AIAA JOURNAL
Vol. 38, No. 2, February 2000

Accurate Aerodynamic Sensitivity Analysis Using Adjoint Equations

Chun-ho Sung* and Jang Hyuk Kwon†
Korea Advanced Institute of Science and Technology, Taejon 305-701, Republic of Korea

An accurate aerodynamic sensitivity analysis method for the adjoint approach is described. The present method uses the continuous adjoint approach to avoid excessive memory requirement, and the diagonalized alternating direction implicit (DADI) relaxation with multigrid acceleration is used for the fast convergence. A derivation of the continuous adjoint operator is described for the second-order total variation diminishing (TVD) scheme and a new antidiffusive flux for the adjoint equations is proposed to enhance the accuracy of sensitivity. The accuracy of the present method is demonstrated by solving sensitivity problems for two-dimensional airfoils. Comparisons with finite difference results are also presented.

Introduction

THE aerodynamic sensitivity analysis is a very important part in aerodynamic design optimization problems. It takes the large portion of the total computing time, and its accuracy is crucial to the design result.[1] Several techniques have been used for the sensitivity calculation, such as the finite difference method or the adjoint method. In the finite difference method the sensitivities are calculated from the difference of cost function with respect to the perturbation of design variables, but it requires more computing time almost in proportion to the number of design variables.[1,2]

The adjoint sensitivity analysis can be an alternative way to obtain the sensitivity. This approach can give the sensitivity information through solving adjoint equations once without many flowfield analyses. The method has been applied to the fluid dynamics problem by Pironneau[3] and Jameson[4] and applied later to the three-dimensional wing-design problem by Reuther and Jameson.[5] It has been also applied with the unstructured grid system in Refs. 6–9.

The adjoint method can be divided into two categories: one is the discrete approach,[7,9,10] and the other is the continuous approach.[4,6] The discrete approach, which solves large sparse matrices derived from discrete flow governing equations, can give very accurate sensitivity, but requires a large amount of memory to store the coefficients of adjoint equations or much computing time to recalculate the coefficients.[9] On the other hand, the continuous approach solves continuous adjoint equations derived from the differential or integral form of the flowfield governing equations with the proper discretization method. Thus, we can share the same solution procedure and discretization method between two solvers in the continuous approach. However, the continuous adjoint approach has some difficulties in deriving accurate adjoint operators for higher-order spatial discretization methods, such as MUSCL or total variation diminishing (TVD), or the complex turbulence models.[6,7,9]

The purpose of this work is to assess the accuracy of the continuous adjoint sensitivity and to develop a new accurate continuous adjoint formula for Euler equations. The second-order upwind TVD scheme[11] with Roe's approximate Riemann solver[12] is used for the Euler equations. Because the sensitivity highly depends on the numerical scheme of the flow solver, the numerical flux functions of the discretized Euler equations are considered in the discretization of adjoint equations. With this derivation the adjoint formula becomes more consistent with the discrete flow equations, and more accurate sensitivity can be obtained. Most numerical schemes for the Euler equations as well as the one used in this work have some artificial dissipation or upwinding mechanism to stabilize the solution. Because the upwind terms or artificial dissipation terms are functions of the flow variables, their variations to the flow variables should appear in the adjoint formula.[6] The contribution of these terms can be neglected because they will asymptotically converge to zero as the grid size approaches zero. However, Anderson and Venkatakrishnan showed that the error from neglecting these terms becomes larger as the grid size increases.[6] They proposed a hybrid-type adjoint formulation[6] that includes the linearization effect of Roe's Riemann solver[12,13] to the adjoint formula. In the present work the error becomes significantly large in some situations. Therefore, a new upwind flux is also adopted with linearization of Roe's Riemann solver as proposed by Anderson and Venkatakrishnan.[6]

The remaining part of sensitivity error mostly comes from the inconsistent treatment of higher-order terms. The treatment of higher-order terms is highly stuck to the discretization method for the Euler equations. For some high-order algorithms such as MUSCL, it is hard to separate the lower-order and the higher-order fluxes explicitly. Therefore, difficulties can be expected in deriving consistent adjoint formula for higher-order terms. For the TVD schemes, however, the lower-order and higher-order fluxes appear explicitly in the numerical flux as the first-order upwind flux and the antidiffusive flux. Thus, the higher-order flux for the adjoint formula can be easily derived from the discretized Euler equations. In this work the antidiffusive flux for the adjoint equations will be derived from the linearization of the Euler equations. The discretization of adjoint equations is performed to satisfy consistency with the discretized Euler equations.

To show the accuracy of the current adjoint formula, the aerodynamic sensitivity analysis for a two-dimensional airfoil is examined. The accuracy of sensitivity is assessed by comparison with the finite difference sensitivity.

General Adjoint Formulation

The cost function of aerodynamic design can be written as

$$I = I(q, d) \tag{1}$$

where I is the cost function, q is the flow variable vector, and d is the design variable vector that represents the surface shape of airfoil or perturbations on the airfoil. The aerodynamic design optimization problem is to minimize or maximize the cost function I.

The governing equations for physical flowfield can be regarded as constraints for the optimization problem because the flow variables q must satisfy the equations. The flowfield governing equations can be written as

$$R = R(q, d) = 0 \tag{2}$$

where R is the residual of governing equations.

Presented as Paper 98-4983 at the AIAA/USAF/NASA/ISSMO 7th Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, 2–4 September 1998; received 19 December 1998; revision received 15 June 1999; accepted for publication 16 June 1999.

*Doctoral Candidate, Department of Aerospace Engineering, 373-1 Kusong-Dong, Yusong-Gu.

†Associate Professor, Department of Aerospace Engineering, 373-1 Kusong-Dong, Yusong-Gu. Senior Member AIAA.

The variation of the cost function can be expressed with δq and δd as

$$\delta I = \left[\frac{\partial I^T}{\partial q}\right]_I \delta q + \left[\frac{\partial I^T}{\partial d}\right]_{II} \delta d \tag{3}$$

where subscripts I and II mean partial derivatives with respect to q and d, respectively. In the preceding formula we can see that the variation of cost function consists of two parts. The first one contains δq, which can be obtained from the flow analysis, and the second one contains δd, which can be obtained from perturbing geometry.

By introducing Lagrangian multiplier ψ, we can eliminate δq from Eq. (3) (Ref. 4). The resulting variation of the cost function δI can be written as follows:

$$\delta I = \left\{\frac{\partial I^T}{\partial d} + \psi^T \frac{\partial R}{\partial d}\right\}_{II} \delta d \tag{4}$$

The Lagrangian multiplier ψ can be obtained by solving the following adjoint equations:

$$\frac{\partial R^T}{\partial q}\psi = -\frac{\partial I}{\partial q} \tag{5}$$

In practice Eq. (5) can be derived from two different ways. It can be derived from the discretized flow governing equations or the continuous flow governing equations. In this work the continuous approach together with proper boundary conditions is used for the derivation. Finally, we can get the sensitivity G as

$$G = \frac{\delta I}{\delta d} = \left\{\frac{\partial I^T}{\partial d} + \psi^T \frac{\partial R}{\partial d}\right\}_{II} \tag{6}$$

The description of detailed derivations can be found in Jameson.[5] With sensitivity G we can find optimum values of d that maximize or minimize the cost function I by using a numerical optimization method such as the steepest descent or the quasi-Newton method.

