AIAA JOURNAL
Vol. 36, No. 12, December 1998

Effective Two-Point Function Approximation for Design Optimization

Suqiang Xu* and Ramana V. Grandhi†
Wright State University, Dayton, Ohio 45435

A new, two-point approximation of a function that is very straightforward to build and that provides accurate and stable approximation is presented. Earlier developments of the two-point approximations had either incomplete matches at two data points or needed the solution of additional equations to get all of the parameters. The present two-point approximation is an incomplete second-order Taylor-series expansion in terms of intervening variables; the Hessian matrix has only diagonal elements, and it depends on design variables. The exponent of each intervening design variable and the unknown constant of the second-order terms are evaluated by matching the derivatives and the value of the approximation with the previous data point gradients and the value of the original function, respectively. All of the unknowns are identified in a closed form. Both the function and the gradient of the two-point approximation are equal to those of the original function at two data points. Several examples are given to show the accuracy and efficiency of this method.

Introduction

FUNCTION approximation is one of the most important and active fields of research in structural optimization. Accurate function approximations reduce the repetitive cost of finite element analysis. Further information on this topic can be found in several survey papers.[1–4]

One strategy in developing approximation is to use both the function and the gradient information of two data points. Two-point approximation is usually superior to single-point approximation and even better than a multipoint approximation such as multipoint multivariate Hermite approximation.[5,6] For a two-point approximation, an ideal constraint approximation should have the same nonlinearity as the original constraint, and it should match both the function and the derivatives at two data points. Haftka et al.[7] suggested several methods, which include a modified reciprocal, a two-point projection, and an exponential approximation. The modified reciprocal method can be converted to direct and reciprocal expansion by changing the control parameters. The control parameter is evaluated by matching the derivative at the previous point. This method matches both the function and the derivatives at one point and the function at another point. However, it does not embody the nonlinearity of the original function. Haftka's two-point projection is to approximate the function at the projection point of the current design. This is done by a cubic Hermite polynomial and then is extrapolated to the current point. Numerical tests have shown that the two-point projection method is good when it represents interpolation; however, the improvement in accuracy is marginal when it represents extrapolation. A two-point approximation that shows more generality was proposed by Fadel et al.[8] This is a linear Taylor-series expansion in terms of intervening variables, where the exponent for each design variable is determined by matching the derivative of the approximate function with the previous data point gradients. This approximation does not match the exact and approximate function values at the previous point. The index in this method, however, is simple to calculate. Belegundu et al.[9] developed a two-point posynomial approximation, which uses the least-squares approach to find the approximation parameters. This method uses not only the value of the function and its gradient at the current point but also the value of the function at the second point. However, this does not guarantee the matching of both the function and the gradient. Snyman and Stander[10] used only the previous function in creating a two-point quadratic approximation without intervening variables. Wang and Grandhi[5,6,11] presented several two-point approximations for structural reliability analysis and optimization. Wang and Grandhi[11] used a two-point adaptive nonlinearity approximation (TANA), which is also a linear Taylor-series expansion in terms of the intervening variables. But, unlike the two-point exponential approximation (TPEA) suggested by Fadel et al.,[8] this approach has only one exponential value for all variables and the nonlinearity index is determined by matching the function of the previous design point, which resulted in a transcendental algebraic equation. This approach does not match the derivative of a previous point, and solving the transcendental algebraic equation adds computational cost. To match the function value of the nonexpanding point, Wang and Grandhi[6] added a constant to TPEA and named it TANA-1. However, there is no matching of the function value at the expanding point. Furthermore, Wang and Grandhi[6] expanded the function using an incomplete second-order Taylor-series expansion with respect to intervening variables, with an unknown constant to be determined. This approach was referred to as TANA-2. Matching the previous derivative and function values requires $n+1$ equations, and the exponential and constant values are obtained by solving these equations. In this method, the exact function and derivative values are equal to the approximate function and derivative values, respectively, at the previous and current points. Numerical results[5,6] have shown that this approximation is superior to other methods. A drawback to this method is that additional computational cost is needed to solve $n+1$ coupled equations for each of the active constraints.

The present paper suggests a new two-point function approximation that embodies the nonlinearity of the original function and has exact function and gradients at two data points. It looks like TANA-2, but its computational cost is much lower.

Two-Point Approximations

In the following, $\boldsymbol{X} = (x_1, x_2, \ldots, x_n)^T$ is defined as a vector of design variables. $\boldsymbol{X}_1 = (x_{1,1}, x_{2,1}, \ldots, x_{n,1})^T$, $\boldsymbol{X}_2 = (x_{1,2}, x_{2,2}, \ldots, x_{n,2})^T$ refer to the previous and current data points where the function and gradient information is available.

Previous Two-Point Approximation

To better explain the proposed method, earlier developments are discussed first. TPEA was introduced by Fadel et al.[8] who used the following intermediate variables:

$$y_i = x_i^{p_i}, \qquad i = 1, 2, \ldots, n \tag{1}$$

Received Feb. 5, 1998; revision received Aug. 10, 1998; accepted for publication Aug. 10, 1998. This paper is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

*Research Associate, Department of Mechanical and Materials Engineering. Member AIAA.

†Distinguished Professor, Department of Mechanical and Materials Engineering. Associate Fellow AIAA.

Next the approximate function is specified by expanding function $F(X)$ using the first-order Taylor-series expansion with respect to the intermediate variables:

$$\tilde{F}(X) = F(X_2) + \sum_{i=1}^{n} \left(\frac{x_{i,2}^{1-p_i}}{p_i} \right) \frac{\partial F(X_2)}{\partial x_i} \left(x_i^{p_i} - x_{i,2}^{p_i} \right) \tag{2}$$

where the exponent p_i for each variable is evaluated by matching the derivative of the approximate function with the previous data point gradients; p_i is obtained in a closed-form solution.

A similar approach (TANA) was presented in Ref. 11 in which p_i was chosen as the same for all variables. Matching the function value of the previous data point leads to a transcendental equation. The index $p = p_1 = p_2 \ldots p_i \ldots$ is determined by solving the equation through numerical methods.

