AIAA JOURNAL
Vol. 50, No. 10, October 2012

Convergence Error and Higher-Order Sensitivity Estimations

S. Eyi*
Middle East Technical University, 06800 Ankara, Turkey

DOI: 10.2514/1.J051592

The aim of this study is to improve the accuracy of the finite-difference sensitivities of differential equations solved by iterative methods. New methods are proposed to estimate the convergence error and higher-order sensitivities. The convergence error estimation method is based on the eigenvalue analysis of linear systems, but it can also be used for nonlinear systems. The higher-order sensitivities are calculated by differentiating the approximately constructed differential equation with respect to the design variables. The accuracies of the convergence error and higher-order sensitivity estimation methods are verified using Laplace, Euler, and Navier–Stokes equations. The developed methods are used to improve the accuracy of the finite-difference sensitivity calculations in iteratively solved problems. A bound on the norm value of the finite-difference sensitivity error in the state variables is minimized with respect to the finite-difference step size. The optimum finite-difference step size is formulated as a function of the norm values of both convergence error and higher-order sensitivities. The sensitivities calculated with the analytical and the finite-difference methods are compared. The performance of the proposed methods on the convergence of inverse design optimization is evaluated.

Nomenclature

E	=	total error vector
$\hat{u}$, $\hat{v}$	=	grid velocities
$\boldsymbol{X}$	=	design variable vector
$\boldsymbol{x}$	=	grid coordinate vector
w	=	state variable vector
β, γ	=	coefficients used in higher-order sensitivity estimation
δ	=	correction vector
ε	=	convergence error vector
λ	=	eigenvalue
ξ, η	=	coordinates in computational space
φ	=	eigenvector
ψ	=	objective function

I. Introduction

THE efficient and accurate calculations of convergence error and higher-order derivatives are needed in many scientific applications. In the iterative solution of differential equations, estimating the convergence error is immensely useful to determine when to stop the iterative process. In most of the iteratively solved problems, the reduction in the residual is used as the stopping criterion. Unfortunately, the reduction in the residual may not be a reliable measure for the convergence error. In addition to the convergence error, it is also very common in scientific applications to compute the higher-order derivatives of a function with respect to specified parameters. If the evaluation of the function is expensive or involves errors due to noise, the estimation of these derivatives may be more difficult. Using the regular finite-difference methods for calculating the higher-order derivatives amplifies the noise and produces inaccurate results. In the present study, new methods are developed to estimate the convergence error and higher-order derivatives. Although these methods can be implemented for any application that requires convergence error and higher-order derivative estimations, in the present study these methods are used to improve the accuracy and efficiency of finite-difference sensitivity calculations in iteratively solved problems.

In recent years, great effort has been expended in the development of design methods using computational fluid dynamics (CFD) [1,2]. Among these methods, design optimization methods have some advantages because an optimum shape having certain characteristics can be generated while satisfying certain design constraints. The algorithms used in the optimization and CFD are independent of each other and any of the optimization and CFD codes can be coupled together. Even though design optimization methods have many advantages, there are several important issues to be resolved for optimization methods to become more efficient and reliable.

Although there is a class of optimization methods that does not require gradient information, the calculation of this information is one of the most important aspects of gradient-based design optimization. In the literature, the gradients of objective and constraint functions with respect to design variables are referred to as sensitivities. The convergence of design optimization can be significantly enhanced by improving the accuracy of sensitivities. In design optimization, a significant proportion of the computational cost can be expended in the evaluation of the sensitivities. There are different methods to calculate sensitivities: analytical, automatic differentiation, complex-step, and finite-difference. The analytical method provides accurate and efficient sensitivity calculations [3,4]. Calculating sensitivities analytically is more efficient when an adjoint method is used, in which the total cost of the sensitivity analysis is almost independent of the number of design variables [5]. The analytical method requires the development of sensitivity code by differentiating the analysis code with respect to design variables. The development of the sensitivity code is a tedious and error-prone process if hand differentiation is used. However, automatic differentiation methods have advantages over hand differentiation in reducing the effort and time spent for building the sensitivity code [6]. Automatic differentiation methods use a chain-rule-based technique for evaluating the derivatives of functions defined by computer programs. Sensitivities can also be calculated with the complex-step approach [7,8]. In this approach, the accuracy of sensitivities is almost independent of step size but the computation time and memory requirement significantly increase. Calculating sensitivities with the complex-step method may also require a major code revision, and yet this revision may be worth the effort as far as accuracy improvement is concerned. The most common method used in sensitivity calculations is the finite-difference method. The main advantage of the finite-difference approach is that it is nonintrusive. That is, it does not require additional programming to

Presented at the 20th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, June 27–30, 2011; received 2 September 2011; revision received 30 May 2012; accepted for publication 31 May 2012.

*Associate Professor, Department Aerospace Engineering. Associate Member AIAA.

build a dedicated sensitivity code. Despite this advantage, the finite-difference method has some deficiencies when it comes to accuracy and computational cost.

The main objective of the proposed methods in this study is to improve the accuracy of finite-difference sensitivities in iteratively solved problems. Many factors can affect the accuracy of finite-difference sensitivities. For example, using a computer processor or a compiler that has higher precision or employing a higher-order finite-difference stencil may reduce the errors in sensitivities. In iteratively solved problems, the accuracy of sensitivities can also be improved by evaluating the objective and constraint functions from a highly converged solution. However, these improvements may result in an increase in CPU time and computer memory use. For a given stencil, the finite-difference step size is another important factor that affects the accuracy of sensitivities. Studies related to the accuracy improvement of finite-difference sensitivities are not new. Gill et al. [9,10] developed a method to compute the optimum step size for finite-difference sensitivity calculations; Iott et al. [11] subsequently implemented similar methods for the structural design of a swept wing. Haftka [12] also introduced a modified finite-difference approach that reduces the condition error in sensitivities. Eyi and Lee [13] implemented the modified finite-difference method for inverse airfoil design. Barton [14] developed a dynamic step-size adjustment method to find the optimum step size as the optimization progresses. More and Wild estimated the finite-difference sensitivities of noisy functions.[†]

In iteratively solved problems, errors in finite-difference sensitivity calculations usually come from three different sources. The first source is the round-off error due to the finite arithmetic precision of the computer and it depends on the type of computer processor and compiler. The second source is the convergence error, which can be defined as the difference between the exact and iterative solutions of discretized governing equations. Here, the exact solution is defined as the solution that exactly satisfies the discretized governing equations with a zero residual. Although it is not always possible to reduce the residual values to the desired level, the convergence error can be reduced by solving the discretized governing equations with smaller residual tolerance. In practice, finite-difference sensitivities are evaluated when the residual norm has been reduced by three or four orders of magnitude from its original size. The third source of error in finite-difference sensitivities is the truncation error and it results from neglected terms in the Taylor series expansion. A truncation error depends on the accuracy of the finite-difference stencil used in the sensitivity evaluations and the finite-difference step size. In iteratively solved problems, if the round-off error is neglected, the total error can be defined as the summation of the convergence and truncation errors. In the present study, a method is developed to find the optimum step size that can minimize the norm value of total error in finite-difference sensitivities. Although the methods presented in this study and in Gill et al. [9,10] are similar, the present methods are implemented for different types of problems. The aim here is to find the optimum step size for the iteratively solved vector functions as opposed to the noniteratively solved scalar functions presented in Gill et al. [9,10]. In the present method, the calculation of the optimum step size is dependent on two terms: the norm values of the convergence error and the norm values of the higher-order sensitivities.

Having established the types of errors in finite-difference sensitivities, the next step is to estimate the norm values of the convergence error and higher-order sensitivities. There is great interest in estimating the convergence error, not only to improve the accuracy of sensitivity calculations but also to determine the stopping criterion of iteratively solved problems. Ferziger and Peric [15,16] used eigenvalue analysis and assumed that the system of nonlinear equations has linear behavior as it gets closer to the converged solution. Bergstrom and Gebart [17] implemented the same method to estimate the convergence error for a flow problem in a draft tube. Roy et al. [18] used an exponential equation to estimate the convergence error of hypersonic flow problems. Alekseev [19] calculated the convergence error using an adjoint parameter and time derivative. Brezinski [20] developed a method to estimate the error in the solution of linear systems. In the present study, a new method is developed that estimates the convergence error in iteratively solved problems. The method is based on the eigenvalue analysis of linear systems, as presented by Ferziger and Peric [15,16]. An equation is developed between the convergence error and correction vectors. The convergence error vector is expressed as the linear combination of the correction vectors and the coefficients of the correction vectors are calculated by the least-squares minimization of the derived equation.

In addition to the convergence error, the norm value of higher-order sensitivities is also needed for the calculation of optimum step size. Although the finite-difference method can be used for this purpose, the significant increase in the computation time is the main obstacle. In the present study, higher-order sensitivities are approximated from the previously calculated lower-order sensitivities. In recent years, there has been a growing interest in using approximate solutions in the design optimization of computationally expensive problems. Computer experiments [21] and surrogate models [22,23] are frequently used to approximate the original analysis codes. In the present study, an approximate differential equation is developed using the available state variable sensitivities. To estimate the coefficients of this differential equation, the least-squares method is employed. This method is extensively used in the estimation of the coefficients of differential equations [24–26]. Once the differential equation is constructed, the higher-order sensitivities are approximately evaluated by differentiating the differential equation with respect to the design variables.

The overall methods presented in this study can be considered semi-intrusive because estimating the convergence error and higher-order sensitivities requires some small code developments without major modification in the analysis code. However, the effort spent for the implementation of the proposed methods may be small compared to other intrusive methods. The remainder of this paper is organized as follows. In Sec. II, the sources of error in finite-difference sensitivities are analyzed for iteratively solved problems. Then, the methods developed for the estimations of convergence error and higher-order sensitivities are explained. In the last part of Sec. II, the analytical sensitivity method that is used to verify the accuracy of finite-difference sensitivities is presented. In Sec. III, first the accuracy of the convergence error estimation method is demonstrated for Laplace, Euler, and Navier–Stokes equations. Next, the performance of the higher-order sensitivity estimation method is investigated. Later, the accuracy of the optimum step-size estimation with the proposed methods is demonstrated. Finally, the effects of the finite-difference sensitivity error on the convergence of inverse aerodynamic design are presented. The conclusion is given in Sec. IV.

II. Sensitivity Analysis

In shape optimization problems, the objective function ψ can be written in the following form:

$$\psi = \psi[\boldsymbol{w}(\boldsymbol{X}), \boldsymbol{x}(\boldsymbol{X})] \tag{1}$$

where w and $\boldsymbol{x}$ are the state variable and grid coordinate vectors, respectively. In the solution of flow equations, the state variables are the flow variables. In general, the state variables and grid coordinates are functions of the design variable vector, $\boldsymbol{X}$. Although the objective function may be evaluated directly in many shape optimization problems without knowing the detailed functional relation between the objective function and state variables, the formulation shown in Eq. (1) may be useful to analyze the accuracy of sensitivities. The sensitivity of the objective function with respect to the ith design variable is

$$\frac{\mathrm{d}\psi}{\mathrm{d}X_i} = \frac{\partial\psi}{\partial w}\frac{\mathrm{d}w}{\mathrm{d}X_i} + \frac{\partial\psi}{\partial \boldsymbol{x}}\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}X_i} \tag{2}$$

The sensitivity equation, Eq. (2), includes two types of derivatives, explicit and implicit. The explicit derivatives are easy to evaluate because the function whose derivative is taken can be written as an explicit relation of the independent variables. In Eq. (2), the explicit derivatives are $\partial\psi/\partial w$, $\partial\psi/\partial \boldsymbol{x}$, and $\mathrm{d}\boldsymbol{x}/\mathrm{d}X_i$. In implicit derivatives,

[†]Data available at http://www.optimization-online.org/DB_HTML/2010/11/2787.html.

the function whose derivative is taken implicitly depends on the independent variables. The calculation of implicit derivatives may be more difficult and erroneous compared with explicit derivatives. In the same equation, the implicit derivatives are the sensitivities of the state variables, $\mathrm{d}w/\mathrm{d}X_i$, and they are usually calculated using an iterative solution technique. Because one of the objectives of this study is to improve the accuracy of finite-difference calculations of implicit sensitivity terms, the errors that occur in the implicit part of the finite-difference sensitivities are analyzed in the following subsections. Then, methods developed to estimate the convergence error and higher-order sensitivities are introduced. These methods are used in the calculation of the optimum finite-difference step size.

A. Finite-Difference Sensitivity Analysis

Any finite-difference stencil can be used in the evaluation of the state variable sensitivities. In a firstorder forward-difference scheme, the sensitivity vector with respect to the ith component of design variable can be calculated as

$$\frac{\Delta w}{\Delta X_i} = \frac{w(\boldsymbol{X} + \Delta X_i) - w(\boldsymbol{X})}{\Delta X_i} \tag{3}$$

where $w(\boldsymbol{X})$ and $w(\boldsymbol{X} + \Delta X_i)$ are the state variable vectors for the base and perturbed geometries, respectively, which are assumed to be calculated from the exact solution of discretized governing (flow) equations. This solution is the one that exactly satisfies the discretized governing equations with a zero residual. The finite-difference sensitivities calculated using the exact values of state variables have truncation error due to neglected terms in the Taylor series expansion,

$$\frac{\Delta w}{\Delta X_i} = \frac{\mathrm{d}w(\boldsymbol{X})}{\mathrm{d}X_i} + \frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\frac{\Delta X_i}{2} \tag{4}$$

where $\mathrm{d}w/\mathrm{d}X_i$ and $\mathrm{d}^2 w/\mathrm{d}X_i^2$ are the first- and second-order analytical sensitivity vectors, respectively, evaluated at $\boldsymbol{X}$ and $\varsigma \in [\boldsymbol{X}, \boldsymbol{X} + \Delta X_i]$, respectively. Analytical sensitivity vectors can be calculated by analytically differentiating the discretized form of governing equations with respect to the design variables.