Flow Analysis

Governing Equations

The two-dimensional compressible Euler equations can be written in the conservation form as

$$\frac{\partial \boldsymbol{q}}{\partial t} + \frac{\partial \boldsymbol{f}}{\partial x} + \frac{\partial \boldsymbol{g}}{\partial y} = 0 \tag{7}$$

where

$$\boldsymbol{q} = [\rho, \rho u, \rho v, \rho E]^T$$

$$\boldsymbol{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \end{bmatrix}, \qquad \boldsymbol{g} = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vH \end{bmatrix} \tag{8}$$

and ρ, u, v, p, E, H are the density, velocity components for x and y directions, pressure, total energy, and total enthalpy, respectively. For convenience, the physical coordinates system will be transformed to the computational coordinates (ξ, η). In the computational domain the Euler equations yield

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial \xi} + \frac{\partial \boldsymbol{G}}{\partial \eta} = 0 \tag{9}$$

where

$$\boldsymbol{Q} = \frac{1}{\boldsymbol{J}}\boldsymbol{q}, \qquad \boldsymbol{F} = \frac{1}{\boldsymbol{J}}\left(\frac{\partial \xi}{\partial \boldsymbol{x}}\boldsymbol{f} + \frac{\partial \xi}{\partial \boldsymbol{y}}\boldsymbol{g}\right), \qquad \boldsymbol{G} = \frac{1}{\boldsymbol{J}}\left(\frac{\partial \eta}{\partial \boldsymbol{x}}\boldsymbol{f} + \frac{\partial \eta}{\partial \boldsymbol{y}}\boldsymbol{g}\right)$$

and $\boldsymbol{J}$ is the transformation Jacobian that equals to the reciprocal of the cell volume.

Discretization

Equation (9) can be discretized with the cell-centered finite volume method. Integrated cell-wise in the computational domain, the governing equations yield[14]

$$\frac{\mathrm{d}}{\mathrm{d}t}(S\boldsymbol{Q})_{ij} + \boldsymbol{R}_{ij}(\boldsymbol{Q}) = 0 \tag{10}$$

where

$$R_{ij} = \boldsymbol{F}_{i+\frac{1}{2},j} - \boldsymbol{F}_{i-\frac{1}{2},j} + \boldsymbol{G}_{i,j+\frac{1}{2}} - \boldsymbol{G}_{i,j-\frac{1}{2}} \tag{11}$$

Here S_{ij} is the cell volume, and $\boldsymbol{F}, \boldsymbol{G}$ are the numerical fluxes at the cell interfaces. The numerical fluxes of the second-order TVD scheme can be constructed as follows:

$$\boldsymbol{F}_{i+\frac{1}{2},j} = \tfrac{1}{2}\left[\boldsymbol{F}_i + \boldsymbol{F}_{i+1} - (|\tilde{A}|\Delta\boldsymbol{Q} - L)_{i+\frac{1}{2}}\right] \tag{12}$$

$$\boldsymbol{G}_{i,j+\frac{1}{2}} = \tfrac{1}{2}\left[\boldsymbol{G}_j + \boldsymbol{G}_{j+1} - (|\tilde{B}|\Delta\boldsymbol{Q} - L)_{j+\frac{1}{2}}\right] \tag{13}$$

where

$$A = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{Q}} = T_\xi \Lambda_\xi T_\xi^{-1}, \qquad B = \frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}} = T_\eta \Lambda_\eta T_\eta^{-1} \tag{14}$$

The tilde means the Roe's average[12] value, and L is the antidiffusive flux of the second-order upwind TVD scheme.[11] The antidiffusive flux satisfies TVD conditions by using limiter functions. To remove nonphysical solutions, the entropy fix function Ψ is introduced. In this work Harten's entropy fix and the minmod or van Leer limiter are used.[11] The entropy function is

$$\Psi(\lambda) = \begin{cases} |\lambda|, & \lambda \geq \delta \\ \dfrac{1}{2}\dfrac{\lambda^2 + \delta^2}{\delta^2}, & \lambda < \delta \end{cases} \tag{15}$$

and the antidiffusive flux can be expressed as

$$\mathscr{L}^k = \boldsymbol{L}\left(\alpha^k_{i+\frac{1}{2}}, \alpha^k_{i+\frac{1}{2}-\sigma}\right) \tag{16}$$

$$L_{i+\frac{1}{2}} = T\Psi(\Lambda)\mathscr{L}, \qquad \alpha = T^{-1}\Delta\boldsymbol{Q} \tag{17}$$

where σ is the sign of eigenvalue, α is the difference of characteristic variables, $\boldsymbol{L}$ is the limiter function, and the superscript k indicates each component of vector variables. Finally, the numerical flux of the Euler equations can be written as follows:

$$\boldsymbol{F}_{i+\frac{1}{2},j} = \tfrac{1}{2}\left\{\boldsymbol{F}_i + \boldsymbol{F}_{i+1} - \left[\tilde{T}_\xi \tilde{\Omega}_\xi (\tilde{T}_\xi^{-1}\Delta\boldsymbol{Q} - \mathscr{L})\right]_{i+\frac{1}{2}}\right\} \tag{18}$$

where $\Omega = \Psi(\Lambda)$. By setting L equal to zero, the resulting numerical method will be spatially first-order accurate.

The steady-state solution can be obtained by the diagonalized alternating direction implicit (ADI) algorithm and the multigrid method with modified saw-tooth cycle.[15,16] The Riemann invariants are used for the boundary condition at farfield boundary and the flow tangency condition at the body surface.