An improved two-point approximation (TANA-1) was presented in Refs. 5 and 6:

$$\tilde{F}(X) = F(X_1) + \sum_{i=1}^{n} \frac{\partial F(X_1)}{\partial x_i} \frac{x_{i,1}^{1-p_i}}{p_i} \left(x_i^{p_i} - x_{i,1}^{p_i} \right) + \varepsilon \tag{3}$$

where ε represents the higher-order terms of the Taylor-series expansion. To evaluate p_i and ε, the approximate function and its derivatives are matched with the current point; p_i and ε are calculated in closed form. This approach was also extended by considering the second-order Taylor-series expansion (TANA-2):

$$\tilde{F}(X) = F(X_2) + \sum_{i=1}^{n} \frac{\partial F(X_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_i^{p_i} - x_{i,2}^{p_i} \right) + \frac{1}{2} \varepsilon \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i} \right)^2 \tag{4}$$

in which the Hessian matrix has only diagonal elements of the same value ε. Values of p_i and ε are evaluated by matching the derivative and the approximate function with the previous point X_1. The resulting equations are given here to explain the new method:

$$\frac{\partial F(X_1)}{\partial x_i} = \left(\frac{x_{i,1}}{x_{i,2}} \right)^{p_i - 1} \frac{\partial F(X_2)}{\partial x_i} + \varepsilon \left(x_{i,1}^{p_i} - x_{i,2}^{p_i} \right) x_{i,1}^{p_i - 1} p_i \qquad i = 1, 2, \ldots, n \tag{5}$$

$$F(X_1) = F(X_2) + \sum_{i=1}^{n} \frac{\partial F(X_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i} \right) + \frac{1}{2} \varepsilon \sum_{i=1}^{n} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i} \right)^2 \tag{6}$$

To get p_i and ε, coupled equations (5) and (6) need to be solved.

TANA-2 exhibited the best accuracy among the two-point approximations mentioned earlier and even among multipoint Hermite approximations.[5,6] But, as we observe from Eqs. (5) and (6), a drawback is that $n + 1$ coupled equations need to be solved for each active constraint, which is high in computational cost. Wang and Grandhi[6] used an algorithm as follows to solve Eqs. (5) and (6): First, ε is fixed at a small initial value (0.5). Then you have only one unknown in each ith equation; the numerical iteration for calculating each p_i starts from an initial value (1). When p_i is increased or decreased by a step length (0.1), the error between the exact and the approximation functions and the derivative values at X_1 are calculated. Then ε is increased or decreased by a step length (0.1), p_i, and the differences between the exact and approximate functions and the derivative values at X_1 are recalculated. If these differences are smaller than the initial error, e.g., corresponding to initial ε, the iteration is repeated until the allowable error (0.001) or the limitation of ε is reached, and the optimum combination of ε and p_i is determined. Always, only one nonlinear equation with one unknown is solved iteratively. There is no finite element (FE) analysis involved; calculations are done using a closed-form equation, Eq. (5).

To estimate the computational cost of Eqs. (5) and (6), let us consider a structural optimization with m degrees of freedom, n variables, and l constraints. We assume 10% l active constraints at each approximate subproblem. For each possible combination of ε and p_i, Eqs. (5) and (6) need at least $20n$ floating operations for the preceding solution procedure. If p_i and ε have an average of 10 possible values, the total floating operations to get p_i and ε for all variables and all active constraints will be approximately $10 \times 10 \times 20n \times 10\%\ l$. For FE analysis with m degrees of freedom, most of the computational cost is in solving the equilibrium equation. For an $m \times m$ positive-definite and symmetric matrix, the lower and upper factorization and forward and backsubstitution will take about $m^3/6 + m^2$ floating operations.[12] Solving the equilibrium equation of FE analysis with m degrees of freedom will take much less than $m^3/6 + m^2$. This is because the stiffness matrix is banded.

For example, if $m = n = l = 100$, then $10 \times 10 \times 20n \times 10\%\ l = 2 \times 10^6$, $m^3/6 + m^2 = 1.67 \times 10^5$, which means that Eqs. (5) and (6) will take 12 times as many floating operations as FE analysis. To reduce this computational cost, a new approximation is proposed that has about the same accuracy.

New Two-Point Approximation (TANA-3)

We expand the function with respect to the intervening variable $y_i = x_i^{p_i}$ as in Eq. (4):

$$\tilde{F}(X) = F(X_2) + \sum_{i=1}^{n} \frac{\partial F(X_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_i^{p_i} - x_{i,2}^{p_i} \right) + \frac{1}{2} \varepsilon(X) \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i} \right)^2 \tag{7}$$

We specify

$$\varepsilon(X) = \frac{H}{\sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,1}^{p_i} \right)^2 + \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i} \right)^2} \tag{8}$$

where p_i and H are constants to be determined on the basis of information of the previous point. It is shown that $\varepsilon(X)$ will uncouple p_i and H in the equations to compute them; this will make them easier to calculate. Development of Eq. (8) also is discussed in this section.

Differentiating Eq. (7), we get

$$\frac{\partial \tilde{F}(X)}{\partial x_i} = \left(\frac{x_i}{x_{i,2}} \right)^{p_i - 1} \frac{\partial F(X_2)}{\partial x_i} + E(X) \tag{9}$$

where

$$E(X) = \frac{H p_i x_i^{p_i - 1} \left(x_i^{p_i} - x_{i,2}^{p_i} \right) \sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,1}^{p_j} \right)^2}{\left[\sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,1}^{p_j} \right)^2 + \sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,2}^{p_j} \right)^2 \right]^2} - \frac{H p_i x_i^{p_i - 1} \left(x_i^{p_i} - x_{i,1}^{p_i} \right) \sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,2}^{p_j} \right)^2}{\left[\sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,1}^{p_j} \right)^2 + \sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,2}^{p_j} \right)^2 \right]^2} \tag{10}$$

We notice that $E(X)$ diminishes to zero at two data points. That is,

$$E(X_1) = E(X_2) = 0 \tag{11}$$

So,

$$\frac{\partial \tilde{F}(X_2)}{\partial x_i} = \frac{\partial F(X_2)}{\partial x_i} \tag{12}$$

$$\frac{\partial \tilde{F}(X_1)}{\partial x_i} = \left(\frac{x_{i,1}}{x_{i,2}} \right)^{p_i - 1} \frac{\partial F(X_2)}{\partial x_i} \tag{13}$$

It is easy to show that the second-order term of Eq. (7) is zero at the current point X_2 and is $0.5H$ at the previous point X_1. So,

$$\tilde{F}(X_2) = F(X_2) \tag{14}$$

$$\tilde{F}(X_1) = F(X_2) + \sum_{i=1}^{n} \frac{\partial F(X_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i} \right) + \frac{1}{2} H \tag{15}$$