An exact solution of discretized governing equations is only available for certain special problems. The calculation of the exact solution for an arbitrary problem is difficult. In general, the state variables used in finite-difference stencils are calculated from an iterative solution of discretized governing equations. The solution obtained from an iterative method may not exactly satisfy the discretized governing equations. There are two main types of errors in the iterative calculation of state variables. The first type is the round-off error that occurs because the state variables are calculated using a computer capable of handling a fixed number of digits. The round-off error is related to the precision error, which depends on the type of computer processor and compiler that is used. The precision error is defined according to the machine epsilon, which is the smallest floating-point number that can be added to 1.0 with a result that does not round to 1.0. The machine epsilon is a very small number; for example, in this study, it has a value of 5×10^{-17}. Even though the round-off error may play an important role in the stability and the convergence of numerical methods, in stable methods it may be neglected when compared with other errors[‡] [27]. One of the advantages of using an iterative method is that the round-off error in iterative methods does not accumulate as it does in a direct method [28,29]. The second type of error is the convergence error. In iteratively solved problems, the state variables are evaluated when the norm value of residuals in discretized governing equations is reduced to a certain fraction of its original size. Solving equations with a nonzero residual may result in error in the state variables and this is called convergence error. In the present calculations, all floating-point variables are calculated with double precision and the round-off error is neglected compared with the convergence error.

Let $\tilde{w}(\boldsymbol{X})$ and $\tilde{w}(\boldsymbol{X} + \Delta X_i)$ denote the iteratively computed values of $w(\boldsymbol{X})$ and $w(\boldsymbol{X} + \Delta X_i)$, respectively. The relations between the computed and exact values of the state variable vectors can be written as

$$\begin{aligned} \tilde{w}(\boldsymbol{X}) &= w(\boldsymbol{X}) + \varepsilon(\boldsymbol{X}) \\ \tilde{w}(\boldsymbol{X} + \Delta X_i) &= w(\boldsymbol{X} + \Delta X_i) + \varepsilon(\boldsymbol{X} + \Delta X_i) \end{aligned} \tag{5}$$

where $\varepsilon(\boldsymbol{X})$ and $\varepsilon(\boldsymbol{X} + \Delta X_i)$ are the convergence error vectors for the base and perturbed geometries, respectively. Then, sensitivities with the computed state variable vectors can be evaluated as

$$\frac{\Delta \tilde{w}}{\Delta X_i} = \frac{\tilde{w}(\boldsymbol{X} + \Delta X_i) - \tilde{w}(\boldsymbol{X})}{\Delta X_i} \tag{6}$$

The relation between the computed and exact finite-difference sensitivities becomes

$$\frac{\Delta \tilde{w}}{\Delta X_i} = \frac{\Delta w}{\Delta X_i} + \frac{\varepsilon(\boldsymbol{X} + \Delta X_i) - \varepsilon(\boldsymbol{X})}{\Delta X_i} \tag{7}$$

Substituting Eq. (4) into Eq. (7) gives a relation between the computed finite-difference and analytical sensitivities

$$\frac{\Delta \tilde{w}}{\Delta X_i} = \frac{\mathrm{d}w(\boldsymbol{X})}{\mathrm{d}X_i} + \frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\frac{\Delta X_i}{2} + \frac{\varepsilon(\boldsymbol{X} + \Delta X_i) - \varepsilon(\boldsymbol{X})}{\Delta X_i} \tag{8}$$

The total error vector, $E(\Delta X_i)$, can be defined as the difference between the computed finite-difference and analytical sensitivities

$$E(\Delta X_i) = \frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\frac{\Delta X_i}{2} + \frac{\varepsilon(\boldsymbol{X} + \Delta X_i) - \varepsilon(\boldsymbol{X})}{\Delta X_i} \tag{9}$$

Taking the norm values of both sides of Eq. (9) and applying the triangle inequality gives

$$\|E(\Delta X_i)\| \le \left\|\frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\right\|\frac{\Delta X_i}{2} + \frac{1}{\Delta X_i}\{\|\varepsilon(\boldsymbol{X} + \Delta X_i)\| + \|\varepsilon(\boldsymbol{X})\|\} \tag{10}$$

In the present study, $\tilde{w}(\boldsymbol{X})$ and $\tilde{w}(\boldsymbol{X} + \Delta X_i)$ are evaluated when the norm values of $\varepsilon(\boldsymbol{X})$ and $\varepsilon(\boldsymbol{X} + \Delta X_i)$, respectively, are reduced to a predetermined level, $\|\varepsilon_A\|$. The value of $\|\varepsilon_A\|$ is chosen according to the desired accuracy level of the solution. The convergence error vectors, $\varepsilon(\boldsymbol{X})$ and $\varepsilon(\boldsymbol{X} + \Delta X_i)$, are calculated by using the method presented in the next section. Thus, Eq. (10) becomes

$$\|E(\Delta X_i)\| \le \left\|\frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\right\|\frac{\Delta X_i}{2} + \frac{2}{\Delta X_i}\|\varepsilon_A\| \tag{11}$$

The right-hand side of Eq. (11) approximates an upper bound of the norm value of the total error vector. Minimizing the right-hand side of the equation also minimizes an upper bound on the total error in the forward-difference sensitivities. Taking the derivative of the right-hand side of Eq. (11) with respect to the step size, ΔX_i, and solving the resulting equation yield the optimum step size

$$\Delta X_{i\mathrm{opt}} = 2\sqrt{\frac{\|\varepsilon_A\|}{\|[\mathrm{d}^2 w(\varsigma)]/(\mathrm{d}X_i^2)\|}} \tag{12}$$

The optimum step size defined in Eq. (12) minimizes an upper bound of the total error, and it is well defined when the norm values of second-order sensitivities are nonzero. Substituting Eq. (12) into the right-hand side of Eq. (11) gives a bound on the norm value of total error evaluated with optimum step size,

$$\|E(\Delta X_{i\mathrm{opt}})\| \le 2\sqrt{\|\varepsilon_A\|\left\|\frac{\mathrm{d}^2 w(\varsigma)}{\mathrm{d}X_i^2}\right\|} \tag{13}$$

By assuming that the norm values of third-order sensitivities are nonzero and performing a similar calculation to that of the forward-difference method, the optimum step size in the central-difference method can be evaluated as

$$\Delta X_{i\mathrm{opt}} = \sqrt[3]{\frac{3\|\varepsilon_A\|}{\|[\mathrm{d}^3 w(\mu)]/(\mathrm{d}X_i^3)\|}} \tag{14}$$

[‡]Data available at http://www.grc.nasa.gov/WWW/wind/valid/tutorial/errors.html.

where, $\mu \in [X - \Delta X_i, X + \Delta X_i]$. In the central-difference method, a bound on the total error is minimized when the optimum step size, given in Eq. (14), is used. For this case, a bound on the norm value of total error can be evaluated as

$$\|E(\Delta X_{i\text{opt}})\| \leq \frac{1}{2}\sqrt[3]{9(\|\varepsilon_A\|)^2\left\|\frac{\mathrm{d}^3 w(\mu)}{\mathrm{d}X_i^3}\right\|} \tag{15}$$

In the first-order sensitivity calculations, the magnitude of the optimum step size and the resulting sensitivity error are functions of the norm values of the convergence error and higher-order state sensitivities. In the past, similar relations were proposed to calculate the optimum step size for scalar objective functions [9,10]. In most of the previous studies, the objective functions were evaluated directly, without using an iterative solution technique. As a consequence, only the round-off error was considered in the calculation of the objective functions. In iteratively solved problems, in addition to the round-off error, the convergence error has to be considered. There are limited studies on the calculation of the optimum step size for iteratively calculated vector-valued functions[§]; thus, the motivation in this study is to calculate the optimum step size for a state variable vector that is evaluated using an iterative solution method.

B. Convergence Error Estimation

The calculation of the optimum finite-difference step size requires the norm values of the convergence error. In this section, a new method to estimate the convergence error in iteratively solved problems is presented. Although the developed method is based on the eigenvalue analysis of linear systems, it can also be used for nonlinear systems, especially during near convergence when nonlinear systems behave like linear systems and for which the error estimation is most needed.

In this section, the iterative solution of a system of linear equations is reviewed first. Similar reviews can also be found in Ferziger and Peric [15,16]. The system of linear equations can be defined in the following form:

$$Aw = b \tag{16}$$

in which w is the exact solution of the system. An iterative scheme can be constructed by splitting matrix A as follows:

$$A = M - N \tag{17}$$

In this splitting, M is chosen so that the system can be easily solvable with an iterative scheme,

$$M\tilde{w}^{n+1} = N\tilde{w}^n + b \tag{18}$$

where $\tilde{w}^n$ is the iterative solution of the state variable vector after n iterations. Because w is the exact solution of the system, it also satisfies the iterative scheme given in Eq. (18),

$$Mw = Nw + b \tag{19}$$

At iteration n, the convergence error vector can be defined as the difference between the iterative and exact solution vectors,

$$\varepsilon^n = \tilde{w}^n - w \tag{20}$$

Defining the correction vector δ^n as

$$\delta^n = \tilde{w}^{n+1} - \tilde{w}^n \tag{21}$$

and using Eq. (20), the following relation between the correction and convergence error vectors can be written as

$$\delta^n = \varepsilon^{n+1} - \varepsilon^n \tag{22}$$

Subtracting Eq. (19) from Eq. (18) gives a relation between the convergence error vectors of two successive iterates,

$$\varepsilon^{n+1} = M^{-1}N\varepsilon^n \tag{23}$$

The iterative method converges if the spectral radius of the matrix $M^{-1}N$ is less than one. The convergence of the iterative scheme can be analyzed with the use of eigenvalues, λ_k, and eigenvectors, φ_k,

$$M^{-1}N\varphi_k = \lambda_k\varphi_k, \qquad k = 1, KMAX \tag{24}$$

where $KMAX$ is the total number of state variables in the system. For the case of complex eigenvalues, the following equation is also satisfied:

$$M^{-1}N\varphi_k^* = \lambda_k^*\varphi_k^*, \qquad k = 1, KMAX \tag{25}$$

where λ_k^*, and φ_k^* are the complex conjugates of the eigenvalues and eigenvectors, respectively. Most of the time, iterative methods have complex eigenvalues. In the present derivation, considering the more general case and assuming real numbers as the special case of complex numbers, complex eigenvalues and eigenvectors are used. Using the linearly independent eigenvectors, the initial error ε^0 may be expressed as a linear combination of eigenvectors,

$$\varepsilon^0 = \sum_{k=1}^{KMAX} a_k\varphi_k + a_k^*\varphi_k^* \tag{26}$$

where a_k *and* a_k^* are generalized Fourier coefficients. The combination of Eqs. (23) and (26) yields

$$\varepsilon^1 = \sum_{k=1}^{KMAX} M^{-1}N(\varphi_k a_k + \varphi_k^* a_k^*) \tag{27}$$

Substituting Eqs. (24) and (25) into Eq. (27) gives

$$\varepsilon^1 = \sum_{k=1}^{KMAX} a_k\lambda_k\varphi_k + a_k^*\lambda_k^*\varphi_k^* \tag{28}$$

By induction, the error vector at iteration n can be written as

$$\varepsilon^n = \sum_{k=1}^{KMAX} a_k(\lambda_k)^n\varphi_k + a_k^*(\lambda_k^*)^n\varphi_k^* \tag{29}$$

After a number of iterations, the contribution of the largest eigenvalue, λ_1, becomes more significant and the error vector can be approximated as

$$\varepsilon^n = a_1(\lambda_1)^n\varphi_1 + a_1^*(\lambda_1^*)^n\varphi_1^* \tag{30}$$

The derivation of these equations can also be found in Ferziger and Peric [15,16]. In the present study, the following method is developed to estimate the convergence error, which is one of the original contributions of this paper. Rearranging Eq. (30) yields the following relationship between the convergence error vectors at iteration $n+1$, n, and $n-1$:

$$\varepsilon^{n+1} = (\lambda_1 + \lambda_1^*)\varepsilon^n - \lambda_1\lambda_1^*\varepsilon^{n-1} \tag{31}$$

It is not difficult to show that the convergence error vector can be expressed as a function of the correction vectors,

$$\varepsilon^{n+1} = \frac{(C_1 + C_2)\delta^n + C_2\delta^{n-1}}{C_1 + C_2 - 1} \tag{32}$$

Substituting Eq. (22) into Eq. (31) gives the following relationship between the correction vectors at iteration n, $n-1$, and $n-2$:

$$\delta^n = C_1\delta^{n-1} + C_2\delta^{n-2} \tag{33}$$

in which C_1 and C_2 are real numbers and functions of eigenvalues

$$C_1 = \lambda_1 + \lambda_1^* \qquad C_2 = -(\lambda_1\lambda_1^*) \tag{34}$$

In the calculation of the convergence error vector, first, the coefficients C_1 and C_2 are determined from the least-squares solution of Eq. (33). Then, using Eq. (32), the error vector can be evaluated as a function of the correction vectors of two successive iterations. To calculate the coefficients, the correction vectors from the current and previous two iterations must be stored. In this method, the calculation of the values of eigenvalues is not required because the values of coefficients C_1 and C_2 are sufficient to determine the convergence error vectors.