Adjoint Equations

Continuous Adjoint Equations

The cost function of the aerodynamic design problem can be expressed with pressures on the body surface p and the shape of body h, which is a function of design variable d. Let us assume that the body surface can be transformed onto the ξ axis by coordinate transformation. Then, the cost function I can be written as

$$I = \int_B h(d)g(p)\,\mathrm{d}l = \int_B h(d)g(p)\,\mathrm{d}\xi \tag{19}$$

where $g(p)$ is the lift or drag coefficient or a blended function of both. By introducing costate variables, we can get the variation of cost function δI without variations of pressure δp. The full details of the derivations can be found in Jameson[5] and Reuther.[2] In this paper only the result will be described. The continuous adjoint

equations for the Euler equations derived from Eq. (9) can be written as

$$C_1^T \frac{\partial \psi}{\partial \xi} + C_2^T \frac{\partial \psi}{\partial \eta} = 0 \tag{20}$$

where

$$C_1 = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{Q}}, \qquad C_2 = \frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}} \tag{21}$$

and ψ is called by the costate variables or the Lagrangian multiplier. The boundary conditions for the adjoint equations at the body surface are determined as follows[5]:

$$\psi_2 \frac{1}{J}\frac{\partial \eta}{\partial x} + \psi_3 \frac{1}{J}\frac{\partial \eta}{\partial y} = h\frac{\mathrm{d}g}{\mathrm{d}p} \tag{22}$$

where ψ_2 and ψ_3 are the second and third components of costate variable ψ, respectively. With the preceding adjoint equations and the body surface boundary condition the variation of the cost function yields

$$\delta I = \int_B g\,\delta h\,\mathrm{d}\xi - \int_D \psi^T \left\{ \frac{\partial}{\partial \xi}\left[\delta\left(\frac{1}{J}\frac{\partial \xi}{\partial x}\right)\boldsymbol{f} + \delta\left(\frac{1}{J}\frac{\partial \xi}{\partial y}\right)\boldsymbol{g}\right] + \frac{\partial}{\partial \eta}\left[\delta\left(\frac{1}{J}\frac{\partial \eta}{\partial x}\right)\boldsymbol{f} + \delta\left(\frac{1}{J}\frac{\partial \eta}{\partial y}\right)\boldsymbol{g}\right]\right\} \mathrm{d}\xi\,\mathrm{d}\eta \tag{23}$$

where D is the computational domain and B is its boundary. The variations of metric, such as $\delta[(1/J)(\partial \xi/\partial x)]$, are referred to as grid sensitivities that can be found by the finite difference method.

Discretization of Adjoint Equations

The adjoint equations can be discretized by any stable numerical scheme. The most natural way is to use the same numerical scheme as that of the flow solver. Although the adjoint equations become a linear system of equations after the discretization, it is useful to use a transient algorithm rather than a direct inversion. With use of the transient algorithm, the memory requirement can be reduced, and the large matrix inversion problem can be avoided.[9] In this work the same procedure with the flow solver—the multigrid diagonalized alternating direction implicit (DADI) method—is used to get the steady solution. The body surface boundary conditions for the adjoint equations are Eq. (22), and ψ are set to zero at the far field.

The spatial discretization of adjoint equations requires some artificial dissipation because of its hyperbolic properties.[4] Any upwind or central method can be used. However, the numerical method that is the same as the flow solution method can be the best choice because many subroutines can be shared in the code as well as the same shortcomings and good properties with each other. Another merit, the most important one, of using the same numerical method is that the discretized adjoint equations become much more consistent with the discretized Euler equations. In other words, more accurate sensitivity can be computed.

The semidiscretized adjoint equations by using flux-difference splitting and the second-order TVD can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}(S\psi)_{ij} = R'_{ij} \tag{24}$$

where

$$R'_{ij} = \boldsymbol{F}'_{i+\frac{1}{2},j} - \boldsymbol{F}'_{i-\frac{1}{2},j} + \boldsymbol{G}'_{i,j+\frac{1}{2}} - \boldsymbol{G}'_{i,j-\frac{1}{2}} \tag{25}$$

The numerical fluxes for the adjoint equations at the cell faces are defined as

$$\boldsymbol{F}'_{i+\frac{1}{2},j} = \tfrac{1}{2}\left\{A^T\psi + \tilde{T}_\xi^{-T}\tilde{\Omega}_\xi(\tilde{T}_\xi^T\Delta\psi - \mathscr{E})\right\}_{i+\frac{1}{2}} \tag{26}$$

$$\boldsymbol{G}'_{i+\frac{1}{2},j} = \tfrac{1}{2}\left[B^T\psi + \tilde{T}_\eta^{-T}\tilde{\Omega}_\eta(\tilde{T}_\eta^T\Delta\psi - \mathscr{E})\right]_{i+\frac{1}{2}} \tag{27}$$

The preceding numerical fluxes are rational and can be obtained by straightforward discretization of the adjoint equations. Poor accuracy can result from the lack of consistency with discretized Euler equations, and so new numerical fluxes with enhanced accuracy are derived from discretized Euler equations.[2] For convenience, only ξ direction fluxes are considered, with the antidiffusive terms neglected. The numerical fluxes for η direction can be derived in the same manner. The variation of the residual of the Euler equations δR_i can be written as

$$\delta R_i = \sum_j a_{ij}\,\delta \boldsymbol{Q}_j \tag{28}$$

Now the residual of the adjoint equations R', which can be found by taking transpose of the coefficient matrix $[a_{ij}]$, yields

$$R'_i = \sum_j a_{ji}^T \psi_j \tag{29}$$

The coefficient matrix $[a_{ij}]$ can be found from the numerical flux of the Euler equations. Because the numerical fluxes have metric terms, superscripts are used to distinguish these terms. Then, a_{ij} is

$$a_{ij} = \begin{cases} A_j^{i+\frac{1}{2}} - |\tilde{A}|_{i+\frac{1}{2}}, & j = i+1 \\ A_j^{i+\frac{1}{2}} + |\tilde{A}|_{i+\frac{1}{2}} - A_j^{i-\frac{1}{2}} - |\tilde{A}|_{i-\frac{1}{2}}, & j = i \\ -A_j^{i-\frac{1}{2}} + |\tilde{A}|_{i-\frac{1}{2}}, & j = i-1 \end{cases} \tag{30}$$

where

$$A_j^{i+\frac{1}{2}} = \left(\frac{\partial f}{\partial q}\right)_j \left(\frac{1}{J}\frac{\partial \xi}{\partial x}\right)_{i+\frac{1}{2}} + \left(\frac{\partial g}{\partial q}\right)_j \left(\frac{1}{J}\frac{\partial \xi}{\partial y}\right)_{i+\frac{1}{2}} \tag{31}$$

Finally, F' can be found as

$$\boldsymbol{F}'_{i+\frac{1}{2},j} = \tfrac{1}{2}\left[\left(A_i^{i+\frac{1}{2}}\right)^T(\psi_i + \psi_{i+1}) + (|\tilde{A}^T|\Delta\psi)_{i+\frac{1}{2}}\right] \tag{32}$$