From Eqs. (12) and (14), we know the function and derivative values of the approximation match with the exact ones at $\boldsymbol{X}_2$. Then $n+1$ uncoupled equations are obtained by matching the derivatives and the value of the approximation with the gradients and the value of the original function at the previous point, $\boldsymbol{X}_1$:

$$\frac{\partial F(\boldsymbol{X}_1)}{\partial x_i} = \left(\frac{x_{i,1}}{x_{i,2}}\right)^{p_i - 1} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i}, \qquad i = 1, 2, \ldots, n \tag{16}$$

$$F(\boldsymbol{X}_1) = F(\boldsymbol{X}_2) + \sum_{i=1}^{n} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right) + \frac{1}{2} H \tag{17}$$

From Eqs. (16) and (17), we get p_i and H in a closed-form solution as follows:

$$p_i = 1 + \ln\left[\frac{\partial F(\boldsymbol{X}_1)}{\partial x_i} \bigg/ \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i}\right] \bigg/ \ln\left[\frac{x_{i,1}}{x_{i,2}}\right] \qquad i = 1, 2, \ldots, n \tag{18}$$

$$H = 2\left[F(\boldsymbol{X}_1) - F(\boldsymbol{X}_2) - \sum_{i=1}^{n} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right)\right] \tag{19}$$

When $[\partial F(\boldsymbol{X}_1)/\partial x_i]/[\partial F(\boldsymbol{X}_2)/\partial x_i]$ or $x_{i,1}/x_{i,2}$ is less than zero or other computational problems occur in Eq. (18), we assign a special value to p_i (1 or -1). For example, consider the optimization iterations near the convergence domain. The variables hardly change, and so $x_{i,1}/x_{i,2}$ is close to 1. In this situation, one can assign $p_i = 1$ or -1.

It is summarized that both the approximate function and derivative values at two data points are equal to the exact counterparts, as Eqs. (12), (14), (16), and (17) indicate; all of the exponents p_i and constant H values can be obtained in a closed-form solution, as Eqs. (18) and (19) show.

We derive Eq. (8) using the following concept: Differentiating Eq. (7), we obtain

$$\frac{\partial \tilde{F}(\boldsymbol{X})}{\partial x_i} = \left(\frac{x_i}{x_{i,2}}\right)^{p_i - 1} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i} + \varepsilon(\boldsymbol{X})\left(x_i^{p_i} - x_{i,2}^{p_i}\right) x_i^{p_i - 1} p_i + \frac{1}{2} \frac{\partial \varepsilon(\boldsymbol{X})}{\partial x_i} \sum_{j=1}^{n} \left(x_j^{p_j} - x_{i,2}^{p_j}\right)^2, \qquad i = 1, 2, \ldots, n \tag{20}$$

Next let the approximate derivatives and function match the exact derivatives and function at $\boldsymbol{X}_1$ to determine p_i and $\varepsilon(\boldsymbol{X})$:

$$\frac{\partial F(\boldsymbol{X}_1)}{\partial x_i} = \left(\frac{x_{i,1}}{x_{i,2}}\right)^{p_i - 1} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i} + \varepsilon(\boldsymbol{X}_1)\left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right) x_{i,1}^{p_i - 1} p_i + \frac{1}{2} \frac{\partial \varepsilon(\boldsymbol{X}_1)}{\partial x_i} \sum_{j=1}^{n} \left(x_{j,1}^{p_i} - x_{j,2}^{p_i}\right)^2, \qquad i = 1, 2, \ldots, n \tag{21}$$

$$F(\boldsymbol{X}_1) = F(\boldsymbol{X}_2) + \sum_{i=1}^{n} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right) + \frac{1}{2} \varepsilon(\boldsymbol{X}_1) \sum_{i=1}^{n} \left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right)^2 \tag{22}$$

We can select many $\varepsilon(\boldsymbol{X})$, including a constant function. If a constant function is selected, Eq. (7) becomes TANA-2. We now select an $\varepsilon(\boldsymbol{X})$ to uncouple p_i and $\varepsilon(\boldsymbol{X})$ in Eq. (21). One approach is to let the summation of the second and third terms in Eq. (21) equal zero. That is,

$$\varepsilon(\boldsymbol{X}_1)\left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right) x_{i,1}^{p_i - 1} p_i + \frac{1}{2} \frac{\partial \varepsilon(\boldsymbol{X}_1)}{\partial x_i} \sum_{j=1}^{n} \left(x_{j,1}^{p_j} - x_{j,2}^{p_j}\right)^2 = 0 \qquad i = 1, 2, \ldots, n \tag{23}$$

This requires that

$$\frac{\partial \varepsilon(\boldsymbol{X}_1)}{\partial x_i} \bigg/ \varepsilon(\boldsymbol{X}_1) = -\frac{2\left(x_{i,1}^{p_i} - x_{i,2}^{p_i}\right) x_{i,1}^{p_i - 1} p_i}{\sum_{j=1}^{n} \left(x_{j,1}^{p_j} - x_{j,2}^{p_j}\right)^2}, \qquad i = 1, 2, \ldots, n \tag{24}$$

Substitute $\boldsymbol{X}$ for $\boldsymbol{X}_1$ in Eq. (24) to obtain

$$\frac{\partial \varepsilon(\boldsymbol{X})}{\partial x_i} \bigg/ \varepsilon(\boldsymbol{X}) = -\frac{2\left(x_i^{p_i} - x_{i,2}^{p_i}\right) x_i^{p_i - 1} p_i}{\sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,2}^{p_j}\right)^2}, \qquad i = 1, 2, \ldots, n \tag{25}$$

$$\frac{\partial \varepsilon(\boldsymbol{X})}{\varepsilon(\boldsymbol{X})} = -\frac{2\left(x_i^{p_i} - x_{i,2}^{p_i}\right) x_i^{p_i - 1} p_i}{\sum_{j=1}^{n} \left(x_j^{p_j} - x_j^{p_j}\right)^2} \partial x, \qquad i = 1, 2, \ldots, n \tag{26}$$

Integrate Eq. (26) to obtain $\varepsilon(\boldsymbol{X})$:

$$\varepsilon(\boldsymbol{X}) = \frac{H}{\sum_{j=1}^{n} \left(x_j^{P_j} - x_{j,2}^{P_j}\right)^2} \tag{27}$$

If Eq. (27) is used in Eq. (7), then Eq. (7) will become TANA-1. The experiences with TANA-1 are known.

