Chapter 4
Sensitivity Analysis

Nam H. Kim

Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL, USA

1 INTRODUCTION

As discussed in Chapter eae495, the numerical optimization techniques are classified as either local (typically gradient-based) or global (typically nongradient-based or evolutionary) algorithms. Advantages and disadvantages of each algorithm are discussed in Chapter eae495. In the viewpoint of this chapter, the former requires both function values and gradients, while the latter only requires function values. This chapter focuses on how to calculate the gradients during optimization. Although the optimization can be applied to any engineering applications, we will explain the calculation of gradients in structural applications.

In optimization problems, the objective and constraint functions are called *performance measures*. Sensitivity, or gradient, is the rate of performance measure change with respect to design variable changes. With structural analysis, the sensitivity analysis provides critical information, the gradient, for optimization. Obviously, the performance measure is presumed to be a differentiable function of the design, at least in the neighborhood of the current design point. For complex engineering applications, it is not simple to prove a performance measure's differentiability with respect to the design. In this chapter, we assume that the performance measure is continuously differentiable with respect to the design.

In general, a performance measure depends on the design. For example, a change in the cross-sectional area of a beam would affect the structural weight. This type of dependence is simple if the expression of weight is known in terms of the design variables. For example, the weight of a straight beam with a circular cross section can be expressed as

$$W(r) = \rho \pi r^2 L \tag{1}$$

where ρ is the density of the material, r the radius, and L the length of the beam. If the radius is a design variable then the design sensitivity of W with respect to r would be

$$\frac{\mathrm{d}W}{\mathrm{d}r} = 2\rho\pi r L \tag{2}$$

This type of function is *explicitly dependent* on the design, since the function can be explicitly written in terms of that design. Consequently, only algebraic manipulation is involved, and no expensive computation is required to obtain the sensitivity of an explicitly dependent performance measure.

However, in most cases, performance measures do not explicitly depend on the design. For example, when the stress in complex frames is considered as a performance measure, there is no simple way to express the sensitivity of stress explicitly in terms of the radius $x = r$ of the cross section. In a linear elastic problem, the stress of the structure is often determined from the displacement, which can be calculated using finite element analysis. In such a case, the sensitivity

Encyclopedia of Aerospace Engineering.
Edited by Richard Blockley and Wei Shyy
 ISBN: 978-0-470-75440-5

of stress $\sigma(\boldsymbol{q})$ can be written as

$$\frac{\mathrm{d}\sigma}{\mathrm{d}x} = \frac{\mathrm{d}\sigma}{\mathrm{d}\boldsymbol{q}}^{\mathrm{T}} \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}x} \tag{3}$$

where $\boldsymbol{q}$ is the vector of displacements of the beam. Since the expression of stress as a function of displacement is usually known, $\mathrm{d}\sigma/\mathrm{d}\boldsymbol{q}$ can easily be obtained. The only difficulty is the computation of $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$, which is the state variable (displacement) sensitivity with respect to the design variable x.

When a design engineer wants to compute the design sensitivity of performance measures such as stress $\sigma(\boldsymbol{q})$ in equation (3), the structural problem has presumably already been solved. One of the most common approaches in solving structural problems is to use the principle of virtual work, which is convenient for formulating the equations of equilibrium, as

$$\iint_V \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{\varepsilon}\,\mathrm{d}V = \iint_V \delta\boldsymbol{u}^{\mathrm{T}}\boldsymbol{b}\,\mathrm{d}V + \iint_A \delta\boldsymbol{u}^{\mathrm{T}}\boldsymbol{h}\,\mathrm{d}A \tag{4}$$

for all $\delta\boldsymbol{u}$ that belong to the space of kinematically admissible displacements. In equation (4), $\delta\boldsymbol{\varepsilon}$ is the virtual strain, $\boldsymbol{C}$ the elasticity matrix, $\boldsymbol{\varepsilon}$ the strain vector, $\boldsymbol{b}$ ~~denotes~~ the external load per unit volume, and $\boldsymbol{h}$ ~~reflects~~ tractions acting on the outer surface A of the structure.