Equation (32) is first-order accurate, but its extension to the second order is straightforward by adding the antidiffusive flux. The accuracy of adjoint sensitivity with Eq. (32) becomes poor as the grid size increases.[6] Of course, sensitivities of the second-order scheme can be inaccurate because antidiffusive fluxes are not consistent. The sensitivity of the first-order scheme with the adjoint equation also differs from the finite difference result. The inaccuracy of the first-order sensitivity can be almost removed by including linearization of Roe's Riemann solver[6] in the adjoint numerical flux. The linearization of $|\tilde{A}|$ is cumbersome but has been derived previously.[6,13] With inclusion of these terms, the numerical flux should be modified as

$$R'_{ij} = \boldsymbol{F}'^{+}_i - \boldsymbol{F}'^{-}_i + \boldsymbol{G}'^{+}_j - \boldsymbol{G}'^{-}_j \tag{33}$$

$$\boldsymbol{F}'^{+}_i = \frac{1}{2}\left\{\left(A_i^{i+\frac{1}{2}}\right)^T(\psi_i + \psi_{i+1}) + \left[\left(\frac{\partial \Phi}{\partial Q_i}\right)^T \Delta\psi\right]_{i+\frac{1}{2}}\right\} \tag{34}$$

$$\boldsymbol{F}'^{-}_i = \frac{1}{2}\left\{\left(A_i^{i-\frac{1}{2}}\right)^T(\psi_i + \psi_{i-1}) + \left[\left(\frac{\partial \Phi}{\partial Q_i}\right)^T \Delta\psi\right]_{i-\frac{1}{2}}\right\} \tag{35}$$

where

$$\Phi_{i+\frac{1}{2}} = (|\tilde{A}|\Delta Q)_{i+\frac{1}{2}} \tag{36}$$

The preceding flux will be denoted by new upwind flux (NUF). With NUF the accuracy of the first-order sensitivity is somewhat enhanced, but the second-order sensitivity still has some error from the inconsistency of antidiffusive flux.

New Antidiffusive Flux for Adjoint Equations

To get more accurate sensitivity for the second-order scheme, the antidiffusive flux should be modified so that it is consistent with the discretized Euler equations. The exact linearization of antidiffusive terms is too complicated. But simple approximation of the limiter function and linearization of the antidiffusive flux help us to construct an antidiffusive flux for the adjoint equations. At the steady state the signs of eigenvalues of flux Jacobian matrices will be fixed. By taking variation with respect to Q, the variation of the

antidiffusive flux for the Euler equations at the cell face k can be expressed as

$$\delta\mathcal{E}_k = c_k^{k-1}\delta\alpha_{k-1} + c_k^k\delta\alpha_k + c_k^{k+1}\delta\alpha_{k+1} \tag{37}$$

where c_a^b are coefficients calculated from limiter functions. The subscript a means the cell face where the antidiffusive flux will be applied, and superscript b implies the cell face of three candidates where the amount of antidiffusive flux comes from. For some limiter functions, such as the minmod limiter, c_a^b become simply 0 or constant. For others, such as van Leer limiter, c_a^b will not explicitly appear in Eq. (37), but they can be found after upcoming linearization. In this case c_a^b become a function of characteristic variable α.

With the preceding representation Eq. (37) can be expressed with D, the amount of the antidiffusive flux in the residual of Euler equations, and its coefficients matrix. For simplicity, the face indices at $i-5/2$, $i-3/2$, $i-1/2$, $i+1/2$, $i+3/2$, $i+5/2$ will be represented as 0, 1, 2, 3, 4, 5, respectively, and $\tilde{T}\tilde{\Omega}$ as E. Then, δD_i are

$$\begin{Bmatrix} \delta D_{i-2} \\ \delta D_{i-1} \\ \delta D_i \\ \delta D_{i+1} \\ \delta D_{i+2} \end{Bmatrix} = H \begin{Bmatrix} \delta\alpha_1 \\ \delta\alpha_2 \\ \delta\alpha_3 \\ \delta\alpha_4 \end{Bmatrix} \tag{38}$$

where

$$H = \begin{bmatrix} h_{11} & h_{12} & & \\ h_{21} & h_{22} & h_{23} & \\ h_{31} & h_{32} & h_{33} & h_{34} \\ & h_{42} & h_{43} & h_{44} \\ & & h_{53} & h_{54} \end{bmatrix}$$

$$\begin{aligned}
h_{11} &= (E_0c_0^1 - E_1c_1^1), & h_{12} &= -E_1c_1^2 \\
h_{21} &= (E_1c_1^1 - E_2c_2^1), & h_{22} &= (E_1c_1^2 - E_2c_2^2) \\
h_{23} &= -E_2c_2^3, & h_{31} &= E_2c_2^1 \\
h_{32} &= (E_2c_2^2 - E_3c_3^2), & h_{33} &= (E_2c_2^3 - E_3c_3^3) \\
h_{34} &= -E_3c_3^4, & h_{42} &= E_3c_3^2 \\
h_{43} &= (E_3c_3^3 - E_4c_4^3), & h_{44} &= (E_3c_3^4 - E_4c_4^4) \\
h_{53} &= E_4c_4^3, & h_{54} &= (E_4c_4^4 - E_5c_5^4)
\end{aligned} \tag{39}$$

Because $\alpha_i = \tilde{T}_{i+1/2}^{-1}(Q_i - Q_{i-1})$, $\delta\alpha_i$ requires the linearization of $\tilde{T}$, which includes Roe average values. By assuming that $\tilde{T}$ is constant, $\delta\alpha_i$ can be approximated as

$$\delta\alpha_i = \tilde{T}_{i+\frac{1}{2}}^{-1}(\delta Q_i - \delta Q_{i-1}) \tag{40}$$

Then, Eq. (38) becomes

$$\begin{Bmatrix} \delta D_{i-2} \\ \delta D_{i-1} \\ \delta D_i \\ \delta D_{i+1} \\ \delta D_{i+2} \end{Bmatrix} = H \begin{Bmatrix} \tilde{T}_1^{-1}(\delta Q_{i-1} - \delta Q_{i-2}) \\ \tilde{T}_2^{-1}(\delta Q_i - \delta Q_{i-1}) \\ \tilde{T}_3^{-1}(\delta Q_{i+1} - \delta Q_i) \\ \tilde{T}_4^{-1}(\delta Q_{i+2} - \delta Q_{i+1}) \end{Bmatrix} \tag{41}$$

Expanding Eq. (41) for D_i yields

$$\{\delta D_i\} = H'\{\delta Q_i\} \tag{42}$$

where

$$\begin{aligned}
\delta D_i = {} & -E_2c_2^1\tilde{T}_1^{-1}\delta Q_{i-2} + E_2c_2^1\tilde{T}_1^{-1}\delta Q_{i-1} \\
& -(E_2c_2^2 - E_3c_3^2)\tilde{T}_2^{-1}\delta Q_{i-1} + (E_2c_2^2 - E_3c_3^2)\tilde{T}_2^{-1}\delta Q_i \\
& -(E_2c_2^3 - E_3c_3^3)\tilde{T}_3^{-1}\delta Q_i + (E_2c_2^3 - E_3c_3^3)\tilde{T}_3^{-1}\delta Q_{i+1} \\
& + E_3c_3^4\tilde{T}_4^{-1}\delta Q_{i+1} - E_3c_3^4\tilde{T}_4^{-1}\delta Q_{i+2}
\end{aligned} \tag{43}$$