Increasing the number of eigenvalues may improve the convergence error estimation. For example, approximating Eq. (29) by using the first and second largest eigenvalues, the convergence error vector can be formulated as

$$\varepsilon^n = a_1(\lambda_1)^n\varphi_1 + a_1^*(\lambda_1^*)^n\varphi_1^* + a_2(\lambda_2)^n\varphi_2 + a_2^*(\lambda_2^*)^n\varphi_2^* \tag{35}$$

The following relation can be derived to calculate the convergence error vector at iteration $n+1$,

[§]Data available at http://www.optimization-online.org/DB_HTML/2010/11/2787.html.

$$\varepsilon^{n+1} = \frac{(C_1 + C_2 + C_3 + C_4)\delta^n + (C_2 + C_3 + C_4)\delta^{n-1} + (C_3 + C_4)\delta^{n-2} + (C_4)\delta^{n-3}}{C_1 + C_2 + C_3 + C_4 - 1} \tag{36}$$

Similarly, the relation between the correction vector at iteration n and the correction vectors at iterations $n-1$, $n-2$, $n-3$, and $n-4$ can be written as

$$\delta^n = C_1\delta^{n-1} + C_2\delta^{n-2} + C_3\delta^{n-3} + C_4\delta^{n-4} \tag{37}$$

In parallel with the previous derivations, the real coefficients C_1, C_2, C_3, and C_4 in Eq. (36) are the functions of eigenvalues and these coefficients are calculated from the least-squares solution of Eq. (37),

$$\begin{aligned} C_1 &= \lambda_1 + \lambda_1^* + \lambda_2 + \lambda_2^* \\ C_2 &= -(\lambda_1\lambda_1^* + \lambda_1\lambda_2 + \lambda_1\lambda_2^* + \lambda_1^*\lambda_2 + \lambda_1^*\lambda_2^* + \lambda_2\lambda_2^*) \\ C_3 &= \lambda_1\lambda_1^*\lambda_2 + \lambda_1\lambda_1^*\lambda_2^* + \lambda_1\lambda_2\lambda_2^* + \lambda_1^*\lambda_2\lambda_2^* \\ C_4 &= -(\lambda_1\lambda_1^*\lambda_2\lambda_2^*) \end{aligned} \tag{38}$$

In the calculation of these coefficients, the correction vectors must be stored from the current and last four iterations. As explained in the calculation of the convergence error vector, the values of eigenvalues are not required. The convergence error vector can be calculated as a function of coefficients C_1, C_2, C_3, and C_4.

The relations given in the preceding equations can be generalized for an arbitrary number of eigenvalues. Approximating Eq. (29) by using the M_{eigen} number of eigenvalues, the convergence error becomes

$$\begin{aligned} \varepsilon^n = {} & a_1(\lambda_1)^n\varphi_1 + a_1^*(\lambda_1^*)^n\varphi_1^* + \ldots + a_{M_{\text{eigen}}}(\lambda_{M_{\text{eigen}}})^n\varphi_{M_{\text{eigen}}} \\ & + a_{M_{\text{eigen}}}^*(\lambda_{M_{\text{eigen}}}^*)^n\varphi_{M_{\text{eigen}}}^* \end{aligned} \tag{39}$$

By induction, the convergence error vector can be generalized for an arbitrary number of eigenvalues, M_{eigen}, as

$$\varepsilon^{n+1} = \frac{(\sum_{m=1}^{2M_{\text{eigen}}} C_m)\delta^n + (\sum_{m=2}^{2M_{\text{eigen}}} C_m)\delta^{n-1} + (\sum_{m=3}^{2M_{\text{eigen}}} C_m)\delta^{n-2} + \ldots + (C_{2M})\delta^{n-2M_{\text{eigen}}+1}}{\sum_{m=1}^{2M_{\text{eigen}}} C_m - 1} \tag{40}$$

Similarly, the correction vector can be generalized as

$$\delta^n = \sum_{m=1}^{2M_{\text{eigen}}} C_m\delta^{n-m} \tag{41}$$

As previously explained, the coefficients C_m in Eq. (40) are real numbers, and they are determined from the least-squares solution of Eq. (41). In the calculation of the coefficients, the correction vectors from the current and last $2M_{\text{eigen}}$ iterations must be stored. Although increasing the number of eigenvalues may improve the accuracy of the convergence error estimation, this improvement may also result in an increase in the amount of memory required to store the correction vectors from the previous iterations.

In many iteratively solved problems, the convergence rate is superlinear, so that the convergence accelerates as the error gets smaller. For example, in Newton's method, a quadratic convergence rate can be achieved and the residual can be reduced to the order of machine epsilon in a few iterations. In such cases, the estimation of convergence error with an arbitrary number of eigenvalues may be difficult. There may not be sufficient iterations to determine the coefficients in Eq. (41). A smaller number of eigenvalues may be used in these problems.

C. Higher-Order Sensitivity Estimation

In a finite-difference scheme, estimating the optimum step size for the first-order sensitivity calculations requires the norm values of higher-order sensitivities. As stated in Eqs. (12) and (14), the optimum step sizes in the forward- and central-difference schemes are inversely proportional to the norm values of the second- and third-order sensitivities, respectively. One of the easiest approaches is to approximate the norm values of these sensitivities as unity. However, this approximation may degrade the accuracy of sensitivities. Another approach is to use the finite-difference method. Although this method is more accurate, it requires additional function evaluations and step-size estimation. An algorithm for estimating an appropriate step size for the second-order sensitivities of a scalar objective function is given in Gill et al. [9,10]. In this algorithm, the optimum step size is decided if the values of the relative condition error are within an acceptable range (0.001–0.1). The condition error is defined as the error in sensitivities due to the inaccurate values of the objective function. Even though this method is useful for estimating the step size of the second-order sensitivities, satisfying the relative condition error in a given range may require several function evaluations and thus may significantly increase the computational cost. Therefore, using a finite-difference method to estimate the second-order sensitivities may not be very practical if there are a large number of design variables.

In the design optimization of complex and computationally expensive systems, computer experiments and statistical methods are frequently used [21–23]. The basic approach in these methods is the construction an approximate or surrogate model of the computationally expensive analysis code. Another original contribution of this paper is the development of an approximate method to estimate the norm values of the higher-order state sensitivities. Here, the intention is not to achieve highly accurate calculations of higher-order sensitivities but to estimate the order of magnitude without significantly increasing the computer's CPU time and memory usages. Most of the laws in physics and technology are governed by simple linear relationships between the function and its first-order derivative. Although different types of approximation are possible, the following approximate differential equation is constructed because of its simplicity

$$\beta_i \frac{\mathrm{d}w(\boldsymbol{X})}{\mathrm{d}X_i} + \gamma_i - w(\boldsymbol{X}) = 0 \tag{42}$$

where $w(\boldsymbol{X})$ and $\mathrm{d}w(\boldsymbol{X})/\mathrm{d}X_i$ are the state variable and the first-order sensitivity vectors of ith design variable, respectively. The coefficients β_i and γ_i are constants, to be determined by a least-squares method. The following function, $\mathcal{F}_i$, is minimized for the known values of the kth state variables and sensitivities:

$$\mathcal{F}_i = \sum_{k=1}^{KMAX} \left(\frac{\mathrm{d}w_k}{\mathrm{d}X_i} + \gamma_i - w_k\right)^2 \tag{43}$$

where $KMAX$ is the size of the state variable or sensitivity vectors and is equal to the total number of state or sensitivity variables in the solution domain. Taking the derivatives of Eq. (43) with respect to the

coefficients of β_i and γ_i and setting the resulting equations to zero, a system of equations can be constructed. Once the unknown coefficients β_i and γ_i are calculated, the differential equation given in Eq. (42) is differentiated with respect to design variables X_i and the second-order sensitivities can be approximated as

$$\frac{\mathrm{d}^2 w_k}{\mathrm{d}X_i^2} \approx \left(\frac{\mathrm{d}w_k}{\mathrm{d}X_i}\right) \Big/ \beta_i \tag{44}$$

In Eq. (44), evaluating the value of β_i from Eq. (42) gives another relation for the second-order sensitivities,

$$\frac{\mathrm{d}^2 w_k}{\mathrm{d}X_i^2} \approx \left(\frac{\mathrm{d}w_k}{\mathrm{d}X_i}\right)^2 \Big/ (w_k - \gamma_i) \tag{45}$$

The second-order state sensitivities calculated with Eqs. (44) and (45) have the same value if the coefficients β_i and γ_i exactly satisfy Eq. (42). However, if the coefficients are calculated with a least-squares approximation, the differential equation is not exactly satisfied and Eqs. (44) and (45) may give different values. In the present study, the geometric mean of Eqs. (44) and (45) is used to estimate the magnitude of second-order sensitivities,

$$\left|\frac{\mathrm{d}^2 w_k}{\mathrm{d}X_i^2}\right| \approx \sqrt{\frac{(\mathrm{d}w_k/\mathrm{d}X_i)^3}{\beta_i(w_k - \gamma_i)}} \tag{46}$$

Estimating the optimum step size for the central-difference scheme requires the norm values of the third-order sensitivities. The third-order sensitivities can be estimated by differentiating the approximate differential equation twice. Similar to the derivations of Eqs. (44) and (45), two equations can be derived to estimate the third-order sensitivities. The geometric mean of these equations can be used to calculate the third-order sensitivities,

$$\left|\frac{\mathrm{d}^3 w_k}{\mathrm{d}X_i^3}\right| \approx \sqrt{\frac{(\mathrm{d}^2 w_k/\mathrm{d}X_i^2)^2(\mathrm{d}w_k/dX_i)}{\beta_i(w_k - \gamma_i)}} \tag{47}$$

As in the forward-difference scheme, the coefficients β_i and γ_i in Eq. (47) can be estimated from the least-squares minimization of Eq. (42). However, in the central-difference scheme, a better method is available to estimate these coefficients. The following differential equation can be constructed by differentiating Eq. (42) with respect to design variable X_i:

$$\beta_i \frac{\mathrm{d}^2 w_k}{\mathrm{d}X_i^2} - \frac{\mathrm{d}w_k}{\mathrm{d}X_i} = 0 \tag{48}$$

In Eq. (48), the second-order sensitivities can be evaluated using the information available from the calculation of the first-order sensitivities with the central-difference scheme. Hence, in the central-difference scheme, in addition to Eq. (42), Eq. (48) can also be minimized with respect to the coefficients β_i and γ_i. In this study, for the central-difference scheme, the following function is minimized to evaluate the coefficients of β_i and γ_i:

$$\mathcal{F}_i = \sum_{k=1}^{KMAX} \left[\left(\beta_i \frac{\mathrm{d}w_k}{\mathrm{d}X_i} + \gamma_i - w_k\right)^2 + \left(\beta_i \frac{\mathrm{d}^2 w_k}{\mathrm{d}X_i^2} - \frac{\mathrm{d}w_k}{\mathrm{d}X_i}\right)^2 \right] \tag{49}$$

In the proposed approximate method, some entries of the higher-order sensitivity vector are not defined because the value inside the square root of Eqs. (46) and (47) may be negative. However, this situation may not cause any problems in the calculation of the norm values of higher-order sensitivities as long as the number of these entries is small. Undefined values of higher-order sensitivities can be excluded in the calculation of the norm values. Having undefined entries in higher-order sensitivity vectors may be related to using an inappropriate form of differential equation in Eq. (42). More research may be needed to decide on the form of the approximate differential equation.

The main advantage of using the approximate method is that when compared to first-order sensitivity calculations, the increase in CPU time and computer memory usage for the higher-order sensitivity calculations is almost negligible. In the approximate method, a 2×2 matrix is solved for each design variables. Once the β_i and γ_i coefficients are evaluated, the second- and third-order state sensitivities can be calculated using Eqs. (46) and (47), respectively.

In design optimization, the following procedures can be used in the implementation of the approximate method for the optimum step-size calculation. At the first iteration of the design, the optimum step sizes can be calculated by assuming the norm values of the higher-order sensitivities to be unity. Starting from the first iteration, after the first-order sensitivities are evaluated, the approximate method can be used to estimate the higher-order sensitivities. Assuming that the norm values of the higher-order sensitivities do not change significantly from one design iteration to next, the norm values evaluated from the previous iteration can be used to estimate the optimum step size in the current iteration.

In this study, the accuracy of the approximate method is verified with the finite-difference method. The algorithm given in Gill et al. [9,10] is modified to calculate the finite-difference step size of the higher-order sensitivities for vector-valued functions. The original algorithm is used to calculate the finite-difference step size of the second-order sensitivities for scalar objective functions. In the modified algorithm, the absolute values of scalar functions are replaced by the norm values of vector functions.

D. Analytical Sensitivity Analysis

In this study, analytical sensitivities are used to verify the accuracy of the finite-difference sensitivities. To develop the analytical sensitivity code, the discretized Euler equations used in finite-difference sensitivity calculations are differentiated with respect to the design variables. A brief description of the analytical sensitivity method is presented below. Detailed information about the method can be found in Eyi and Lee [30].

The two-dimensional unsteady Euler equations in the Cartesian coordinates can be written as

$$\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0 \tag{50}$$

where w is the flow variable vector and f and g are the flux vectors in the x and y directions, respectively. The system given in Eq. (50) can be transformed from a physical space (x, y) to a computational space (ξ, η) as

$$\frac{\partial W}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial G}{\partial \eta} = 0 \tag{51}$$

where

$$W = wh, \qquad h = x_\xi y_\eta - x_\eta y_\xi$$
$$F = fy_\eta - gx_\eta, \qquad G = gx_\xi - fy_\xi$$

In the preceding equations, x_ξ, y_ξ, x_η, and y_η are the transformation metrics and h is the Jacobian of the transformation. The characteristic boundary conditions are used for the far-field. On an airfoil surface, zero normal mass fluxes are enforced and the pressures are extrapolated using the normal momentum equation.