Another differential equation that satisfies Eq. (24) is

$$\frac{\partial \varepsilon(\boldsymbol{X})}{\partial x_i} \bigg/ \varepsilon(\boldsymbol{X}) = -\frac{2\left(x_i^{p_i} - x_{i,2}^{p_i}\right) x_i^{p_i - 1} p_i + m\left(x_i^{p_i} - x_{i,1}^{p_i}\right)^{m-1} x_i^{p_i - 1} p_i}{\sum_{j=1}^{n} \left(x_j^{p_j} - x_{j,2}^{p_j}\right)^2 + \left(x_j^{p_j} - x_{j,1}^{p_j}\right)^m} \qquad i = 1, 2, \ldots, n \tag{28}$$

The solution of Eq. (28) is

$$\varepsilon(\boldsymbol{X}) = \frac{H}{\sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,1}^{p_i}\right)^2 + \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i}\right)^m} \tag{29}$$

where m could be any integer except 1. We use $m = 2$ for the symmetry of $(x_i^{p_i} - x_{i,1}^{p_i})$ and $(x_i^{p_i} - x_{i,2}^{p_i})$. Equation (8) was obtained in this manner.

Comparison of TANA-2 with TANA-3

Comparing TANA-2 with TANA-3, we see that both TANA-2 and TANA-3 have the exact derivative and function values at two data points and that both embody the nonlinearity of the original function. Both are incomplete second-order Taylor-series expansions with respect to the intervening variables. The difference is the remaining diagonal terms of the Hessian matrix, in which TANA-2 has constant diagonal terms, whereas TANA-3 has a function of design variables. One of the most important advantages of TANA-3 over TANA-2 is that analytical solutions can be obtained for Eqs. (18) and (19), whereas only numerical solutions can be obtained for Eqs. (5) and (6).

TANA-2 and TANA-3 can be understood from another view. Let us look at the following expansions of a function:

$$\tilde{F}(\boldsymbol{X}) = F(\boldsymbol{X}_2) + \sum_{i=1}^{n} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i^{p_i}} \left(x_i^{p_i} - x_{i,2}^{p_i}\right) + \frac{1}{2} \sum_{j=1}^{n} \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i}\right) \frac{\partial^2 F(\boldsymbol{X}_2)}{\partial x_i^{p_i} \partial x_j^{p_j}} \left(x_j^{p_j} - x_{j,2}^{p_j}\right) \tag{30}$$

$$\tilde{F}(\boldsymbol{X}) = F(\boldsymbol{X}_2) + \sum_{i=1}^{n} \frac{\partial F(\boldsymbol{X}_2)}{\partial x_i^{p_i}} \left(x_i^{p_i} - x_{i,2}^{p_i}\right) + \frac{1}{2} \sum_{j=1}^{n} \sum_{i=1}^{n} \left(x_i^{p_i} - x_{i,2}^{p_i}\right) \frac{\partial^2 F[\boldsymbol{X}_2 + \zeta(\boldsymbol{X} - \boldsymbol{X}_2)]}{\partial x_i^{p_i} \partial x_j^{p_j}} \left(x_j^{p_j} - x_{j,2}^{p_j}\right) \tag{31}$$

where $\zeta \in [0, 1]$; Eq. (30) is a full second-order Taylor-series expansion. Equation (31) is a first-order Taylor-series expansion plus second-order truncation. The second derivatives in Eq. (30) are constant, whereas the second derivatives in Eq. (31) are variable. It can be considered that ε in TANA-2 tries to simulate $\partial^2 F(\boldsymbol{X}_2)/\partial(x_i^{p_i})^2$ in Eq. (30), whereas $\varepsilon(\boldsymbol{X})$ in TANA-3 tries to simulate $\partial^2 F[\boldsymbol{X}_2 + \zeta(\boldsymbol{X} - \boldsymbol{X}_2)]/\partial(x_i^{p_i})^2$ in Eq. (31). Both TANA-2 and TANA-3 discard nondiagonal terms of the second-order parts and use only the function value and the first-derivative information at two data points.

Numerical Examples

Based on previous analyses, TANA-3 and TANA-2 should exhibit the same or very similar accuracy. The objective of numerical examples is just to prove this point. What readers should keep in mind is that the most important difference between TANA-3 and TANA-2 is the computational cost. This section includes two parts. The first is for simple problems with closed-form solutions, whereas the second part presents applications in structural optimization. To demonstrate the accuracy of both TANA-2 and TANA-3, we also present the results of the linear and reciprocal approximations.

Part 1: Comparisons Using Closed-Form Problems

Two examples are selected to examine the difference between TANA-2 and TANA-3. We use relative error as the criterion in evaluating the accuracy. The relative error is defined as

$$\text{Relative error} = \frac{\text{Exact} - \text{Approximation}}{\text{Exact}} \times 100\% \tag{32}$$

The test points are derived using

$$\boldsymbol{X} = \boldsymbol{X}_2 + \alpha \boldsymbol{D} \tag{33}$$

where $\boldsymbol{X}_2$ is an expanding point, α is a step length, and $\boldsymbol{D}$ is a search direction.

Example 1. This example is used in Ref. 6 to compare the two-point approximations. The constraint function is defined as

$$g(X) = \frac{10}{x_1} + \frac{30}{x_1^3} + \frac{30}{x_2} + \frac{2}{x_2^3} + \frac{25}{x_3} + \frac{108}{x_3^3} + \frac{40}{x_4} + \frac{47}{x_4^3} - 1.0 \tag{34}$$

All of the approximations are expanded at $\boldsymbol{X}_2(1.2, 1.2, 1.2, 1.2)$. The relative errors of four methods are plotted in Figs. 1a, 1b, 1c, and 1d for $\boldsymbol{D}_1(1, 1, 1, 1)$, $\boldsymbol{D}_2(-1, 1, -1, 1)$, $\boldsymbol{D}_3(1, 0, 1, 0)$, and $\boldsymbol{D}_4(0, 1, 0, 1)$, respectively. This shows that both TANA-2 and