Since the structural equation (4) is difficult to be solved analytically, numerical methods are often employed. Finite element analysis is one of the most popular numerical methods to solve the structural equation (4). After discretizing the structural equation with a set of finite elements and applying the principle of virtual work in each element, the following linear algebraic equation can be obtained:

$$\boldsymbol{K}(x)\boldsymbol{q} = \boldsymbol{F}(x) \tag{5}$$

Equation (5) is a matrix equation of finite elements if $\boldsymbol{K}$ and $\boldsymbol{F}$ are understood to be the stiffness matrix and load vector, respectively. Suppose the explicit expressions of $\boldsymbol{K}(x)$ and $\boldsymbol{F}(x)$ are known and differentiable with respect to x. Since the stiffness matrix $\boldsymbol{K}(x)$ and load vector $\boldsymbol{F}(x)$ depend on the design x, solution $\boldsymbol{q}$ also depends on the design x. However, it is important to note that this dependency is implicit, which is why we need to develop a methodology for sensitivity analysis. As shown in equation (3), $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$ must be computed using the governing equation (5). This can be achieved by differentiating equation (5) with respect to x, as

$$\boldsymbol{K}(x)\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}x} = \frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q} \tag{6}$$

Assuming that the explicit expressions of $\boldsymbol{K}(x)$ and $\boldsymbol{F}(x)$ are known, $\mathrm{d}\boldsymbol{K}/\mathrm{d}x$ and $\mathrm{d}\boldsymbol{F}/\mathrm{d}x$ can be evaluated. Thus, if solution $\boldsymbol{q}$ from equation (5) is known, then $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$ can be computed from equation (6), which can then be substituted into equation (3) to compute $\mathrm{d}\sigma/\mathrm{d}x$. Note that the stress performance measure is *implicitly dependent* on the design through state variable $\boldsymbol{q}$.

In this chapter, it is assumed that the general performance measure ψ depends on the design explicitly and implicitly. That is, the performance measure ψ is presumed to be a function of design x and state variable $\boldsymbol{q}(x)$, as

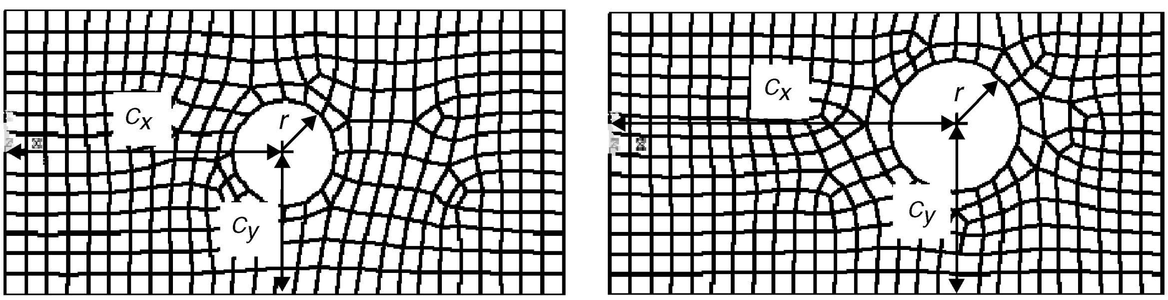
$$\psi = \psi(\boldsymbol{q}(x), x) \tag{7}$$

The sensitivity of ψ can thus be expressed as

$$\frac{\mathrm{d}\psi(\boldsymbol{q}(x), x)}{\mathrm{d}x} = \left.\frac{\partial\psi}{\partial x}\right|_{\boldsymbol{q}=\mathrm{const}} + \left.\frac{\partial\psi}{\partial\boldsymbol{q}}\right|_{x=\mathrm{const}}^{\mathrm{T}} \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}x} \tag{8}$$

The only unknown term in equation (8) is $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$. Various computational methods to obtain $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$ are introduced in the following sections.

From the sensitivity analysis point of view, design variables are often classified as either shape or non-shape design. In general, a structural equation is represented by an integral form. When a design variable changes the integral



(a) Initial design (b) Perturbed design

Figure 1. Example of shape design variables in a plate with a hole. (a) Initial design; (b) perturbed design.

domain, it is called a shape design. Otherwise, the design is referred to as a non-shape design. Since a non-shape variable appears as a parameter in the structural equation, it is relatively easy to differentiate the structural equation with respect to the non-shape design. However, in the case of shape design, the change in design variables modifies the integral domain, which is more difficult to differentiate. When finite element analysis is used to solve the structural equation, the shape design variables change the finite element mesh. Figure 1 illustrates how the shape design variables (C_x, C_y, and r) change the structural domain. In the initial design (Figure 1a), C_x and C_y describe the location of the center of a hole and r ~~is~~ the radius of the hole. When these variables ~~are changed~~ (Figure 1b), the finite element mesh needs to be modified.