In analytical sensitivity calculations, the material derivative approach is used in which the sensitivity of a flow variable consists of two parts. The first part is the rate of change of the flow variable at a fixed point of domain and is often referred to as the local sensitivity, denoted as $\partial w/\partial X_i$. The second part is the convective part, representing the change due to the variation of the domain itself,

$$\frac{\mathrm{d}w}{\mathrm{d}X_i} = \frac{\partial w}{\partial X_i} + \hat{V} \cdot \nabla w \tag{52}$$

Here, $\hat{V}$ is called the design shape velocity or grid velocity and its components, $\hat{u}$ and $\hat{v}$, can be defined as

$$\hat{V} = (\hat{u}, \hat{v}), \qquad \hat{u} = \frac{\mathrm{d}x}{\mathrm{d}X_i}, \qquad \hat{v} = \frac{\mathrm{d}y}{\mathrm{d}X_i} \tag{53}$$

In the calculation of the analytical sensitivities, a direct differentiation method is used. In this method, the Euler equations are differentiated with respect to design variables X_i,

$$\frac{\mathrm{d}}{\mathrm{d}X_i}\left(\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y}\right) = 0 \tag{54}$$

Eq. (54) can be written in generalized coordinates as

$$\frac{\partial \bar{W}}{\partial t} + \frac{\partial \bar{F}}{\partial \xi} + \frac{\partial \bar{G}}{\partial \eta} + H = 0 \tag{55}$$

where

$$\bar{W} = \frac{\mathrm{d}w}{\mathrm{d}X_i}h, \qquad \bar{F} = \frac{\mathrm{d}f}{\mathrm{d}X_i}y_\eta - \frac{\mathrm{d}g}{\mathrm{d}X_i}x_\eta, \qquad \bar{G} = \frac{\mathrm{d}g}{\mathrm{d}X_i}x_\xi - \frac{\mathrm{d}f}{\mathrm{d}X_i}y_\xi,$$
$$H = \frac{\partial}{\partial \xi}(f\hat{v}_\eta - g\hat{u}_\eta) + \frac{\partial}{\partial \eta}(g\hat{u}_\xi - f\hat{v}_\xi) + \frac{R}{h}\frac{\mathrm{d}h}{\mathrm{d}X_i}$$

In the preceding equations, the residual of the flow analysis, R, can be neglected when the flow analysis is fully converged. The boundary conditions of Eq. (55) are implemented by differentiating the boundary conditions of the Euler equations with respect to the design variables. In the solution of the Euler and sensitivity equations, the same numerical discretization methods are used.

III. Results

The performance of the optimum step-size estimation method is demonstrated. As already stated, the optimum finite-difference step size can be evaluated as a function of two parameters, the norm values of the convergence error and the higher-order sensitivities. In this section, first, the accuracy of the methods developed for estimating the convergence error and the higher-order sensitivities are verified. Then, the performances of the calculation of the optimum step size with the proposed methods are demonstrated. Finally, the effects of the optimum step-size calculation on the convergence of inverse aerodynamic design are investigated.

A. Convergence Error Estimation

The accuracy of the method for estimating the convergence error is tested for both linear and nonlinear problems. All calculations in this study were performed with double precision on a 1.5 GHz Pentium IV dual core processor. As a linear problem, a two-dimensional Laplace equation in a square domain ($0 < x < 1; 0 < y < 1$) is solved using the successive over-relaxation method with a relaxation parameter of Ω. The Laplace equation is one of the few equations whose exact solution can be analytically evaluated with the appropriate boundary conditions. A second-order central-difference scheme is used on a uniform Cartesian grid. The boundary condition is chosen to satisfy the solution $\varphi(x, y) = 100xy$. Because the terms related to the truncation error are eliminated, this analytical solution is also the exact solution of the Laplace equation that is discretized with the second-order and central-difference scheme [15,16]. The real value of the convergence error is calculated as the difference between the exact solution and the computed solution from the current iteration. The estimated error is defined as the error calculated with the proposed method explained in Sec. II.B. To have better understanding of the order of the error in each entry of the convergence error vector, the norm values are redefined according to following relations:

$$\|\varepsilon\|_p = \left(\frac{1}{KMAX}\sum_{k=1}^{KMAX}|\varepsilon_k|^p\right)^{\frac{1}{p}},$$
$$\|\varepsilon\|_\infty = \max(|\varepsilon_1|, \ldots, |\varepsilon_{KMAX}|) \tag{56}$$

The first equation given in Eq. (56) for $p = 1$ and $p = 2$ corresponds to the absolute-mean and root-mean-square, respectively. Unless otherwise stated, in all figures to predict the convergence error, the number of eigenvalues, M_{eigen}, is set to 8. Solving the Laplace equation with a grid size of 80×80 gives the number of state variables, $KMAX$, of 6400.

Figure 1a shows the $\|\varepsilon\|_2$ norm values of the real and estimated convergence errors and their differences. The sole reason for showing the norm value of the residual vector in Fig. 1a is to see the effectiveness of using residual values to estimate the convergence error. The norm values of the residual are almost two orders of magnitude smaller than that of the convergence error. Hence, instead of the convergence error, if the residual was used in Eq. (12), the optimum step size in the forward-difference method would be one order of magnitude smaller than the actual values. The results show that although the residual itself is not a good parameter to predict the convergence error, the proposed method can accurately estimate the convergence error. As the number of iterations increases, the difference between the real and estimated errors decreases. In Fig. 1b, the effects of the number of eigenvalues on the estimation of the convergence error are presented. The results show that increasing the number of eigenvalues from 1 to 128 improves the accuracy of the error prediction. Increasing the number of eigenvalues decreases the amplitude of oscillation in the estimated error; however, this requires more memory to store the correction vectors. As shown in Eq. (41), using M_{eigen} number of eigenvalues entails the storing of the correction vectors from the last $2M_{\text{eigen}}$ iterations. In the estimation of the convergence error with M_{eigen} number of eigenvalues, the convergence error estimation starts after $2M_{\text{eigen}}$ number of iterations. In the present study, until that iteration is reached, the convergence error is estimated with the maximum number of available eigenvalues. The convergence error estimation starts at the fourth iteration by using only one eigenvalue. Between iterations four and $2M_{\text{eigen}}$, the convergence error at iteration n is estimated using the $n/2$ or $(n-1)/2$ number of eigenvalues, depending on whether the iteration number is even or odd, respectively.

The norm values of the convergence error estimated using the proposed method and the method given in Ferziger and Peric [15,16] are compared in Fig. 1c. In this comparison, the value of the over-relaxation parameter, Ω, is set to 1.95, which is the maximum attainable value for this grid size. The solution diverges with higher values of Ω. This figure clearly shows the superiority of the proposed method. In the method given in Ferziger and Peric, the convergence error is estimated by using the largest eigenvalue, hence, the accuracy level of the method is fixed. However, the accuracy level of the proposed method can be adjusted by changing the number of eigenvalues. In Fig. 1c, the proposed method uses two different numbers of eigenvalues, 8 and 128. In terms of estimating the convergence error, the proposed method with 8 eigenvalues is more successful than the method given in Ferziger and Peric. Figure 1c also shows that the accuracy of the proposed method improves as the number of eigenvalues increases. If the proposed method uses 128 eigenvalues, the real and estimated errors cannot be distinguished after 100 iterations. Figure 1d shows the effects of the value of the relaxation parameter, Ω, on the estimation of the convergence error. In all cases, the values of estimated and real errors are very close to each other.

Next, the performance of the convergence error estimation method is analyzed for nonlinear problems. Two-dimensional Euler and Navier–Stokes equations are solved around a NACA 0012 airfoil at transonic flow conditions ($M = 0.730$, $\alpha = 2.78$ deg, $Re = 6.5 \times 10^6$). Solving the equations at transonic flow condition increases the nonlinearity in the solution. A finite-volume method is used for spatial discretization and the flow variables are defined at the cell centers. Centered differencing is used for the spatial derivatives and the second-order and fourth-order artificial viscosities are added to enforce numerical stability. Time integration is performed using an explicit four-stage Runga–Kutta scheme. Local time stepping and a multigrid method are implemented to accelerate the convergence to obtain a steady-state solution. The multigrid level for the Euler and

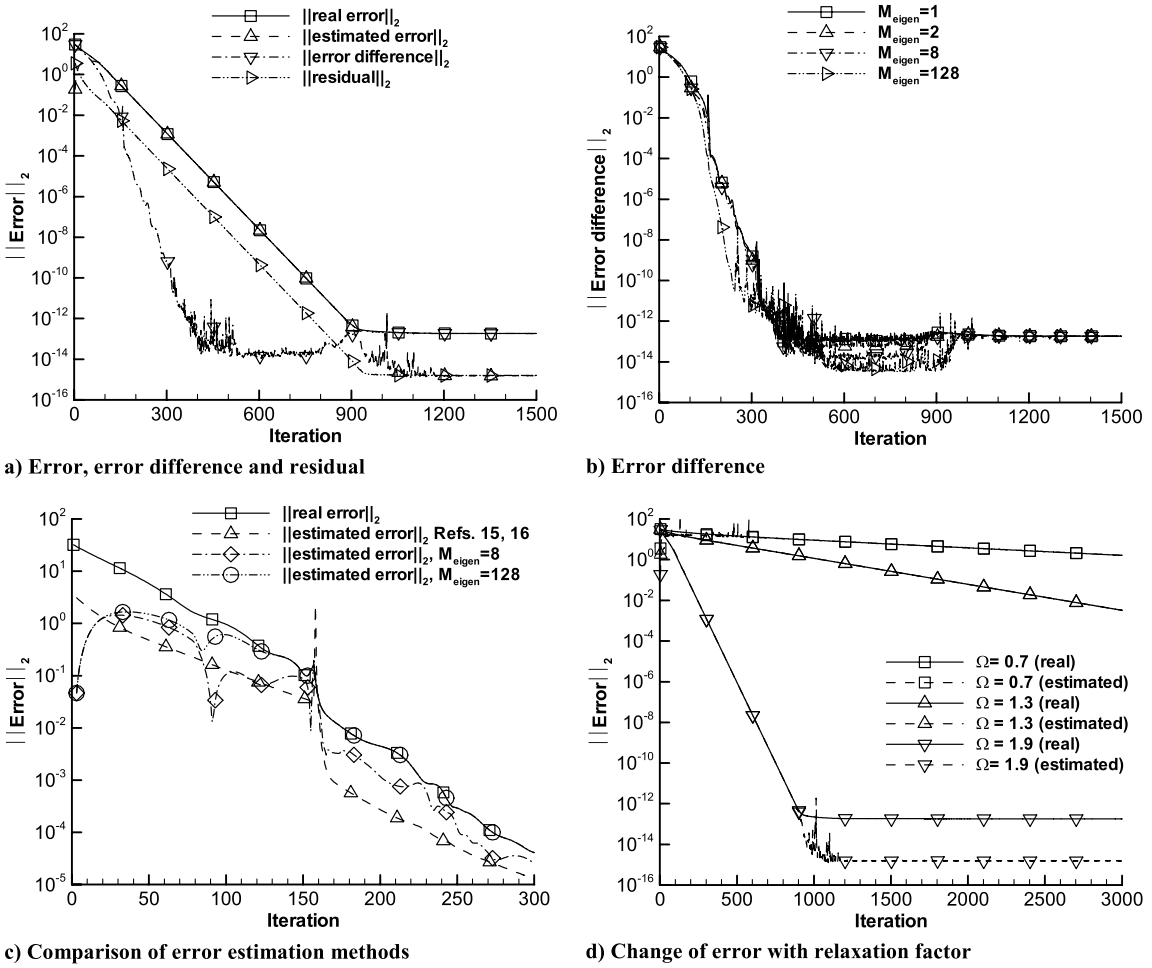


Fig. 1 Convergence error estimation for the Laplace equation.

Navier–Stokes solutions are three and four, respectively. Characteristic boundary conditions are imposed at the far-field boundary based on a one-dimensional eigenvalue analysis. In the solution of the Euler equations, zero normal mass flux is enforced at the airfoil surface and the pressures are extrapolated from the inside cells using the normal momentum equation. In the Navier–Stokes solutions, no-slip and adiabatic wall conditions are used at the airfoil surface and the Baldwin–Lomax eddy-viscosity model is included for turbulence closure. In the Euler computations, the grid size is 129×33. In the Navier–Stokes computations, the grid size is 257×65 and the minimum grid spacing next to the wall is 10^{-5}. Having four variables in each cell, the values of *KMAX* are 17028 and 66820 in the solutions of the Euler and Navier–Stokes equations, respectively. It is difficult to find analytical relations for the exact solution of the Euler and Navier–Stokes equations. Therefore, in these problems, the exact solutions are estimated by iterating solutions until the residual norm becomes on the order of a machine epsilon in double precision. The real convergence error is calculated as the difference between the solutions for the current iteration and the exact solution. As in the solution of the Laplace equation, the estimated convergence error is calculated by using the method presented in Sec. II.B. The convergence error is calculated on the finest grid of a multigrid cycle and all four conservative flow variables are used in the error estimation.

As a nonlinear problem, the convergence error analysis is first performed on the Euler equations. Figure 2a shows the change of the real and estimated errors and their difference in relation to the number of iterations. In this figure, the convergence error is estimated using 8 eigenvalues. Although the equations are nonlinear and the flow conditions enforce the nonlinearity, the real and estimated errors are almost the same. The difference between the estimated and real errors decreases as the number of iterations increases. Figure 2a also shows the variation of the residual with respect to the number of iterations, and it can be seen that there is a large difference between the residual and convergence error. Therefore, the residual may not be a good parameter to predict the error. The effects of the number of eigenvalues on the estimation of the convergence error are analyzed in Fig. 2b. Results show that the accuracy of convergence estimation method can be significantly improved by increasing the number of eigenvalues. Figure 2c shows that in all three norms, the proposed method predicts the convergence error very accurately. From Fig. 2d, it can be seen that the value of the Courant–Friedrichs–Lewy (CFL) number does not affect the performance of the proposed error estimation method.