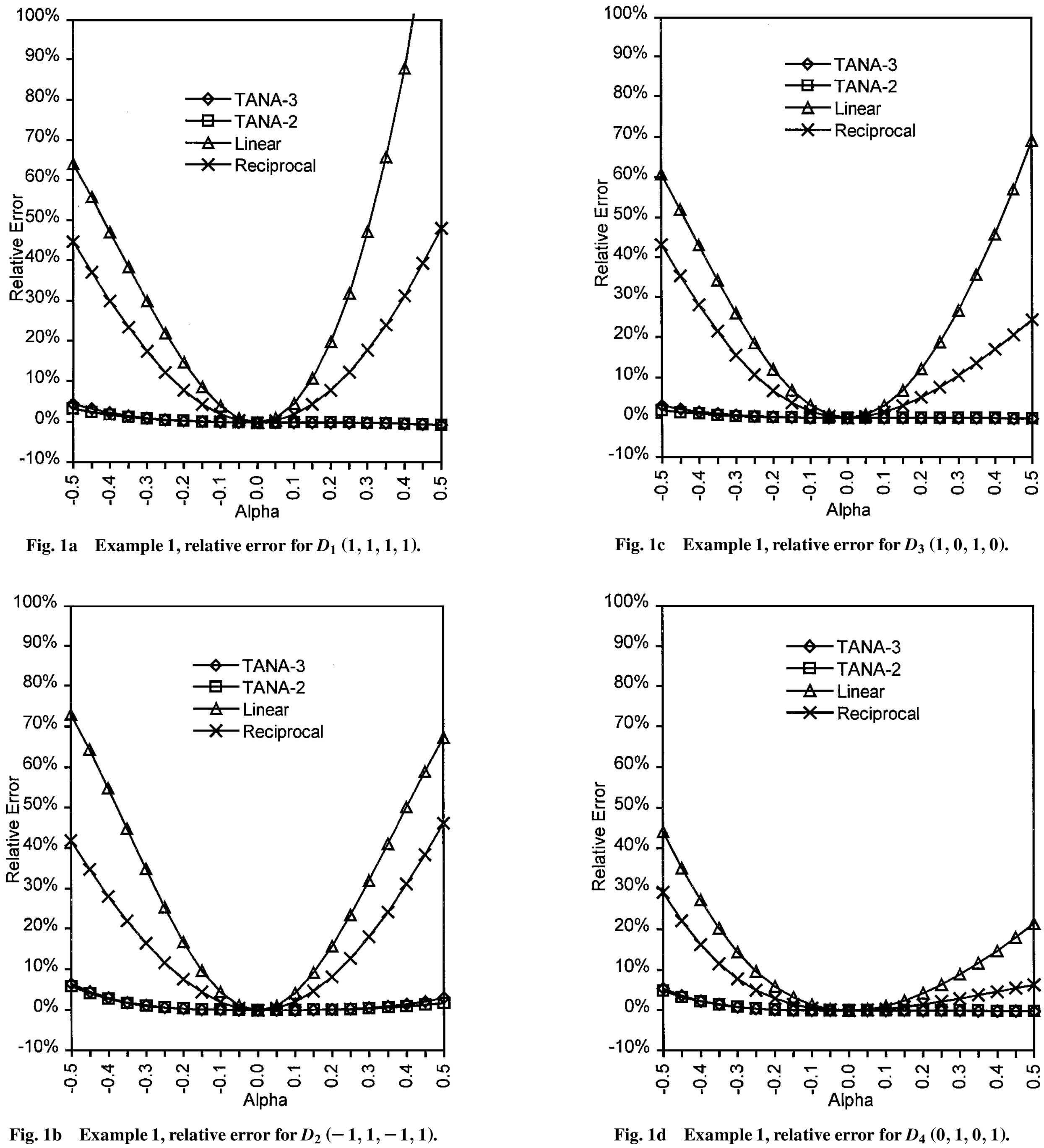

Fig. 1a Example 1, relative error for D_1 (1, 1, 1, 1).

Fig. 1b Example 1, relative error for D_2 (−1, 1, −1, 1).

Fig. 1c Example 1, relative error for D_3 (1, 0, 1, 0).

Fig. 1d Example 1, relative error for D_4 (0, 1, 0, 1).

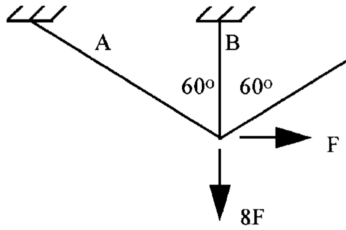

Fig. 2 Three-bar truss structure.

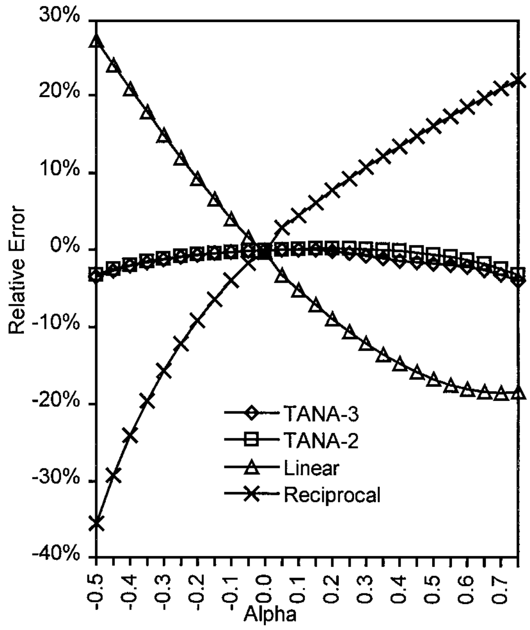

Fig. 3 Example 2, relative error.

TANA-3 behave almost the same and have higher accuracy (with relative error less than 6.5%) compared with the linear and reciprocal approximation (with relative error up to 143% and 73% for the linear and reciprocal, respectively). So, both TANA-3 and TANA-2 work very well for this example. The exponents for x_1, x_2, x_3, and x_4 in TANA-2 are -2.7375, -1.45, -2.825, and -2.475, respectively, and ε is 0.5527 (see Ref. 6). The exponents in TANA-3 are $-2.6383, -1.5010, -2.8298$, and -2.4591, respectively, which are very close to the corresponding values in TANA-2; H is 0.2422.

