rapid Approximate Determination of Nonlinear Solutions: Application to Aerodynamic Flows and Design/Optimization Problems," *Progress in Astronautics and Aeronautics: Transonic Aerodynamics,* Vol. 81, edited by D. Nixon. AIAA, New York, 1982, pp. 637-659.

[6]Rizk, M.H., "The Single-Cycle Scheme: A New Approach to Numerical Optimization," *AIAA Journal,* Vol. 21, Dec. 1983, pp. 1640-1647.

[7]Rizk, M.H., "A Single-Cycle Scheme for Optimization Problems with Constraints," Research and Technology Div., Flow Industries, Inc., Kent, Wash., Rept. 285, Dec. 1983.

[8]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, June 1981.

[9]Salas, M.D., Jameson, A., and Melnik, R.E., "A Comparative Study of the Nonuniqueness Problem of the Potential Equation," AIAA Paper 83-1888, July 1983.