Now the adjoint equations for the antidiffusive flux can be found by simply transposing H' as

$$\{D_i'\} = H'^T\{\psi_i\} \tag{44}$$

Then, the contributions of antidiffusive flux to the residual for the adjoint equations D' can be found as

$$\begin{aligned}
D_i' = {} & (E_1\tilde{T}_2^{-1})^T c_1^2(\psi_{i-1} - \psi_{i-2}) \\
& + \left[(E_2\tilde{T}_2^{-1})^T c_2^2 - (E_2\tilde{T}_3^{-1})^T c_2^3\right](\psi_i - \psi_{i-1}) \\
& + \left[(E_3\tilde{T}_2^{-1})^T c_3^2 - (E_3\tilde{T}_3^{-1})^T c_3^3\right](\psi_{i+1} - \psi_i) \\
& - (E_4\tilde{T}_3^{-1})^T c_4^3(\psi_{i+2} - \psi_{i+1}) \\
= {} & \tilde{T}_2^{-T}(c_1^2\tilde{\Omega}_1\alpha_1' + c_2^2\tilde{\Omega}_2\alpha_2' + c_3^2\tilde{\Omega}_3\alpha_3') \\
& - \tilde{T}_3^{-T}(c_2^3\tilde{\Omega}_2\alpha_2' + c_3^3\tilde{\Omega}_3\alpha_3' + c_4^3\tilde{\Omega}_4\alpha_4')
\end{aligned}$$

The difference of characteristic variables for the adjoint equations is defined as

$$\alpha' = \tilde{T}^T\Delta\psi \tag{45}$$

Finally, the new antidiffusive (NAD) flux for the adjoint equations d can be found as

$$d_k = \tilde{T}_k^{-T}\left[c_{k-1}^k(\tilde{\Omega}\alpha')_{k-1} + c_k^k(\tilde{\Omega}\alpha')_k + c_{k+1}^k(\tilde{\Omega}\alpha')_{k+1}\right] \tag{46}$$

In the preceding formulation the contributions of linearization of Roe's Riemann solver are not considered for simplicity.

The c_a^b can be found from the linearization of antidiffusive flux d. For example, the coefficient for van Leer limiter functions are derived from the antidiffusive flux, which is

$$D_i = \begin{cases} \dfrac{2\alpha_i\alpha_{i-\sigma}}{\alpha_i + \alpha_{i-\sigma}}, & \alpha_i\alpha_{i-\sigma} > 0 \\ 0, & \alpha_i\alpha_{i-\sigma} \le 0 \end{cases} \tag{47}$$

Then, the coefficients are

$$c_i^k = \begin{cases} \dfrac{2\alpha_{i-\sigma}\alpha_{i-\sigma}}{(\alpha_i + \alpha_{i-\sigma})^2}, & k = i \\ \dfrac{2\alpha_i\alpha_i}{(\alpha_i + \alpha_{i-\sigma})^2}, & k = i-\sigma \\ 0, & \text{otherwise} \end{cases} \tag{48}$$

for nonzero D_i. For the minmod limiter the coefficients can be 1 or 0, but they are inherently nondifferentiable. In this work c_a^b for the minmod limiter are modified as follows:

$$c_i^k = \begin{cases} \frac{1}{2}c_i + c_k, & k = i+\sigma \\ \frac{1}{2}c_i, & k = i \\ 0, & \text{otherwise} \end{cases} \tag{49}$$

where σ is the sign of eigenvalue.

Results and Discussion

The proposed adjoint equations are applied to a sensitivity analysis problem for an airfoil. The sensitivities of both the drag and lift coefficients are calculated with 20 Hicks–Henne functions.[17] The first 10 Hicks–Henne functions are applied to the upper surface, and the last 10 functions are applied to the lower surface. The resulting airfoil can be written as

$$y(x) = y^0(x) + \sum d_i h_i(x) \tag{50}$$

where y^0 is the initial geometry, d_i are design variables, and h_i are Hicks–Henne functions, which are defined as

$$h_1(x) = x^{0.25}(1-x)e^{-20x} \tag{51}$$

$$h_i(x) = \sin^3(\pi x^{t_i}), \qquad 2 \le i \le 10 \tag{52}$$

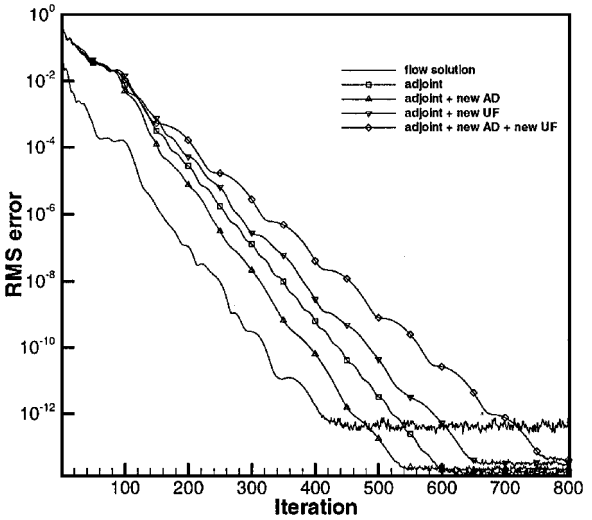


Fig. 1 Convergence histories of the flow solution and the adjoint solutions.

where $x = 0$ is the leading edge and $x = 1$ means the trailing edge. In Eq. (52) t_i locates the maximum of the sine bump, and it is defined as

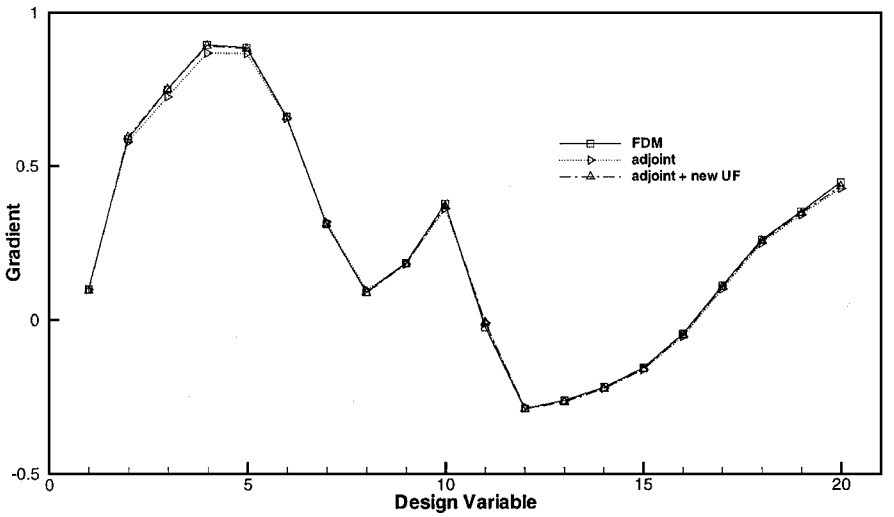
$$t_i = \tfrac{1}{10}(i - 1), \qquad 2 \le i \le 10 \tag{53}$$