Example 2: Three-bar structure. The three-bar truss example shown in Fig. 2 is taken from Ref. 13. The truss is designed subject to stress and displacement constraints with cross-sectional area A_a, A_b, and $A_c (A_a = A_c)$ as design variables. The approximation of member C with a stress constraint is examined. Using normalized variables, the stress constraints are written as

$$g(\boldsymbol{X}) = 1 + \frac{\sqrt{3}}{3x_1} - \frac{2}{x_2 + 0.25x_1} \tag{35}$$

where $g(\boldsymbol{X})$ is expanded at the point $\boldsymbol{X}_2(1.0, 1.0)$ for the linear, reciprocal, TANA-2, and TANA-3 approximations. The previous point is taken as (0.75, 1.25). The search direction is $\boldsymbol{D} = (-0.5, 1.0)$. The exponent sets for TANA-2 are $p_1 = -1.150$, $p_1 = -0.270$, and $\varepsilon = 0.370$. TANA-3 has $p_1 = -2.8742$, $p_1 = -0.2527$, and $H = 0.01654$. We plot the results of TANA-3 and TANA-2 in Fig. 2, accompanied by other approximations. Figure 3 shows that TANA-3 and TANA-2 have almost the same accuracy and that both are better than the linear and reciprocal approximations. TANA-3 and TANA-2 have a maximum relative error of only 3.8%, whereas the reciprocal and linear approximations have 35% and 27%, respectively.

Part 2: Applications in Structural Optimization

A computer program based on the formulations of TANA-3, TANA-2, linear, and reciprocal approximations has been developed. Several structural problems are used to test the adaptivity of TANA-3 and compare it with TANA-2, linear, and reciprocal approximations. These examples include the size optimization of 10- and 25-bar structures. The constraints cover displacements and stresses. For stress constraints, internal forces are first approximated as intermediate response functions. Stresses are calculated by using the approximated forces. An explicit approximate subproblem is solved by using the SQP algorithm of the DOT[14] program. The absolute value of p_i calculated in Eq. (18) may result in a large value and cause the approximation to deteriorate. The maximum $p_{\max}$ of exponential value p_i is limited. When p_i calculated in Eq. (18) is less than $-p_{\max}$ or greater than $p_{\max}$, it is rounded up to $-p_{\max}$ or down to $p_{\max}$. TANA-2 uses the same limit of p_i as TANA-3. The functions approximated include the displacements and internal forces for truss structures. For statically determinate trusses, the displacement is a linear function of the reciprocal of areas but the force is constant. So we take $p_{\max} = 1.5$ for all problems presented in this paper; $p_{\max} = 1.5$ provides enough space for p_i, although all examples are statically indeterminate. (In fact, $p_{\max} = 1.0 \sim 3$ can work well for the examples, but $p_{\max} = 1.5$ is a typical value. For a beam problem, a larger $p_{\max}$, such as 1–5, can be taken, depending on different intermediate variables.) In TANA-3 and TANA-2, reciprocal approximation is used for the first iteration. We compare two aspects: stability and efficiency. A good method should be converged in as few FE analyses as possible without oscillations of objective function and maximum constraint violation.

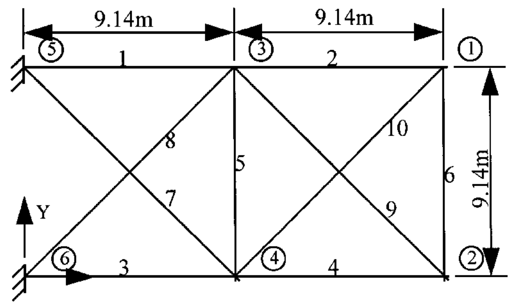

Fig. 4 Ten-bar truss structure.

Example 3: Ten-bar structure with only stress limits (case 1). A 10-bar planar truss with stress constraints is shown in Fig. 4. The material properties and nodal loading are given as Young's modulus $E = 6.90 \times 10^{10}$ N/m^2, weight density $\rho = 2.77 \times 10^3$ kg/m^3, allowable stresses $\sigma_a = \pm 1.72 \times 10^8$ N/m^2, one-load case, and $P_{2y} = P_{4y} = -4.45 \times 10^5$ N. The area of each bar is considered as a design variable. The initial value and minimum size limit are 6.45×10^{-3} and 6.45×10^{-5} m^2, respectively, for every variable. This problem is convergent only with move limits. Two move limits are selected: 50% and 75%. Iteration histories of structural weight and maximum constraint are shown in Tables 1 and 2. Under a 50% move limit, TANA-3, TANA-2, linear, and reciprocal approximations need 9, 11, 10, and 14 iterations, respectively, to reach the optimum points. For a 75% move limit, TANA-3, TANA-2, linear, and reciprocal methods converge within 7, 10, 6, and 13 iterations. TANA-3 and linear methods perform well. TANA-2 is better than the reciprocal but worse than TANA-3 and linear methods. We also note that the linear method works very well for this example. As we know, the stresses usually have a tendency to be linear with respect to the reciprocals of truss member areas. The internal forces do not behave the same way. It is possible for the internal forces to tend to be linear with respect to area. For the first subproblem of the optimization procedure, we use the reciprocal approximation for TANA-3 and TANA-2, which may cause TANA-3 and TANA-2 be slower than the linear approximation in this example.

The linear approximation is usually inferior in other examples where both the stress and displacement constraints exist. The displacement is usually more linear with respect to the reciprocal variables than the direct variables. When the linear approximation is used for the displacement, it can produce large errors.

Example 4: Ten-bar structure with both stress and displacement constraints (case 2). This example is the same as example 3, with the addition of displacement constraint on each degree of freedom. The

Table 1 Iteration histories of 10-bar structure with stress constraints (50% move limits)

Iteration number	TANA-3		TANA-2		Linear		Reciprocal	
	Obj.,[a] kg	Max. con.[b]	Obj., kg	Max. con.	Obj., kg	Max. con.	Obj., kg	Max. con.
1	1905.21	−0.1814599	1905.21	−0.1814599	1905.21	−0.181460	1905.21	−0.1814599
2	1082.65	7.4412E−02	1082.65	7.4412E−02	1081.75	3.5503E−02	1082.65	7.4412E−02
3	845.08	2.0914E−02	845.67	2.0871E−02	819.42	5.0068E−02	859.06	1.8415E−02
4	781.97	4.4230E−03	781.61	1.2582E−02	787.55	5.0648E−02	789.60	0.1456985
5	749.42	1.8496E−02	744.42	2.5144E−02	758.63	4.9847E−03	769.26	1.4867E−02
6	734.21	6.8554E−03	733.62	5.9579E−02	738.79	1.4380E−03	759.09	1.1579E−02
7	727.04	6.8863E−03	742.43	8.9091E−03	728.53	6.8521E−04	744.38	1.6307E−03
8	723.27	1.5981E−03	730.80	2.2169E−02	724.27	1.3232E−05	736.48	5.7083E−03
9	723.31	1.4782E−05	722.27	4.7012E−02	722.95	1.6731E−03	731.94	4.4265E−03
10			723.00	3.0667E−03	723.54	1.6224E−03	729.03	3.7027E−03
11			723.31	1.2636E−05			726.94	2.8520E−03
12							725.45	2.9615E−03
13							723.90	8.7654E−04
14							723.40	3.8754E−04

[a]Obj. = objective function. [b]Max. con. = maximum constraint.