Last, the performance of the proposed method is demonstrated for the Navier–Stokes equations, in which there are additional nonlinear terms related to viscous fluxes and turbulence modeling. The real and estimated errors are determined by methods similar to those used in the Euler equations. Figure 3a shows that the proposed method can estimate the convergence error very accurately for nonlinear problems. The estimated error almost exactly matches the real error and large reductions are achieved in the difference between the estimated and real errors. In Fig. 3b, it can be seen that increasing the number of eigenvalues decreases the difference between the real and estimated error. However, these improvements in error estimation are achieved at the cost of a large increase in CPU time. For many engineering problems, setting the number of eigenvalues between 2 and 8 may be sufficient. The real and estimated errors comparison using different norms is shown in Fig. 3c, and again, excellent results are achieved for nonlinear problems. In iterative schemes, it is not always possible to reduce the residual values to the order of the round-off error. In the solution of the Navier–Stokes equations, only the CFL number of 1.5 reduces the residual values to that order. Hence, the effects of the CFL numbers on the accuracy of the convergence error estimation method could not be explored for the

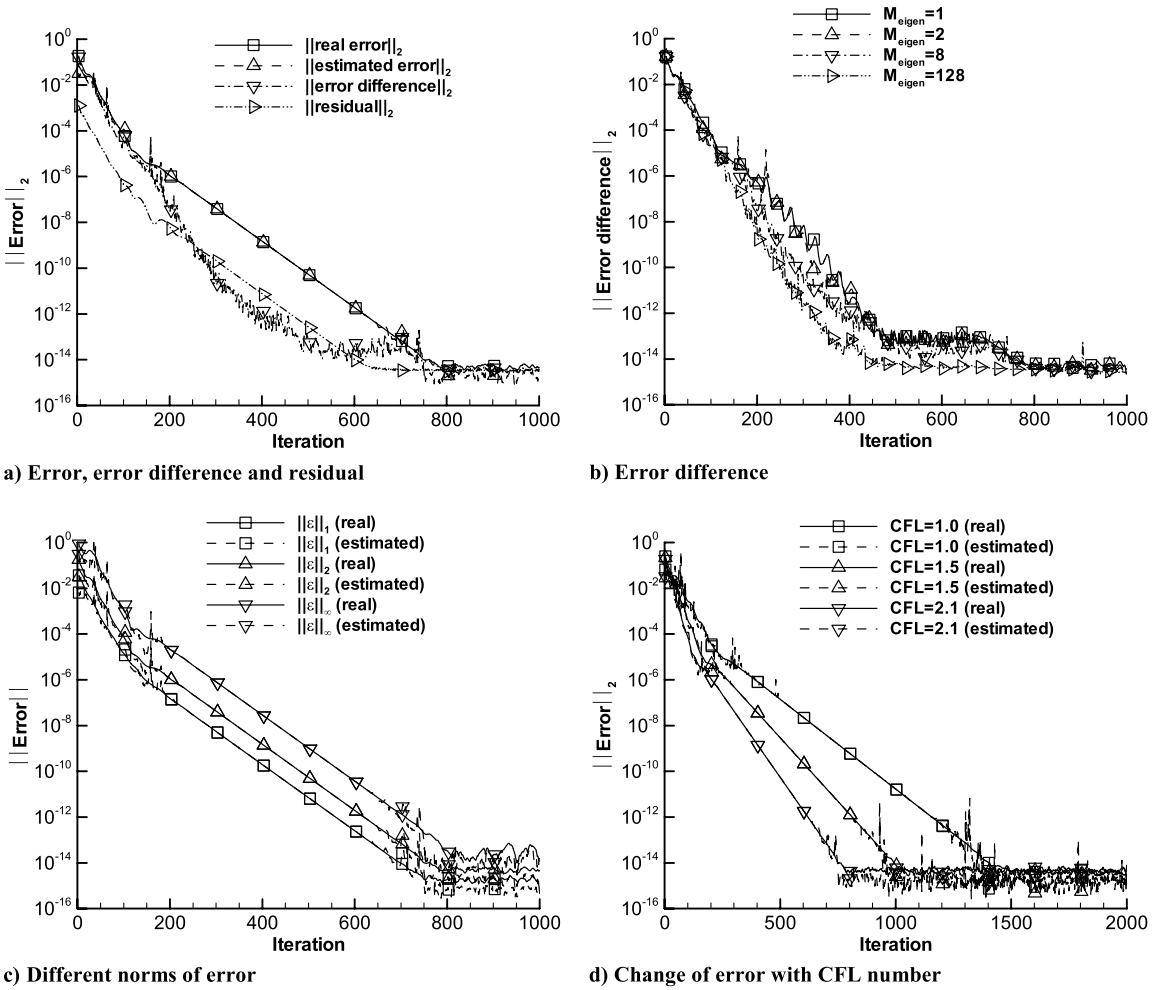


Fig. 2 Convergence error estimation for the Euler equations.

Navier–Stokes equations. Figure 3d compares real and estimated errors at the initial iterations with the estimated error being calculated with different numbers of eigenvalues. Results show that the convergence error can be accurately estimated in the first twenty iterations, which is a sizeable advantage for finite-difference sensitivity calculations.

B. Higher-Order Sensitivity Estimation

In the finite-difference calculation of the first-order sensitivities, the estimation of the optimum step size requires the norm values of the higher-order sensitivities, i.e., the second-order sensitivities for the forward-difference scheme and the third-order sensitivities for the central-difference scheme. The finite-difference calculation of higher-order sensitivities significantly increases the computational time. In Sec. II.C, the approximate method was proposed to estimate the norm values of the higher-order state sensitivities. Even though this method is computationally efficient, the accuracy of the method may be degraded because the differential equation used to estimate the higher-order sensitivities is approximately constructed. In this section, the accuracy of the approximate method is verified using the finite-difference method. A central-difference scheme is used to calculate the value of the higher-order sensitivities. In the estimation of an appropriate finite-difference step size, an algorithm similar to the one given in Gill et al. [9,10] is implemented. The higher-order state sensitivities are evaluated by solving the Euler and Navier–Stokes equations for the same geometry, flow conditions, and grids used in the convergence error estimation. The sizes of the state variable and sensitivity vectors used in the convergence error and higher-order sensitivity estimations are the same; that is, the values of *KMAX* are 17028 and 66820 in the solutions of the Euler and Navier–Stokes equations, respectively. In higher-order sensitivity calculations using finite-difference and approximate methods, the norm values of convergence error in state variables is reduced to 10^{-16}. The Hicks–Henne and Wagner shape functions [31] are used to modify the airfoil geometry and the weighting coefficient of these shape functions are used as design variables. The total geometry change perpendicular to the airfoil chord is defined as a linear combination of shape functions,

$$\Delta y = \sum_{i=1}^{IMAX} X_i f_i(x) \tag{57}$$

In Eq. (57), Δy is the total geometry change perpendicular to the airfoil chord, f_i is the shape function, which is controlled by the ith design variable, X_i, x is the chord-wise location, and $IMAX$ is the total number of design variables. In the calculation of the sensitivities, all the design variables are fixed except the one for which the sensitivity is calculated. Once the airfoil geometry is modified, a new grid is generated by translating the old grid points with a distance of Δy. A total of fourteen design variables are used; the first seven variables modify the upper surface of the airfoil and the remaining variables modify the lower surface.

First, the Euler equations are used in the comparison of the higher-order sensitivities calculated with the finite-difference and the proposed approximate methods. The results with the Hicks–Henne functions are shown in Fig. 4a for the norms of $\|.\|_2$. Each Hicks–Henne function has only one maximum, and the location of the maximum in a chord-wise direction can be controlled by the user. Depending on the location of the maximum, each member of the Hicks–Henne functions is effective in modifying particular parts of the airfoil. The norm values of the higher-order state sensitivities have the largest values for the third design variable, while the

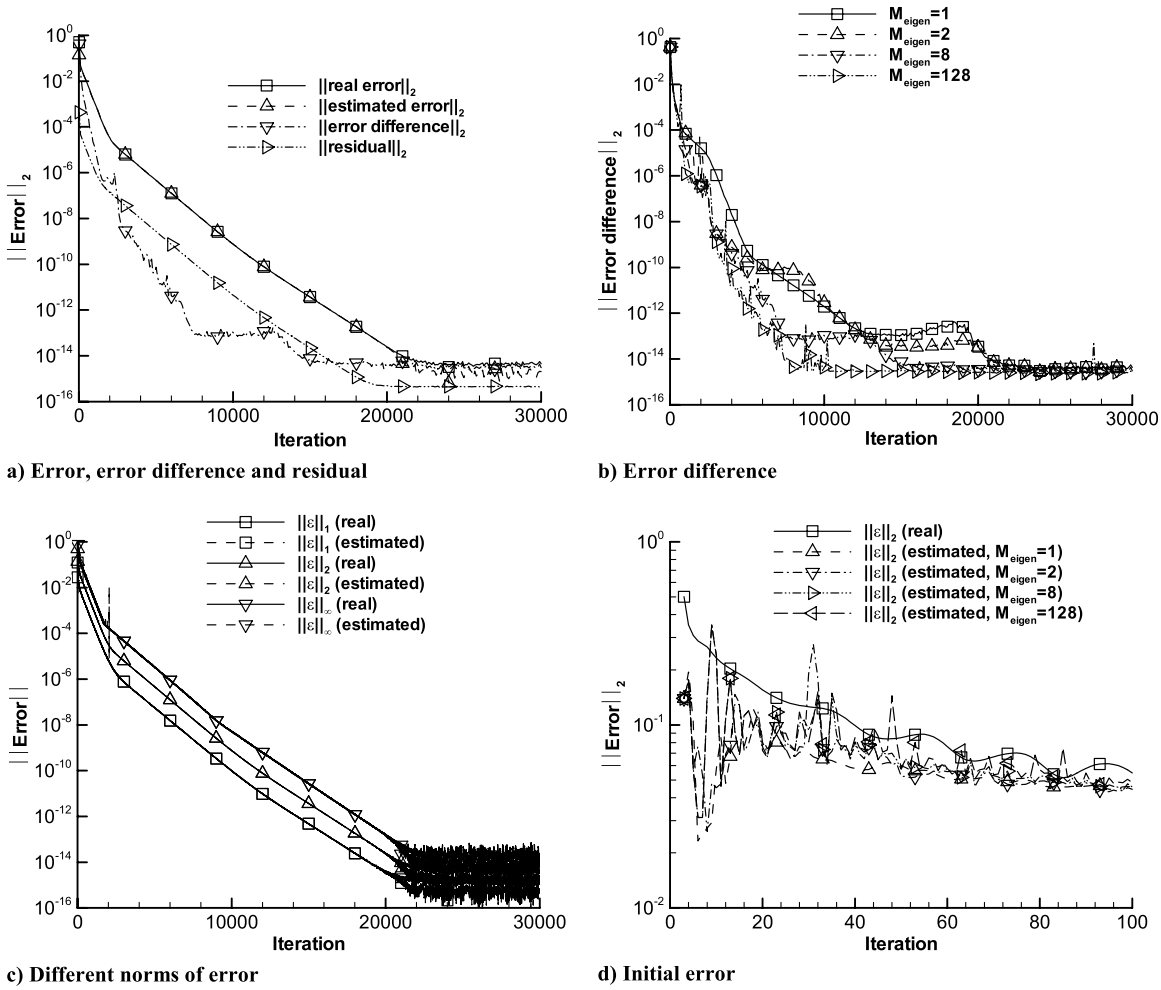


Fig. 3 Convergence error estimation for the Navier–Stokes equations.

higher-order state sensitivities of the seventh and fourteenth design variables have the second largest values. The eighth design variable produces the smallest norm value in the higher-order sensitivities. There is approximately a two-orders of magnitude difference between the largest and smallest norm values of the higher-order sensitivities. There are some discrepancies between the norm values of the higher-order sensitivities calculated with the finite-difference and the proposed approximate methods. However, these discrepancies are moderate, such that the approximate method can be used to estimate the order of magnitude of the higher-order sensitivities.

To investigate the effects of different shape functions on the performance of the proposed approximate method, a similar evaluation is performed using the Wagner functions, which are quite different in shape from those of the Hicks–Henne functions. Although the Hicks–Henne functions have only one maximum, the Wagner functions have one global minimum or maximum; but they may also have several local minimums and maximums. The Wagner functions may be more effective in simultaneously modifying the different locations of the airfoil geometry. In Fig. 4b, the values of the higher-order sensitivities of the Wagner functions are shown with the $\|.\|_2$ norm. The largest and smallest norm values of the higher-order sensitivities are produced by the fifth and ninth design variables, respectively, and there is a two-orders of magnitude difference between them. The results indicate that the proposed approximate method is successful in estimating the order of the higher-order sensitivities. Because the approximate method does not require any flow analysis, the CPU time needed to evaluate higher-order sensitivities is almost negligible. In both the Hicks–Henne and Wagner functions, the ratio of CPU time of the proposed approximate method to that of the finite-difference method is on the order of 0.001. The variations of the norm values of the second- and third-order sensitivities with respect to the design variables show a similar behavior. The norm values of the third-order sensitivities resemble the amplification of the norm values of the second-order sensitivities. Although the norm values of the higher-order sensitivities change with respect to the norm types, the variation of the norm values with respect to the design variables shows similar behavior for all three norms.