Convergence of Adjoint Solutions

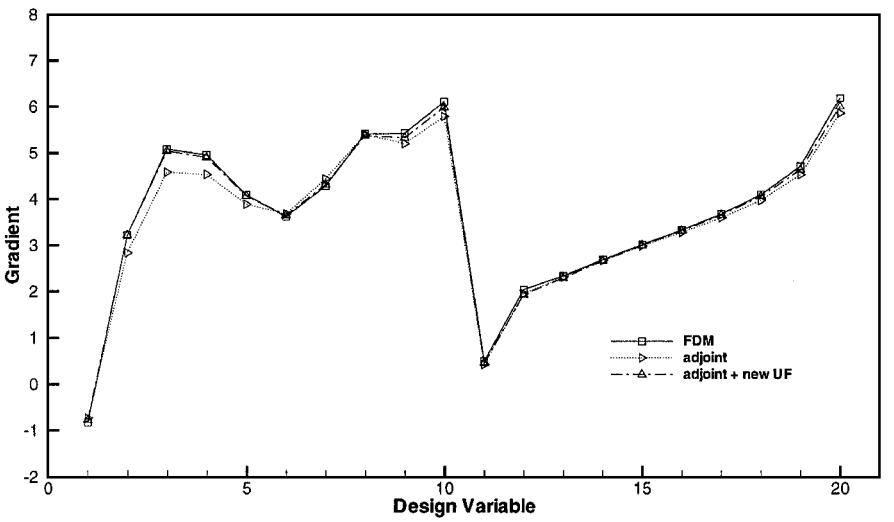
In aerodynamic design optimizations the convergence speed of the numerical scheme is important because the adjoint equations and the flow governing equations should be analyzed many times.

The solution methods, used for the adjoint equations in this work, are classified as four types according to its numerical flux. The first type is the solution method with Eq. (32) and will be designated as adjoint in each figure. The second type uses the NUF as in Eqs. (34) and (35). The third type is Eq. (32) with the NAD Eq. (46), and the last type includes both of the NAD and the NUF.

Figure 1 shows the convergence history of the Euler equations and four types of adjoint solutions. The L2 norm of density residual is plotted for the flow solution and that of the first Lagrangian multiplier is used for the history. The geometry is the RAE 2822 airfoil, and the flow Mach number is 0.73 at an angle of attack of 2.78 deg. The grid system has 129×33 points, and O-type topology is used.



Sensitivity of C_D



Sensitivity of C_L

Fig. 2 Sensitivities for the RAE 2822 with the first-order-accurate scheme.

The four-level multigrid with modified sawtooth cycle and the mesh sequencing strategy[16] are used to accelerate the convergence speed. All adjoint solutions are converged down to the machine accuracy. Because the mesh-sequencing effect is degraded in adjoint solutions, the convergence speeds of adjoint solutions at the initial stage are slower than that of the flow solution. The inclusion of NUF has an adverse effect on the convergence speed. One interpretation of this effect can be inconsistency between the implicit operator and the residual in the DADI algorithm.[16] The NAD shows two different effects. With the fourth-type numerical flux the convergence becomes about 30% slower than others. The computing times of flow and adjoint solutions are shown in Table 1, where the convergence tolerance of solutions is 10^{-8}. The calculations are performed on the Pentium II PC with 400 MHz CPU. Without the NUF the computing time of the adjoint solution is essentially the same as that of the flow solution. The NAD does not require any additional computing time, but the NUF requires about 50% additional computing time. All adjoint solutions are converged down to the four orders of magnitude within 300 multigrid cycles.

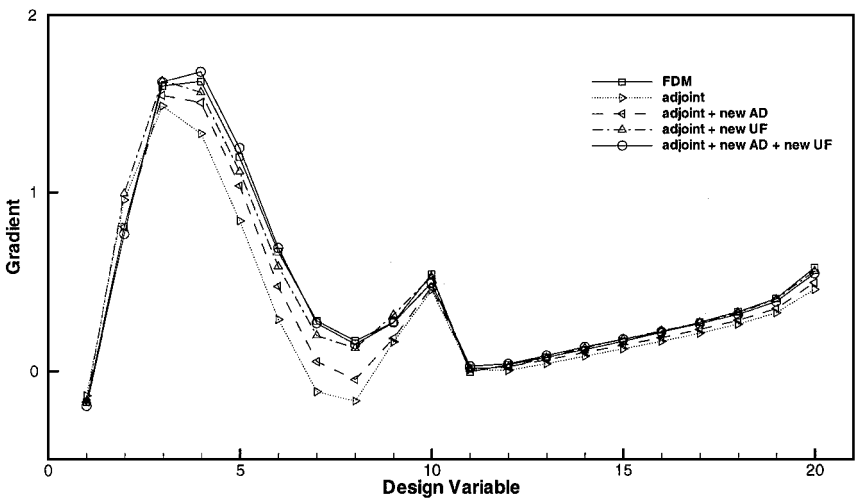
Table 1 Computing time and multigrid cycles for the flow and adjoint solutions (convergence tolerance is 10^{-8})

Solution	MG cycles	CPU, s	Seconds per cycle
Flow	250	66.1	0.26
Adjoint	350	77.6	0.22
Adj. + NAD	313	74.8	0.24
Adj. + NUF	381	129.3	0.34
Adj. + NAD + NUF	454	159.0	0.35

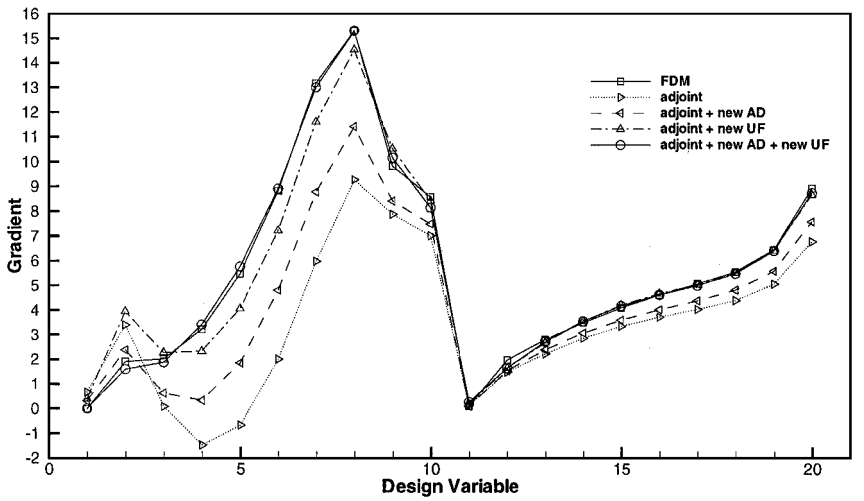
Accuracy of Adjoint Equations

To assess the accuracy of adjoint sensitivity, comparisons with the finite difference sensitivity are presented. To get accurate finite difference sensitivity, all solution procedures are restarted with a uniform flow condition.[1] Although this strategy requires much more computing time, the influence of initial solutions can be removed from the finite difference sensitivities. The flow solutions are converged down to eight orders of magnitude for both finite difference and adjoint sensitivity, and the adjoint solutions are converged down to three orders of magnitude, which is sufficient to get accurate sensitivity.[2] The adjoint sensitivity is computed from Eq. (23), and the grid sensitivity is obtained from the finite difference method.