Table 2 Iteration histories of 10-bar structure with stress constraints (75% move limits)

Iteration number	TANA-3		TANA-2		Linear		Reciprocal	
	Obj.,[a] kg	Max. con.[b]	Obj., kg	Max. con.	Obj., kg	Max. con.	Obj., kg	Max. con.
1	1905.21	−0.1814599	1905.21	−0.1814599	1905.21	−0.1814599	1905.21	−0.1814599
2	841.44	0.2672212	841.44	0.2671512	823.69	4.8006E−02	841.44	0.2671512
3	816.34	9.6188E−03	802.85	2.8929E−02	743.43	8.8431E−02	817.15	1.6915E−02
4	749.28	1.6353E−03	753.19	0.1242206	727.81	4.7351E−02	763.58	3.8541E−02
5	726.00	1.0286E−02	736.12	0.1104941	723.18	3.5655E−03	745.92	8.8006E−02
6	722.86	2.9851E−03	735.07	8.7960E−02	723.72	−1.7374E−4	739.79	1.3679E−02
7	723.31	1.0728E−05	726.49	0.1487311			734.98	5.9657E−03
8			723.40	5.3523E−02			731.71	2.5004E−03
9			723.40	3.5548E−03			728.67	2.9754E−03
10			723.40	−7.4267E−04			726.63	3.0989E−03
11							725.17	3.1738E−03
12							723.72	9.7858E−04
13							723.36	1.9407E−03

[a]Obj. = objective function. [b]Max. con. = maximum constraint.

Table 3 Iteration histories of 10-bar structure with stress and displacement constraints

Iteration number	TANA-3		TANA-2		Reciprocal	
	Obj.,[a] kg	Max. con.[b]	Obj., kg	Max. con.	Obj., kg	Max. con.
1	1905.21	0.9697877	1905.21	0.9697877	1905.21	0.9697877
2	2550.89	0.1194136	2550.89	0.1194136	2550.89	0.1194136
3	2333.24	0.2662637	2546.03	5.1469E−02	2554.11	−8.2085E−03
4	2274.22	0.3073106	2321.17	5.3676E−02	2503.27	−1.1567E−02
5	2272.91	8.3335E−02	2093.58	0.4547451	2440.43	−1.2465E−02
6	2300.65	5.5652E−03	2266.69	0.5170441	2367.38	−2.2432E−03
7	2297.78	2.1457E−06	2283.62	1.5179E−02	2259.33	2.6036E−02
8	2297.56	9.5963E−05	2310.36	−1.3037E−03	2297.74	2.8543E−03
9			2297.56	3.4034E−04	2300.37	2.3543E−05
10			2298.01	1.3661E−04	2300.37	2.3543E−05
11			2298.01	1.3661E−04		

[a]Obj. = objective function. [b]Max. con. = maximum constraint.

displacement limit is $\pm 8.89 \times 10^{-3}$ m. No move limits are imposed. Table 3 shows the objective function and maximum constraints as the function of iteration number. TANA-3, TANA-2, and reciprocal need 8, 11, and 10 FE analyses, respectively, to reach the optimum. TANA-3 finds the optimum with the fewest iterations and produces lightest weight. The linear method is not convergent.

Example 5: Twenty-five-bar truss with stress and displacement constraints. The geometric information is shown in Fig. 5. Young's modulus $E = 6.90 \times 10^{10}$ N/m^2 and weight density $\rho = 2.77 \times 10^3$ kg/m^3. The displacement limit is $\pm 8.89 \times 10^{-3}$ m. The initial value and minimum size limit are 6.45×10^{-3} and 6.45×10^{-5} m^2, respectively, for every variable. This structure is subjected to two load cases. The first load case is $P_{1x} = 4.45 \times 10^3$ N,

Table 4 Variable linking and allowable stresses for 25-bar structure

Variable number	Tensile stress limit, N/m^2	Compression stress limit, N/m^2	Corresponding elements of each design variable
1	2.76×10^8	-2.42×10^8	1–2
2	2.76×10^8	-7.99×10^7	1–4; 2–3; 1–5; 2–6
3	2.76×10^8	-1.19×10^8	2–4; 2–5; 1–6; 1–3
4	2.76×10^8	-2.42×10^8	4–5; 3–6
5	2.76×10^8	-2.42×10^8	3–4; 5–6
6	2.76×10^8	-4.66×10^7	3–10; 6–7; 5–8; 4–9
7	2.76×10^8	-4.80×10^7	4–7; 3–8; 5–10; 6–9
8	2.76×10^8	-7.64×10^7	6–10; 3–7; 4–8; 5–9

Table 5 Iteration histories of 25-bar structure with stress and displacement constraints

Iteration number	TANA-3		TANA-2		Reciprocal	
	Obj.,[a] kg	Max. con.[b]	Obj., kg	Max. con.	Obj., kg	Max. con.
1	1501.47	−0.7779445	1501.47	−0.7779445	1501.47	−0.7779445
2	258.18	−3.9815E−04	258.18	−3.9815E−04	258.18	−3.9815E−04
3	248.09	−2.8288E−04	247.73	−3.0553E−04	248.23	−3.8743E−05
4	247.65	4.3272E−05	247.48	1.9824E−04	247.78	−8.3684E−05
5	247.61	2.3806E−04			247.58	−1.0770E−03

[a]Obj. = objective function. [b]Max. con. = maximum constraint.