In the finite-difference calculation of the third-order sensitivities with the Navier–Stokes equations, some difficulties are encountered. For example, the main obstacle is finding the appropriate finite-difference step size for the third-order sensitivity calculation. These difficulties may be related to the nondifferentiable terms in artificial dissipation and turbulence modeling. The third-order sensitivities can be estimated with the proposed approximate method without any problems. Because these sensitivities with the finite-difference method are not available, the comparison of the finite-difference and the proposed approximate method is performed only with the second-order sensitivities. The second-order sensitivities calculated with the Wagner functions are shown in Fig. 4c. In Navier–Stokes calculations, the largest and the smallest norm values of the second-order sensitivities are calculated with the same design variables used in the Euler equations. In the results calculated with the Navier–Stokes equations, there are approximately two-orders of magnitude differences between the largest and smallest norm value of the second sensitivities. The ratio of the CPU time of the approximate method to that of finite-difference method being on the order of 0.0001 is almost negligible. For the same type of shape function, the variation of the norm values of the second-order sensitivities with respect to the design variables appears to be similar. However, the magnitude of the norm values of the second-order sensitivities is larger in the Navier–Stokes computations.

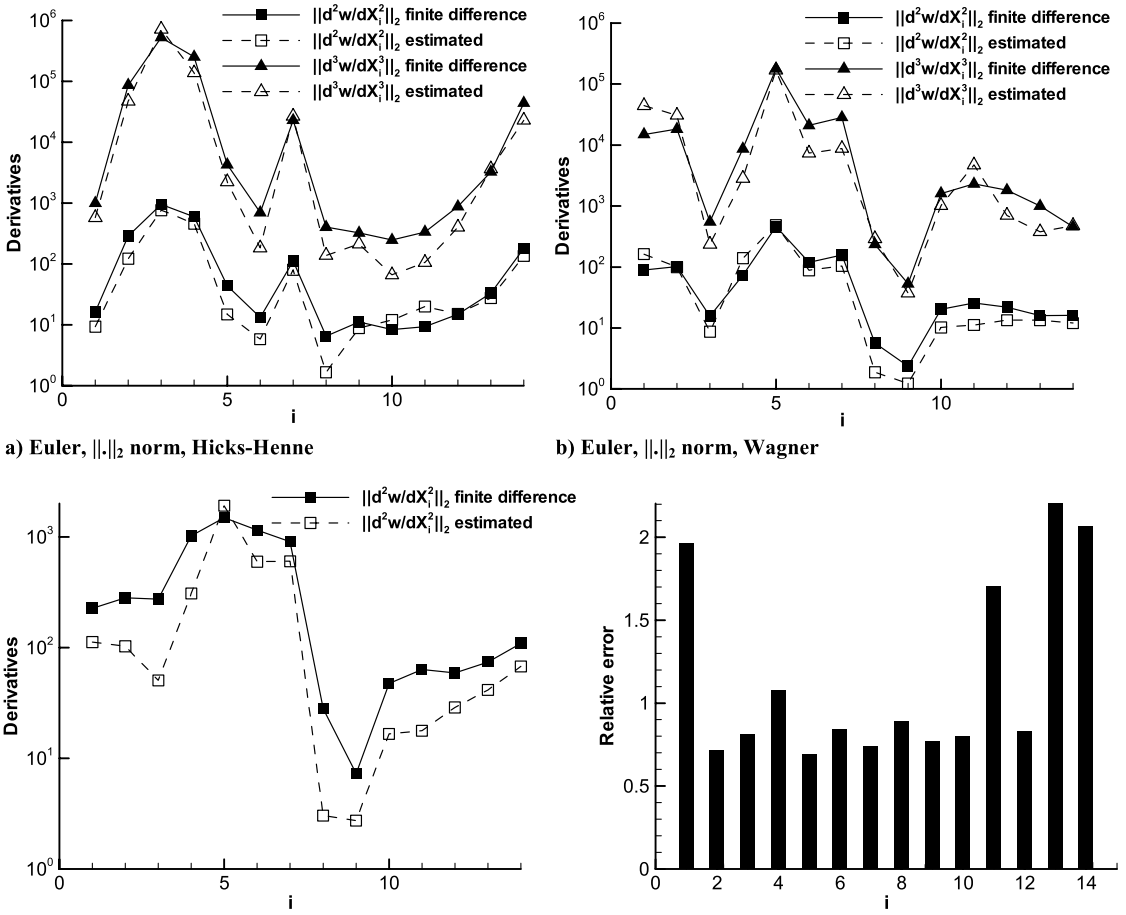


Fig. 4 Second- and third-order sensitivity estimation.

To evaluate the overall performance of the proposed approximate method, the relative errors in higher-order sensitivities are calculated. For each design variable, these calculations are performed for eighteen different cases. The relative errors in the second-order sensitivities are calculated using the Euler and the Navier–Stokes equations for three different norm types and two different shape functions. In the calculation of the relative errors in the third-order sensitivities, only the Euler equations and the same shape functions and norm types are used. The relative errors in the second- and third-order sensitivities are calculated using the following relation:

$$\text{Relative Error} = \frac{\left| \left\| \frac{\mathrm{d}^n w}{\mathrm{d}X_i^n} \right\|_{\text{approximate}} - \left\| \frac{\mathrm{d}^n w}{\mathrm{d}X_i^n} \right\|_{\text{finite-difference}} \right|}{\left\| \frac{\mathrm{d}^n w}{\mathrm{d}X_i^n} \right\|_{\text{finite-difference}}} \qquad (58)$$

$n = 2, 3$

For each design variable, the largest value of the relative error is selected among the second- and third-order sensitivities. Figure 4d shows the largest value of relative error for each design variable, namely, the worst cases. As seen from the figure, the largest values of the relative error are less than one in most of the design variables. In this study, the largest value of the relative error is observed in the evaluation of the second-order derivative with the Euler equations. The thirteenth design variable in the Wagner function produces a relative error value of approximately two, and this error is measured with the $\|.\|_\infty$ norm. Although for some of the design variables, the magnitude of the relative error may be large, in general, the proposed method is accurate enough to estimate the order of magnitude of the higher-order sensitivities, which the optimum step-size calculations may need the most.

C. Finite-Difference Sensitivity Calculations

The variation of error in the finite-difference sensitivities with respect to the convergence error and step size is examined. The error in the finite-difference sensitivities is defined as the difference between the sensitivities evaluated with the finite-difference and analytical methods. The analytical and finite-difference sensitivities are calculated for the same discretized Euler equations. The finite-difference sensitivities are evaluated with different step sizes, ΔX, and convergence error levels, ε. Because the perturbation magnitude is small, in order to reduce the number of iterations, the flow field in perturbed geometry is initialized with the converged solution of base geometry. However, if the finite-difference step size is too small for a given convergence error level, the iterations in the perturbed geometry stop after a few iterations. This may cause oscillation in the norm values of sensitivity error. To prevent the oscillations, for each norm value of convergence error, a minimum finite-difference step size is determined. For a given step size, if the specified value of convergence error is reached within three iterations, the smaller step sizes are not used. The norm value of error in sensitivities with respect to the step size is plotted for the constant norm value of the convergence error. The averaging effects of norms may cause smooth variations in sensitivity errors even for the small step sizes. The same type of norm is used for both the finite-difference and convergence error calculations. For each convergence error level, the actual values of the optimum step size and corresponding error can be determined from these plots. The optimum step size and the corresponding error can also be estimated from the relations derived in Sec. II.A. The estimated results are marked in the figures with the symbols of filled circles or squares, depending on whether the higher-order sensitivities are calculated with finite-difference or the proposed approximate methods, respectively. The values of the optimum step size and corresponding error are estimated using the norm types

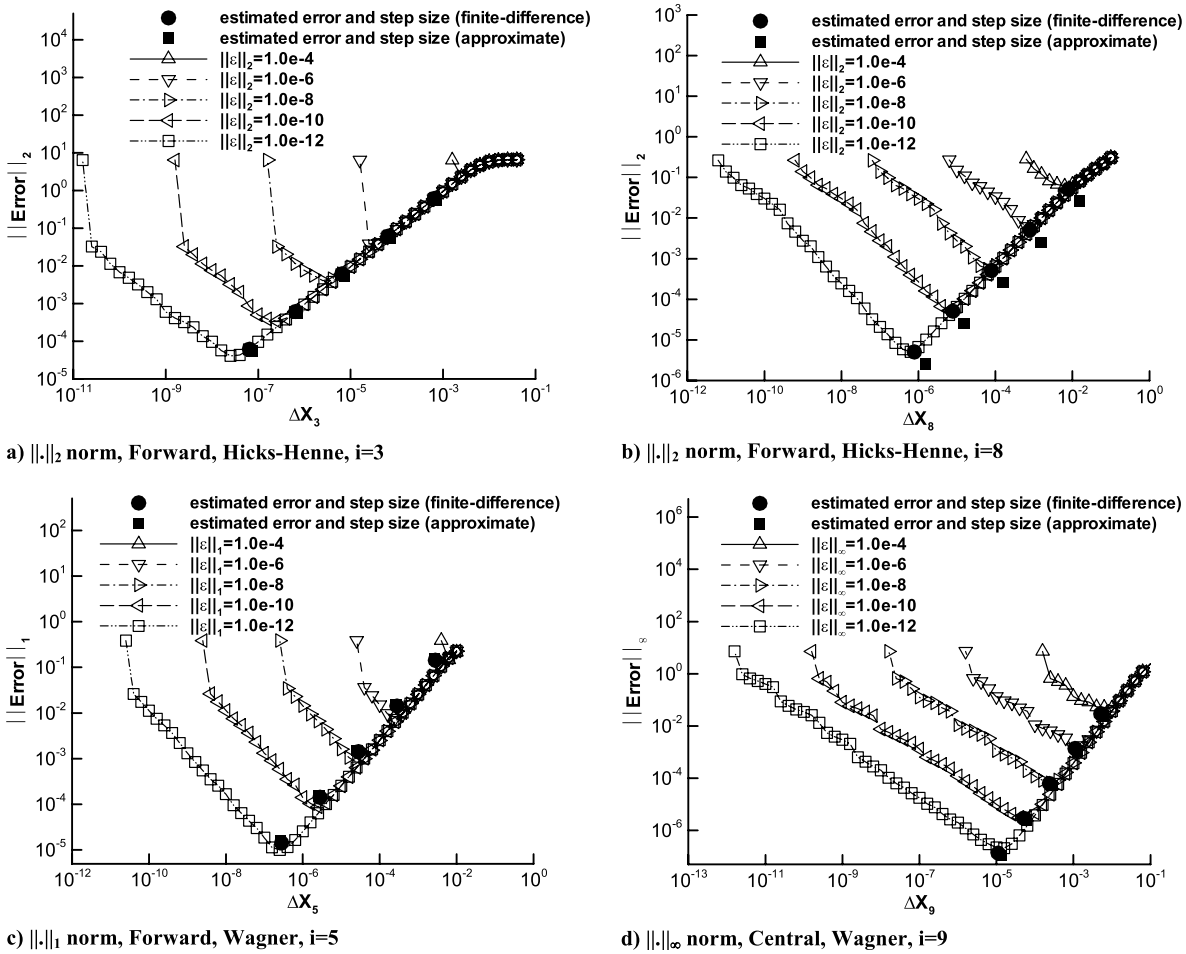
defined in Eq. (56). The optimum step sizes are inversely proportional to the norm values of the higher-order sensitivities. To reduce the number of figures, the results are only shown for the design variables that have the minimum and the maximum norm values in the higher-order sensitivities.

First, the Hicks–Henne functions are used in the accuracy evaluation of the forward-difference scheme. In this scheme, the optimum step size is inversely proportional to the square root of the second-order sensitivities. In the Hicks–Henne functions, the third and eighth design variables produce the largest and smallest norm values in the second-order sensitivities, respectively. Hence, the error in the finite-difference sensitivities is only evaluated for the third and eighth design variables. Figures 5a and 5b show the $\|.\|_2$ norm values of the sensitivity error for the third and eighth design variables, respectively. The results verify that Eqs. (12) and (13) can accurately predict both the optimum step size and the norm values of error in sensitivities. In all three norms, the optimum step sizes of the eighth design variable are almost one order magnitude larger than those of the third design variable. This is expected because the norm values of the second-order sensitivities of the eighth design variable are almost two orders of magnitude smaller than those of the third design variable. For the same reason, the error calculated with the eighth design variable is one order of magnitude smaller than the error calculated with the third design variable. In general, the values of the second-order sensitivities calculated with the finite-difference and the proposed approximate methods give similar results for the optimum step size and the resulting error. Larger discrepancies are observed in the eighth design variable, because the second-order sensitivities calculated with the approximate method have a larger relative error in the eighth design variable.

Next, the accuracy of the forward- and central-difference schemes is evaluated using the Wagner functions. The values of finite-difference and analytical sensitivities are only compared for the fifth and ninth design variables, which produce the largest and smallest norm values, respectively, in the higher-order sensitivities. In the calculation of the forward-difference sensitivities, the fifth design variable is used. In the forward-difference scheme, the optimum step size and the corresponding error are evaluated using the $\|.\|_1$ norm values of convergence error and the second-order sensitivities. Figure 5c shows that the optimum forward-difference step size and the corresponding error are accurately predicted, as in the Hicks–Henne functions. In the central-difference calculations, the ninth design variable is used. For a given convergence error level, the optimum step size in the central-difference scheme is inversely proportional to the cube root of the third-order sensitivities. In the central-difference schemes, the results are demonstrated using the $\|.\|_\infty$ norm values of convergence error and third-order sensitivities. Figure 5d shows that the optimum step sizes and the corresponding sensitivity errors are accurately estimated with Eqs. (14) and (15). As seen from the figures, in both the forward- and central-difference schemes, the values of the optimum step size and error predicted with the proposed approximate method are very close to those predicted with the finite-difference method.