The first case is the sensitivities of C_L and C_D for the RAE 2822 in the transonic regime. The flow condition is that the Mach number is 0.73 and the angle of attack is 2.78 deg. Under this condition the strong shock wave appears on the upper surface, so that the strength and the location of the shock wave are crucial to the C_L and C_D. Figure 2 shows the sensitivities of C_D and C_L with the first-order accurate scheme. The first 10 design variables are located on the upper surface from the leading edge to the trailing edge, and the next 10 variables are on the lower surface. The sensitivity of C_D is relatively accurate even without the NUF, but that of C_L is improved by using the NUF. Figure 3 shows the sensitivity with the second-order-accurate scheme. The role of the NUF is more important in the sensitivity of C_L than in that of C_D. The sensitivity of C_L is much larger than that of C_D because C_L is highly affected by the location of shock on the upper surface and it is highly sensitive to the shape perturbation. Also the error in the sensitivity of C_L is larger than that of C_D. Without NUF and NAD the adjoint sensitivities are somewhat smeared. It implies that the NUF and NAD are helpful to



Sensitivity of C_D



Sensitivity of C_L

Fig. 3 Sensitivities for the RAE 2822 with the second-order-accurate scheme.

predict subtle variations of the location and the strength of the shock waves. The result shows that the NUF is essential to get accurate sensitivity and the NAD also is important.

Table 2 shows the computing time of sensitivity methods. The computing times with four adjoint methods are of the same orders of magnitude; however, the computing cost with the finite difference increases in proportion to the number of design variables as expected. The adjoint method with NAD requires slightly less computing time than that without NAD because the NAD improves the convergence. Therefore, one can improve the accuracy of sensitivity by using the NAD without additional cost.

The second case is the NACA 0012 airfoil with Mach number of 0.8 and the angle of attack of 1.25 deg. Figure 4 shows the sensitivities of C_D and C_L for the NACA 0012 airfoil case where the strong shock appears on the upper surface and the weak shock appears on the lower surface.[14,16] Opposite to the first case, the sensitivities are overestimated without NAD or NUF, but the sensitivity with both of NAD and NUF shows good agreement with the finite difference result.

Table 2 Computing time of sensitivity methods (the convergence tolerance is 10^{-8} for flow and 10^{-3} for adjoint solutions)

Sensitivity method	CPU, s
Finite difference	1304.9
Adjoint	96.3
Adj. + NAD	95.5
Adj. + NUF	114.4
Adj. + NAD + NUF	114.2

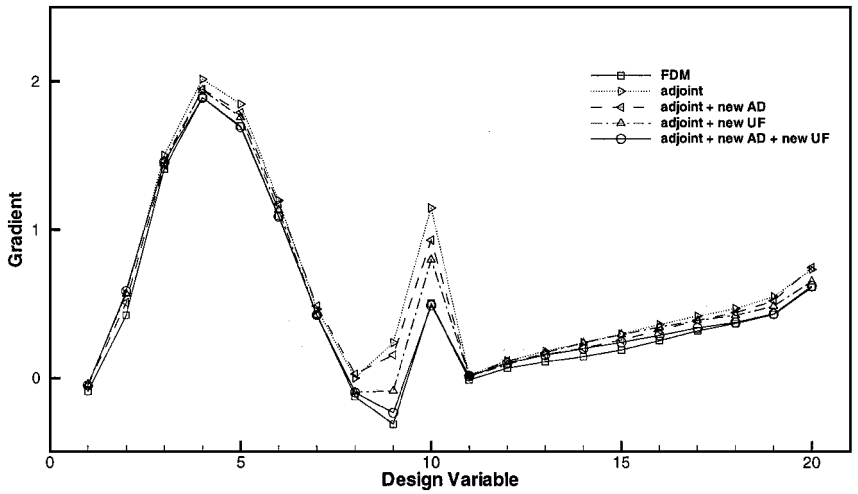
Airfoil Drag Minimization Problem

To show the applicability, the present sensitivity analysis is applied to a two-dimensional airfoil drag minimization problem under the transonic flow conditions. The initial geometry is the RAE 2822 airfoil, and the flow conditions are the same as the previous test case ($M = 0.73$, $\alpha = 2.78$). Under these flow conditions the airfoil has high wave drag caused by the strong shock wave on the upper surface. The design problem is to minimize the wave drag while maintaining the lift coefficient. The cost function of design problem is a penalty function form to keep the lift coefficient constant. The cost function can be written as

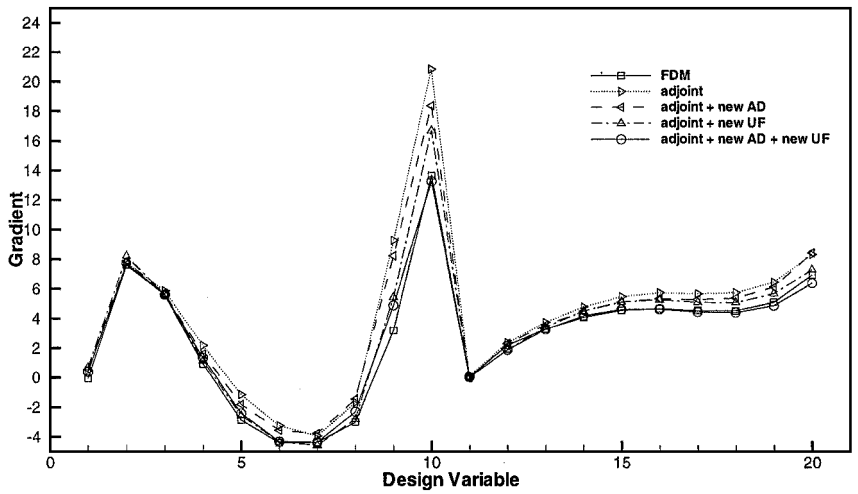
$$I = \tfrac{1}{2}(C_L - C_{L0})^2 + \tfrac{10}{2}C_D^2 \tag{54}$$

where C_{L0} is the initial lift coefficient. Five Hicks–Henne functions are applied to modify the upper surface of the airfoil. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) method and the polynomial interpolation method are used to find the optimum.[18]

Figure 5 shows the pressure distributions on the airfoil before and after the design. The strong shock wave on the upper surface is smeared after design. The designed and the initial airfoils can be seen in Fig. 6. As the result, the aerodynamic coefficients of airfoils are listed in Table 3. The drag coefficient is reduced by 42%, whereas the lift is maintained above 99% of the initial value. The design results are essentially the same for all cases; however, the NAD slightly improves the lift coefficient of the designed airfoil.



Sensitivity of C_D



Sensitivity of C_L

Fig. 4 Sensitivities for the NACA 0012 with the second-order-accurate scheme.