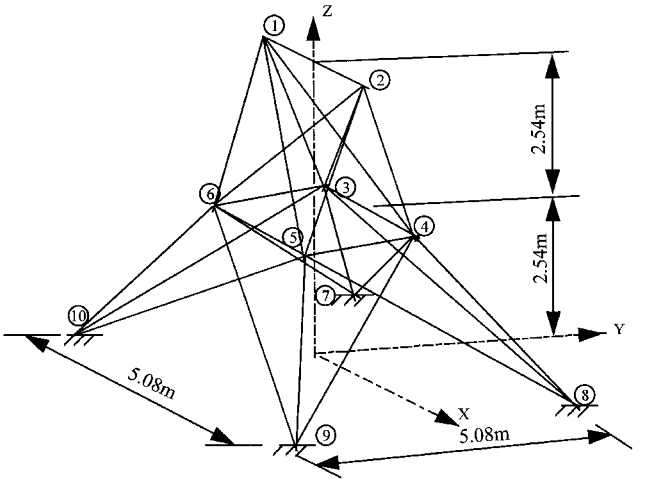

Fig. 5 Twenty-five-bar truss structure.

$P_{1y} = 4.45 \times 10^4$ N, $P_{1z} = -2.225 \times 10^4$ N, $P_{2x} = 4.45 \times 10^4$ N, $P_{2z} = -2.225 \times 10^4$ N, $P_{3x} = 2.225 \times 10^3$ N, and $P_{6x} = 2.225 \times 10^3$ N. The second load case is $P_{1y} = 8.90 \times 10^4$ N, $P_{1z} = -2.225 \times 10^4$ N, $P_{2y} = -8.90 \times 10^4$ N, and $P_{2z} = -2.225 \times 10^4$ N. Variable linking and stress limits are listed in Table 4. This problem is easy to solve for every method except the linear one. As Table 5 shows, TANA-2 needs only four iterations, whereas both TANA-3 and reciprocal methods require five iterations. The linear method does not converge at all.

We observe that TANA-3 and TANA-2 have almost the same accuracy for the problems in Part 1 of this section. TANA-3 exhibits slightly faster convergence than TANA-2 for most problems in Part 2. Thus, we can say that TANA-3 has at least the same ability to approximate the constraints well. Both TANA-3 and TANA-2 also work much better than the linear and reciprocal for most examples.

When we conclude that TANA-3 is as accurate as TANA-2, we like to emphasize again the computational cost difference between TANA-3 and TANA-2. TANA-2 takes more time to solve Eqs. (5) and (6) for every example, whereas TANA-3 needs little effort to get the exponential value. It is the most important advantage of TANA-3 over TANA-2.

Conclusions

A new two-point function approximation has been developed and applied to several examples. This two-point approximation is an incomplete second-order Taylor-series expansion with respect to the intervening variables, where only diagonal terms of the Hessian matrix are retained. The diagonal terms change as the variables change, and so this method is accurate over a wider range of variables. Exponential values for each intervening variable and the parameter in second-order terms are evaluated by matching both the approximate derivatives and function with the exact derivatives and function corresponding to the previous point. Thus, both the approximate function and the derivative values at two data points are equal to the exact counterparts. More important is that all of the exponential and parameter values can be obtained in a closed-form solution, which is computationally inexpensive. The suggested two-point approximation has high accuracy and low computational cost.

Acknowledgment

This research work was supported by NASA Lewis Research Center through Grant NAG 3-1489.

References

[1]Barthelemy, J.-F. M., and Haftka, R. T., "Approximation Concepts for Optimum Structural Design—A Review," *Structural Optimization*, Vol. 5, No. 3, 1993, pp. 129–144.

[2]Vanderplaats, G. N., Thomas, H. L., and Shyy, Y. K., "Review of Approximation Concepts for Structural Synthesis," *Journal of Computing Systems in Engineering*, Vol. 2, No. 1, 1991, pp. 17–25.

[3]Vanderplaats, G. N., "Structural Design Optimization—Status and Direction," *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 38th Structures, Structural Dynamics, and Materials Conference* (Kissimmee, FL), AIAA, Reston, VA, 1997, pp. 1178–1192.

[4]Grandhi, R. V., "Structural Optimization with Frequency Constraints—A Review," *AIAA Journal*, Vol. 31, No. 12, 1993, pp. 2296–2303.

[5]Wang, L., and Grandhi, R. V., "Multi-Point Approximations: Comparisons Using Structural Size, Configuration and Shape Design," *Structural Optimization*, Vol. 12, Nos. 2–3, 1996, pp. 177–185.

[6]Wang, L. P., and Grandhi, R. V., "Improved Two-Point Function Approximation for Design Optimization," *AIAA Journal*, Vol. 33, No. 9, 1995, pp. 1720–1727.

[7]Haftka, R. T., Nachlas, J. A., Watson, L. T., Rizzo, T., and Desai, R., "Two-Point Constraint Approximation in Structural Optimization," *Computer Methods in Applied Mechanics and Engineering*, Vol. 60, No. 3, 1987, pp. 289–301.

[8]Fadel, G. M., Riley, M. F., and Barthelemy, J. F. M., "Two-Point Exponential Approximation Method for Structural Optimization," *Structural Optimization*, Vol. 2, No. 2, 1990, pp. 117–124.

[9]Belegundu, A. D., Rajan, S. D., and Rajgopal, J., "Exponential Approximation in Optimal Design," *Research in Structures, Structural Dynamics and Materials*, compiled by J.-F. M. Barhelemy and A. K. Noor, NASA CP-3064, 1990, pp. 137–150.

[10]Snyman, J. A., and Stander, N., "New Successive Approximation Method for Optimum Structural Design," *AIAA Journal*, Vol. 32, No. 6, 1994, pp. 1310–1315.

[11]Wang, L. P., and Grandhi, R. V., "Efficient Safety Index Calculation for Structural Reliability Analysis," *Computers and Structures*, Vol. 52, No. 1, 1994, pp. 103–111.

[12]Atkinson, K. E., *An Introduction to Numerical Analysis*, 2nd ed., Wiley, New York, 1989, pp. 507–529.

[13]Haftka, R. T., and Gurdal, Z., *Elements of Structural Optimization*, Kluwer Academic, Dordrecht, The Netherlands, 1992, pp. 215–218.

[14]"DOT Users Manual, Version 4.2," Vanderplaats Research and Development, Inc., Colorado Springs, CO, 1995, Chap. 2.

A. D. Belegundu
Associate Editor