D. Inverse Design Optimization

The effects of the accuracy of the sensitivities on the convergence of inverse design, calculated with both the analytical and finite-difference methods, are examined. A least-squares optimization method is used to minimize the pressure discrepancies between the designed and target airfoils. Euler or Navier–Stokes equations are used for the flow analysis. As in previous calculations, the grid sizes are 129×33 and 257×65 in the design with Euler and Navier–Stokes equations, respectively. A total of fourteen design variables



a) $\|.\|_2$ norm, Forward, Hicks-Henne, i=3

b) $\|.\|_2$ norm, Forward, Hicks-Henne, i=8

c) $\|.\|_1$ norm, Forward, Wagner, i=5

d) $\|.\|_\infty$ norm, Central, Wagner, i=9

Fig. 5 Sensitivity error in the Euler equations.

are used to modify the airfoil geometry. The detailed analysis of the inverse design method is given in Eyi and Lee [30]. The following objective function is defined as the surface pressure discrepancies between the target and designed airfoils:

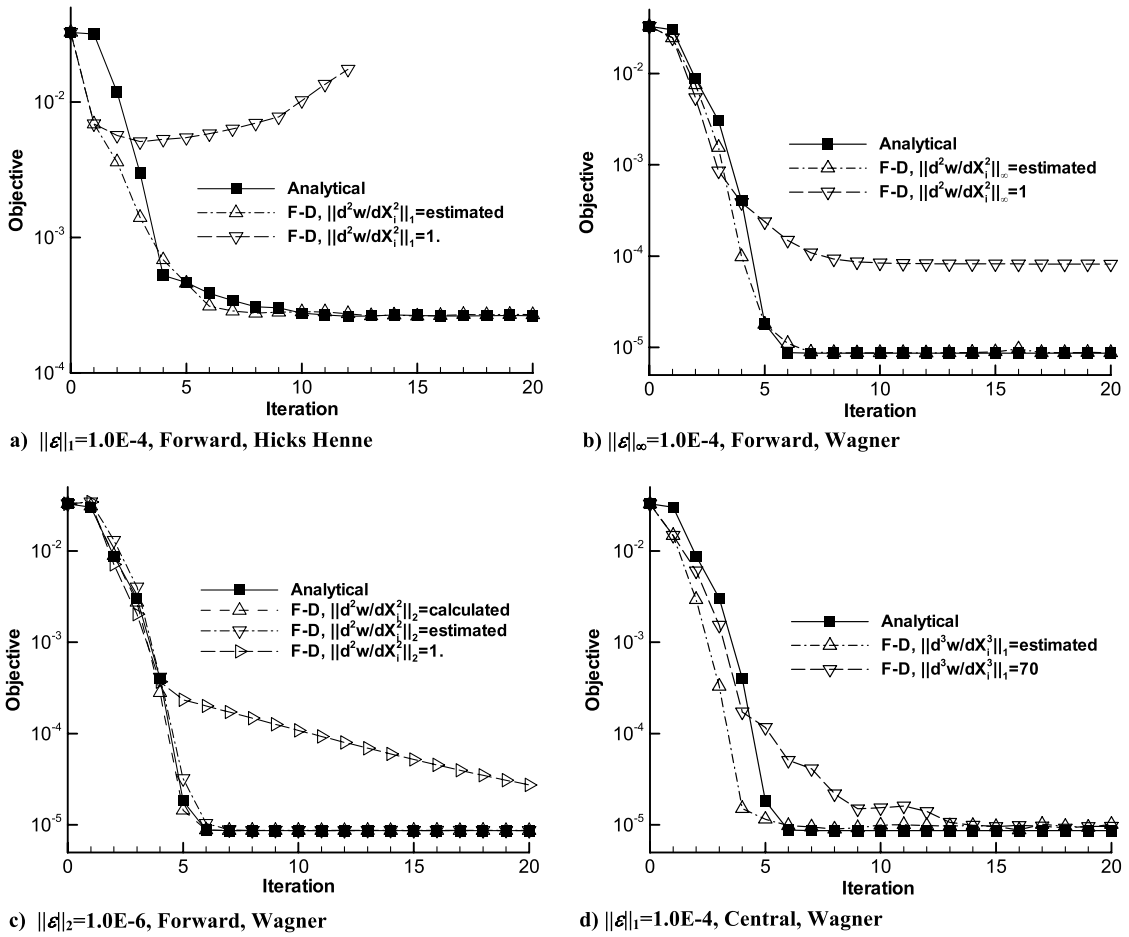
$$\psi = \int_{\Gamma} (p_t - p_d)^2 \, d\Gamma \tag{59}$$

In Eq. (59), p_t and p_d are the surface pressure distributions of the target and the designed airfoils, respectively. Γ is the arc length along the airfoil surface.

The convergences of this objective function evaluated with analytical and finite-difference sensitivities are compared. The forward- and central-difference schemes are used to evaluate the finite-difference sensitivities. In both schemes, sensitivities are evaluated using the optimum finite-difference step size. In the estimation of the optimum step size, the norm values of the convergence error and higher-order sensitivities are evaluated using the proposed methods. In addition to the proposed approximate method, the norm values of higher-order sensitivities are evaluated with the finite-difference method and are assumed to be unity. The results are demonstrated using different norm types. The effects of shape functions on the convergence of design optimization are studied using the Hicks–Henne and Wagner functions.

In the design optimization with the Euler equations, first, the forward-difference scheme is used for the sensitivity calculations. The convergences of the objective function with the Hicks–Henne and Wagner functions for the error level of 10^{-4} are shown in Figs. 6a and 6b, respectively. In the design with the Wagner functions, a larger reduction in the objective function is achieved amounting to more than three orders of magnitude from the initial values. This reduction is approximately two orders of magnitude larger than that obtained with the Hicks–Henne functions. All the finite-difference sensitivities are calculated using the optimum step size. In design with Hicks–Henne functions, the optimum step size is evaluated using the $\|.\|_1$ norm values of the convergence error and higher-order sensitivities. In design with the Wagner functions, the $\|.\|_\infty$ norm values are used. As presented in Eq. (12), the optimum step size in the forward-difference scheme can be evaluated as a function of the norm values of the convergence error and the second-order sensitivities. In both the Hicks–Henne and Wagner functions, the optimum step-size calculation algorithm fails if the second-order sensitivities are calculated with the finite-difference method. For the convergence error level of 10^{-4}, satisfying the relative condition error in a given range may produce very large step sizes for the second-order sensitivity calculations. The geometry change with these step sizes is so large that the iterative solution of the Euler equations diverges. If the norm values of the second-order sensitivities are assumed to be one, designs evaluated with an optimum finite-difference step size do not converge in the Hicks–Henne functions and partially converge in the Wagner functions. If the optimum step size is calculated with approximate second-order sensitivities, larger reductions in the objective function are achieved with the finite-difference sensitivities. In the designs with both the Hicks–Henne and Wagner functions, almost the same amount of reduction in the objective function is achieved with both the analytical and finite-difference sensitivities when the optimum step size is evaluated with the approximate second-order sensitivities given in Eq. (46).

In Fig. 6c, the convergences of the objective function with Wagner functions are shown for a convergence error level of 10^{-6}. The $\|.\|_2$ norm values of the convergence error and higher-order sensitivities are used in the estimation of the optimum step size. The convergence behaviors of the objective function with the analytical and finite-difference sensitivities are almost the same if the second-order sensitivities are evaluated with the finite-difference and the proposed approximate methods. The convergence of the objective function is



a) $||\varepsilon||_1$=1.0E-4, Forward, Hicks Henne

b) $||\varepsilon||_\infty$=1.0E-4, Forward, Wagner

c) $||\varepsilon||_2$=1.0E-6, Forward, Wagner

d) $||\varepsilon||_1$=1.0E-4, Central, Wagner

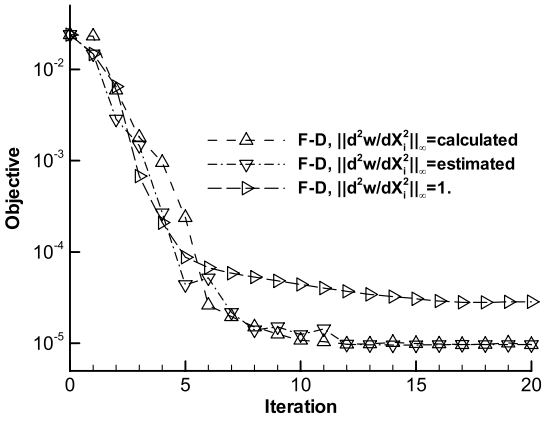
Fig. 6 Convergence of an inverse design with the Euler equations.

delayed if the second-order sensitivities are assumed to be one. For a convergence error level of 10^{-8}, the error in the finite-difference sensitivities is so small that the accuracy of the optimum step size does not significantly affect the convergence of the objective function. Hence, the advantages of using the proposed methods in design optimization may be more significant at high convergence error levels. Consequently, in this study the convergence behaviors of the objective function are only shown for convergence error levels of 10^{-4} and 10^{-6}.

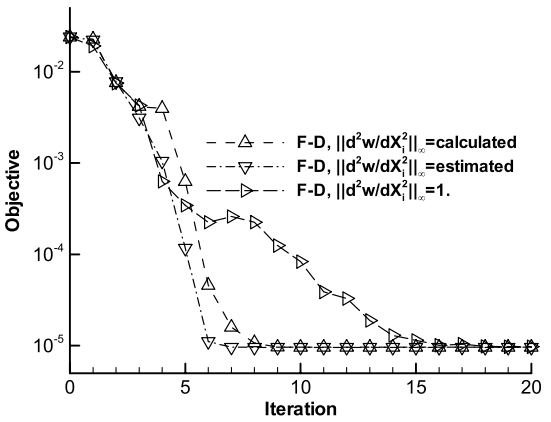
In design optimization with the Euler equations, the central-difference scheme is used for sensitivity calculation and only the Wagner functions are used to modify airfoil geometry. As shown in Eq. (14), the optimum step size in the central-difference scheme can be formulated as a function of the norm values of the convergence error and the third-order sensitivities. Figure 6d shows the convergence of the objective function with the central-difference scheme. For error levels of 10^{-4}, the optimum step-size calculation algorithm fails if the third-order sensitivities are calculated with the finite-difference method. For this error level, the finite-difference step size for the third-order sensitivity calculation is so large that the iterative solution of the Euler equations diverges. The convergence of the objective function also fails if the optimum step size is calculated with an assumption that the norm values of the third-order sensitivities are unity. The norm values of the third-order sensitivities should be increased to 70 to achieve a convergence. Although assuming values of higher-order sensitivities to be unity is a common approach in optimum step-size calculations, the results presented here show the deficiency of this approach. If the optimum step size is calculated with the approximate third-order sensitivities given in Eq. (47), a very fast convergence in objective function can be achieved.

Design optimization is also performed using the Navier–Stokes equations. Because the analytical sensitivities are not available for the Navier–Stokes equations, the convergence of the objective function is only compared with the finite-difference sensitivities. The convergences of the objective function are compared using three different optimum step-size calculation methods. The optimum step sizes are evaluated using the $\|.\|_\infty$ norm values of convergence error and higher-order sensitivities. Figure 7a shows the convergence of the objective function with a convergence error level of 10^{-4}. The value of the objective function converges to the same level if the second-order sensitivities are calculated by the finite-difference and the proposed approximate methods. The convergence of the objective function stalls if the optimum step size is calculated with the assumption that the norm value of the second-order sensitivities is unity. Figure 7b shows the convergence of the objective function with a convergence error level of 10^{-6}. The fastest convergence in objective function is achieved if the second-order sensitivities are evaluated using the proposed approximate method. The convergence of the objective function is delayed if the optimum step sizes are calculated with the assumption that the norm value of the second-order sensitivities is one. In Fig. 7c, the pressure distribution of the initial, target, and designed airfoils are compared. As seen from the figure, the pressure distributions of the designed and target airfoils are accurately matched. In this figure, the accuracy of the Navier–Stokes solution is also verified with experimental data. Figure 7d shows that the geometries of the designed and target airfoils are in good agreement.

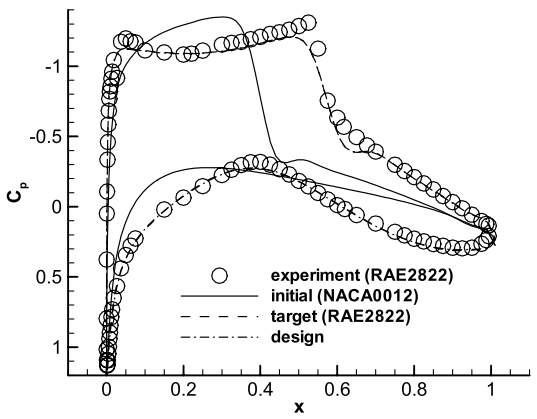
The CPU times of twenty design iterations with different optimum step-size calculations are compared. Table 1 shows the results for three different $\|.\|_\infty$ norm values of the convergence error in the flow and sensitivity analyses: 10^{-4}, 10^{-6}, and 10^{-8}. The shortest CPU time is achieved if the second-order sensitivities in the optimum step-size calculation are estimated with the proposed approximate method. If the second-order sensitivities are calculated with the finite-difference method, the CPU times of design are the longest;



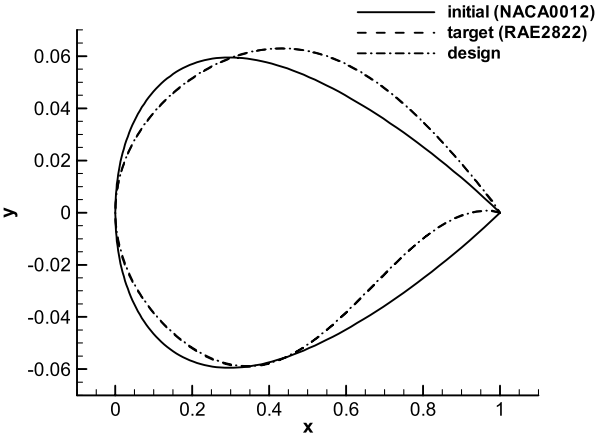
a) $\|\varepsilon\|_\infty$=1.0E-4, Forward, Wagner



b) $\|\varepsilon\|_\infty$=1.0E-6, Forward, Wagner



c) Comparison of surface pressure distributions



d) Comparison of airfoil geometries

Fig. 7 Convergence of an inverse design with the Navier–Stokes equations.