Table 3 Aerodynamic coefficients of the RAE 2822 and the designed airfoils

Airfoil	Sensitivity method	C_L	C_D
RAE 2822	———	0.9268	0.0227
Designed	Adjoint	0.9237	0.0132
	Adj. + NAD	0.9242	0.0132
	Adj. + NUF	0.9236	0.0132
	Adj. + NAD + NUF	0.9242	0.0132

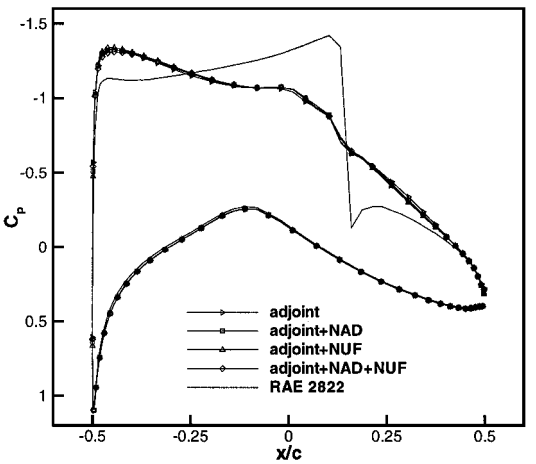


Fig. 5 Pressure distributions on the RAE 2822 and the designed airfoils.

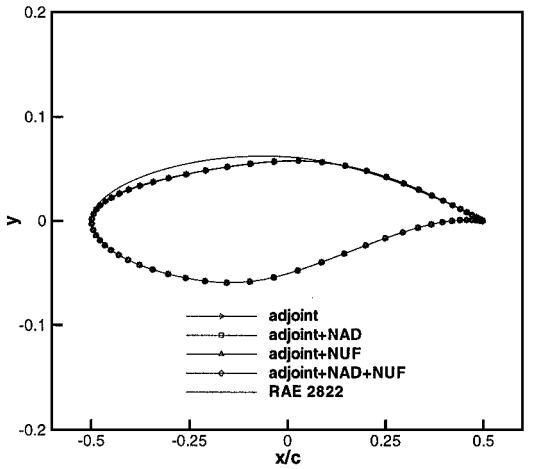


Fig. 6 Surface shapes of the RAE 2822 and the designed airfoils.

Conclusion

The purpose of the present work is to assess the accuracy of adjoint sensitivity and to develop an accurate adjoint method. The adjoint formula for the second-order upwind TVD scheme is developed from the linearization of the discrete Euler equations. The linearization procedure of the antidiffusive flux for the Euler equations is presented. The new antidiffusive flux for the adjoint equations is derived from this procedure. The linearization of Roe's Riemann solver is also included to the upwind flux for the adjoint equations.

The sensitivity with the proposed adjoint formula is assessed by analyzing an airfoil design problem in the transonic regime. The adjoint sensitivity analysis with a spatially first-order-accurate scheme gives accurate sensitivity even without the new upwind flux. With the new upwind flux the sensitivity of C_L is slightly improved. However, the accuracy of the second-order scheme is highly affected by the new upwind flux and the new antidiffusive flux. Without the new antidiffusive flux or the new upwind flux the sensitivity is smeared or overestimated. The sensitivity can be predicted accurately by applying these fluxes. However, the convergence of the new adjoint formula is somewhat degraded, but bounded in twice of that for the flow solution. In practice, because the required accuracy of the adjoint solution is usually less than one half of that for the flow solution, the total cost of one adjoint solution procedure will be less or at least the same with one flow analysis. With the test problem the present method can be applied in the usual manner without any problem.

Acknowledgment

This work was supported by the Ministry of Science and Technology under the leading technology research grant "Design of Emerging Technologies and Prevention of Failure."

References

[1]Eyi, S., and Lee, K., "Effects of Sensitivity Analysis on Airfoil Design," AIAA Paper 98-0909, Jan. 1998.

[2]Reuther, J., "Aerodynamic Shape Optimization Using Control Theory," Ph.D. Dissertation, Dept. of Mechanical and Aeronautical Engineering, Univ. of California, Davis, CA, 1996.

[3]Pironneau, O., "On Optimum Design in Fluid Mechanics," *Journal of Fluid Mechanics*, Vol. 64, 1974.

[4]Jameson, A., "Aerodynamic Design via Control Theory," *Journal of Scientific Computing*, Vol. 3, 1988, pp. 233–260.

[5]Reuther, J., and Jameson, A., "Control Theory Based Airfoil Design Using the Euler Equations," AIAA Paper 94-4272, Sept. 1994.

[6]Anderson, W., and Venkatakrishnan, V., "Aerodynamic Design Optimization on Unstructured Grids with a Continuous Adjoint Formulation," AIAA Paper 97-0643, Jan. 1997.

[7]Anderson, W., and Bonhaus, D., "Aerodynamic Design on Unstructured Grids for Turbulent Flows," NASA TM 112867, June 1997.

[8]Elliott, J., and Peraire, J., "Practical 3D Aerodynamic Design and Optimization Using Unstructured Grids," AIAA Paper 96-4170, Sept. 1996.

[9]Elliott, J., and Peraire, J., "Aerodynamic Optimization on Unstructured Mesh with Viscous Effects," AIAA Paper 97-1849, June 1997.

[10]Burgreen, G., and Baysal, O., "Three-Dimensional Aerodynamic Shape Optimization Using Discrete Sensitivity Analysis," *AIAA Journal*, Vol. 34, No. 9, 1996, pp. 1761–1770.

[11]Yee, H. C., "A Class of High-Resolution Explicit and Implicit Shock-Capturing Methods," NASA TM 101088, Feb. 1989.

[12]Roe, P. L., "Approximate Riemann Solver, Parameter Vectors and Difference Schemes," *Journal of Computational Physics*, Vol. 43, No. 2, 1981, pp. 357–372.

[13]Barth, T., "Analysis of Implicit Local Linearization Techniques for Upwind and TVD Algorithms," AIAA Paper 89-0595, Jan. 1989.

[14]Park, T. S., and Kwon, J. H., "An Improved Multistage Time Stepping for Second-Order Upwind TVD Schemes," *Computers and Fluids*, Vol. 25, No. 7, 1996, pp. 629–645.

[15]Pulliam, T., and Chaussee, D., "A Diagonal Form of an Implicit Approximate-Factorization Algorithm," *Journal of Computational Physics*, Vol. 39, 1981, pp. 347–363.

[16]Park, S.-H., Sung, C.-H., and Kwon, J. H., "An Efficient Multigrid Diagonalized ADI Method Using Second-Order Upwind TVD Schemes," *Proceedings of the KSAS Spring Annual Meeting '98*, 1998.

[17]Hicks, R., and Henne, P., "Wing Design By Numerical Optimization," *Journal of Aircraft*, Vol. 15, No. 7, 1978, pp. 407–412.

[18]Luenberger, D., *Linear and Nonlinear Programming*, Addison–Wesley Longman, Reading, MA, 1989, pp. 71–74.

A. Chattopadhyay
Associate Editor