Table 1 CPU time of inverse design with Euler equations and 14 Wagner functions

Method		$\|\varepsilon\|_\infty = 10^{-4}$	$\|\varepsilon\|_\infty = 10^{-6}$	$\|\varepsilon\|_\infty = 10^{-8}$
Analytical		2 min 57 s	4 min 33 s	7 min 50 s
Forward-difference	F-D, $\|d^2w/dX_i^2\|_\infty$ = calculated	N/A	12 min 14 s	16 min 8 s
	F-D, $\|d^2w/dX_i^2\|_\infty$ = estimated	1 min 21 s	1 min 50 s	2 min 41 s
	F-D, $\|d^2w/dX_i^2\|_\infty = 1$	2 min 12 s	3 min 10 s	3 min 35 s
Central-difference	F-D, $\|d^3w/dX_i^3\|_\infty$ = calculated	N/A[a]	N/A[a]	58 min 45 s
	F-D, $\|d^3w/dX_i^3\|_\infty$ = estimated	2 min 51 s	4 min 25 s	6 min 52 s
	F-D, $\|d^3w/dX_i^3\|_\infty = 70$	4 min 20 s	5 min 45 s	8 min 34 s

[a]N/A, not available.

Table 2 CPU time of inverse design with Navier–Stokes equations and fourteen Wagner functions

Method		$\|\varepsilon\|_\infty = 10^{-4}$	$\|\varepsilon\|_\infty = 10^{-6}$	$\|\varepsilon\|_\infty = 10^{-8}$
Forward-difference	F-D, $\|d^2w/dX_i^2\|_\infty$ = calculated	14 h 47 min 58 s	23 h 38 min 11 s	51 h 0 min 10 s
	F-D, $\|d^2w/dX_i^2\|_\infty$ = estimated	1 h 46 min 56 s	2 h 37 min 18 s	6 h 23 min 1 s
	F-D, $\|d^2w/dX_i^2\|_\infty = 1$	6 h 34 min 43 s	11 h 46 min 45 s	19 h 19 min 47 s

almost 10 times greater compared to those calculated with the proposed approximate method. The designs with analytical sensitivities use the second longest CPU time. This may be due to the use of a direct differentiation method in the evaluation of analytical sensitivities. Calculating analytical sensitivities with an adjoint method may significantly reduce the CPU time of design. Although the CPU time of design in the direct differentiation method is proportional to the number of design variables, in the adjoint method it is almost independent of the number of design variables. If the second-order sensitivities in optimum step-size calculation are assumed to be one, the CPU time for the design increases compared to that evaluated with the proposed approximate method. In both methods, the CPU times used to evaluate the optimum step size are small. However, using the proposed approximate method provides more accurate second-order sensitivities and they are much greater than one. According to Eq. (12), the larger values of the second-order sensitivities result in a smaller step size. Hence, the finite-difference sensitivity calculation with smaller step size reduces the number of flow solution iterations in the perturbed geometry. As seen in Table 1, the CPU times of design with the central-difference sensitivity calculations showed similar trends. In Table 2, the CPU times of the design using Navier–Stokes equations are compared with the different optimum step-size calculations. The CPU times are much longer in designs with Navier–Stokes equations compared to those calculated with the Euler equations. Using a larger grid size and requiring a larger number of iteration to reach the specified convergence level may be the primary reasons that designs employing Navier–Stokes equations use longer CPU time. Similarly, the shortest CPU time is achieved when the second-order sensitivities are calculated with the proposed approximate method. In Tables 1 and 2, the CPU times are given for fourteen design variables. If the number of design variables increases, the CPU times may also increase.

IV. Conclusions

In this study, two new methods were developed to estimate the convergence error and higher-order sensitivities. In the first method, the convergence error is estimated in iteratively solved problems. The method was based on the eigenvalue analysis of linear systems, the accuracy of which was verified for both linear and nonlinear problems. The results showed that the convergence error can be accurately estimated with the developed method. The residual itself, on the other hand, was not considered to be a reliable parameter to predict the convergence error. In the second method, the higher-order sensitivities were evaluated by differentiating the approximately constructed differential equation with respect to the design variables. With this method, the norm values of the second- and the third-order sensitivities are estimated efficiently and accurately. The methods developed for the convergence error and higher-order sensitivity estimation were successfully implemented to calculate the optimum step size in the forward- and central-difference sensitivity evaluations. The norm values of the total error were minimized with respect to the finite-difference step size. The optimum step size was determined as a function of the norm values of both the convergence error and higher-order sensitivities. To confirm whether these methods serve their functionality, an inverse design optimization was performed. In the calculation of the optimum step size, the developed convergence error estimation method was used with different higher-order sensitivity estimation methods. Approximating the norm values of higher-order sensitivities as unity may slow down or stop the convergence of the objective function. The main difficulty in calculating the higher-order sensitivities with the finite-difference method is in the estimation of the appropriate finite-difference step size. Another drawback of this method is the significant increase in design time. On the other hand, the proposed approximate method reduces the CPU time of design and provides a robust and fast convergence in the objective function. The type of differential equation used in the approximate method may be important in the estimation of higher-order sensitivities. To improve the accuracy of the proposed approximate method, further research may be needed on how to select a more appropriate differential equation. Results showed that the advantages of using the proposed methods in design optimization may be more significant at high convergence error levels.

Acknowledgment

This research is supported by The Scientific and Technological Research Council of Turkey (TUBITAK).

References

[1] Reuther, J. J., Jameson, A., Alonso, J. J., Rimlinger, M. J., and Saunders, D., "Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers, Part 1," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 51–60. doi:10.2514/2.2413

[2] Hicken, J., and Zingg, D., "Aerodynamic Optimization Algorithm with Integrated Geometry Parameterization and Mesh Movement," *AIAA Journal*, Vol. 48, No. 2, 2010, pp. 400–413. doi:10.2514/1.44033

[3] Ezertas, A., and Eyi, S., "Performances of Numerical and Analytical Jacobians in Flow and Sensitivity Analysis," AIAA Paper 2009-4140, June 2009.

[4] Hou, G. J.-W., Taylor, A. C., III, and Korivi, V. M., "Discrete Shape Sensitivity Equations for Aerodynamic Problems," *International Journal for Numerical Methods in Engineering*, Vol. 37, No. 13, 1994, pp. 2251–2266. doi:10.1002/nme.1620371307

[5] Nadarajah, S., and Jameson, A., "A Comparison of the Continuous and

Discrete Adjoint Approach to Automatic Aerodynamic Optimization," AIAA Paper 2000-0667, 2000.

[6] Muller, J.-D., and Cusdin, P., "On the Performance of Discrete Adjoint CFD Codes Using Automatic Differentiation," *International Journal for Numerical Methods in Fluids*, Vol. 47, Nos. 8–9, March 2005, pp. 939–945. doi:10.1002/fld.885

[7] Squire, W., and Trapp, G., "Using Complex Variables to Estimate Derivatives of Real Functions," *SIAM Review*, Vol. 40, No. 1, 1998, pp. 110–112. doi:10.1137/S003614459631241X

[8] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., "The Complex-Step Derivative Approximation," *ACM Transactions on Mathematical Software*," Vol. 29, No. 3, 2003, pp. 245–262. doi:10.1145/838250.838251

[9] Gill, P. E., Murray, W., and Wright, M. H., *Practical Optimization*, Academic Press, London, 1992, Chaps. 4, 8.

[10] Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., "Computing Forward-Difference Intervals for Numerical Optimization," *SIAM Journal on Scientific and Statistical Computing*, Vol. 4, No. 2, 1983, pp. 310–321. doi:10.1137/0904025

[11] Iott, J., Haftka, R. T., and Adelman, H. M., "Selecting Step Sizes in Sensitivity Derivatives by Finite-Differences," NASA TM-86382, 1985.

[12] Haftka, R. T., "Sensitivity Calculations for Iteratively Solved Problems," *International Journal for Numerical Methods in Engineering*, Vol. 21, No. 8, 1985, pp. 1535–1546. doi:10.1002/nme.1620210813

[13] Eyi, S., and Lee, K. D., "Efficiency Improvement in Sensitivity Evaluation in Aerodynamic Shape Optimization," AIAA Paper 1996-2506, 1996.

[14] Barton, R. R., "Computing Forward-Difference Derivatives in Engineering Optimization," *Engineering optimization*, Vol. 20, No. 3, 1992, pp. 205–224. doi:10.1080/03052159208941281

[15] Ferziger, J. H., and Peric, M., "Further Discussion of Numerical Errors in CFD," *International Journal for Numerical Methods in Fluids*, Vol. 23, No. 12, 1996, pp. 1263–1274. doi:10.1002/(SICI)1097-0363(19961230)23:12<1263::AID-FLD478>3.0.CO;2-V

[16] Ferziger, J. H., and Peric, M., *Computational Methods for Fluid Dynamics*, Springer–Verlag, Berlin, 2002, Chap. 5.

[17] Bergstrom, J., and Gebart, R., "Estimation of Numerical Accuracy for the Flow Field in a Draft Tube,"*International Journal of Numerical Methods for Heat and Fluid Flow*, Vol. 9, No. 4, 1999, pp. 472–486. doi:10.1108/09615539910266620

[18] Roy, C. J., McWherter-Payne, M. A., and Oberkampf, W. L., "Verification and Validation for Laminar Hypersonic Flowfields, Part 1: Verification,"*AIAA Journal*, Vol. 41, No. 10, 2003, pp. 1934–1943. doi:10.2514/2.1909

[19] Alekseev, A., "An Adjoint-Based A Posteriori Estimation of Iterative Convergence Error," *Computers and Mathematics with Applications*, Vol. 52, Nos. 8–9, Oct. 2006, pp. 1205–1212. doi:10.1016/j.camwa.2006.11.002

[20] Brezinski, C., "Error Estimates for the Solution of Linear Systems," *SIAM Journal on Scientific Computing*,Vol. 21, No. 2, Sept.–Oct. 1999, pp. 764–781. doi:10.1137/S1064827597328510

[21] Schoofs, A. J. G., van Houten, M. H., Etman, L. F. P., and van Capmen, D. H., "Global and Mid-Range Function Approximation for Engineering Optimization," *Mathematical Methods of Operations Research*, Vol. 46, No. 3, 1997, pp. 335–359. doi:10.1007/BF01194860

[22] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., "Surrogate-based Analysis and Optimization," *Progress in Aerospace Sciences*, Vol. 41, No. 1, Jan. 2005, pp. 1–28. doi:10.1016/j.paerosci.2005.02.001

[23] Yew, S. O., Prasanth, B. N., and Andrew, J. K., "Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling," *AIAA Journal*, Vol. 41, No. 4, April 2003, pp. 687–696. doi:10.2514/2.1999

[24] Ramsay, J. O., and Silverman, B. W., *Functional Data Analysis*, 2nd ed., Springer–Verlag, New York, 2005, Chap. 19.

[25] Poyton, A. A., Varziri, M. S., McAuley, K. B., McLellan, P. J., and Ramsay, J. O., "Parameter Estimation in Continuous-time dynamic Models Using Principal Differential Analysis," *Computers and Chemical Engineering*, Vol. 30, No. 4, Feb. 2006, pp. 698–708. doi:10.1016/j.compchemeng.2005.11.008

[26] Bowman, J. M., and Xie, T., "On Using Potential, Gradient and Hessian data in the Least-squares Fits of Potential: Application and Tests for H_2O," *Journal of Chemical Physics*, Vol. 117, No. 23, 2002, pp. 10487–10492. doi:10.1063/1.1520140

[27] Slater, J. W., Dudek, J. C., and Tatum, K. E., "The NPARC Alliance Verification and Validation Archive," NASA TM-209946, 2000.

[28] Wang, N., Chang, S.-H., and Huang, H.-C., "Comparison of Iterative Methods for the Solution of Compressible-Fluid Reynolds Equation," *Journal of Tribology*, Vol. 133, No. 2, 2011, p. 021702. doi:10.1115/1.4003149

[29] Antia, H. M., *Numerical Methods for Scientists and Engineers*, 2nd ed., Birkhäuser Boston, Cambridge, MA, 2002, pp. 100–104.

[30] Eyi, S., and Lee, K. D., "Effects of Sensitivity Derivatives on Aerodynamic Design Optimization," *Inverse Problems in Engineering*, Vol. 2, No. 3, 1996, pp. 213–235. doi:10.1080/174159796088027603

[31] Eyi, S., Hager, J. O., and Lee, K. D., "Airfoil Design Optimization Using The Navier-Stokes Equations," *Journal of Optimization Theory and Applications*, Vol. 83, No. 3, 1994, pp. 447–461. doi:10.1007/BF02207637

R. Haftka
Associate Editor