2 METHODS OF SENSITIVITY ANALYSIS FOR LINEAR STATIC STRUCTURES

Various methods employed in sensitivity analysis are listed in Figure 2. Four approaches are used to obtain ~~the~~ sensitivity: the global finite difference, discrete, continuum, and automatic derivatives. In the finite difference approach, the sensitivity is obtained by either the *forward finite difference* or the *central finite difference method*. In the discrete method, the sensitivity is obtained by taking derivatives of the discrete governing equation. For this process, it is necessary to take the derivative of the stiffness matrix. If this derivative is obtained analytically using the explicit expression of the stiffness matrix with respect to the design variable, it is ~~an~~ *analytical method*, since the analytical expressions of $\boldsymbol{K}(\boldsymbol{x})$ and $\boldsymbol{F}(\boldsymbol{x})$ are used. However, if the derivative is obtained using a finite difference method, ~~the method~~ is called ~~a~~ *semi-analytical method*. In the continuum approach, the derivative of the structural equation (4) is

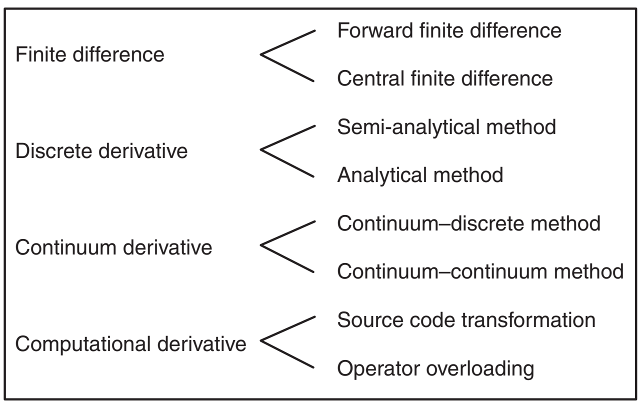


Figure 2. Approaches to design sensitivity analysis.

taken before it is discretized. If the structural problem and sensitivity equations are solved analytically, then it is called the *continuum–continuum method*. However, only very simple, classical problems can be solved analytically. Thus, the continuum sensitivity equation is solved by discretization in the same way that structural problems are solved. Since differentiation is taken at the continuum domain and is then followed by discretization, this method is called the *continuum–discrete method*. Finally, computational, algorithmic, or automatic differentiation refers to a differentiation of the computer code itself.

Except for the global finite differences option, the other three come in direct and adjoint methods (called the reverse mode for automatic derivatives). In the direct method, one obtains the derivatives of the entire structural response and often of intermediate quantities as well. The sensitivities of performance measures can then be obtained from the chain rule of differentiation. In the adjoint method, one defines an adjoint problem that depends on the performance measure. The sensitivities of performance measures can then be obtained using the structural and adjoint responses. Thus, the entire system response sensitivities are not required, which is particularly an advantage in cases with many design variables, but few performance measures of interest.

2.1 Global finite difference method

The easiest way to compute sensitivity information of a performance measure is ~~by using~~ the global finite difference method. Different designs yield different analysis results and, thus, different performance values. The global finite difference method computes the sensitivity by evaluating performance measures at different values of design variables. Although the given optimization problem may have many design variables, a single design variable is considered in the following explanation. If x is the current design then the analysis results provide the value of performance measure $\psi(x)$. In addition, if the design is perturbed to $x + \Delta x$, where Δx represents a small change in the design, then the sensitivity of $\psi(x)$ can be approximated as

$$\frac{\mathrm{d}\psi}{\mathrm{d}x} \approx \frac{\psi(x + \Delta x) - \psi(x)}{\Delta x} \tag{9}$$

Equation (9) is called the *forward difference method* since the design is perturbed by $+\Delta x$. If $-\Delta x$ is substituted in equation (9) for Δx, then the equation is defined as the *backward difference method*. Additionally, if the design is perturbed in both directions, such that the design sensitivity is approximated by

$$\frac{\mathrm{d}\psi}{\mathrm{d}x} \approx \frac{\psi(x+\Delta x) - \psi(x-\Delta x)}{2\Delta x} \tag{10}$$

then the equation is defined as the *central difference method*.

The advantage of the finite difference method is obvious. If structural analysis can be performed and the performance measure can be obtained as a result of structural analysis, then the expressions in equations (9) and (10) become virtually independent of the problem types considered. Consequently, this method has been popular in engineering design.

However, sensitivity computation costs become the dominant concern in the design process. If n represents the number of designs, then $n+1$ number of analyses have to be carried out for either forward or backward difference methods, and $2n + \cancel{1}$ analyses are required for the central difference method. For modern, practical engineering applications, the cost of structural analysis is rather expensive. Thus, this method is infeasible for large-scale problems containing many design variables.

Another major disadvantage of the finite difference method is the degree of accuracy of its sensitivity results. In equation (9), accurate results can be expected when Δx approaches zero. Figure 3 shows some sensitivity results using the finite difference method. The tangential slope of the curve at x_0 is the exact sensitivity value. Depending on perturbation size, the sensitivity results are quite different. For a mildly nonlinear performance measure, relatively large perturbation can still provide a reasonable estimation of sensitivity results. However, for highly nonlinear performances, a large perturbation yields completely inaccurate results. Thus, the determination of perturbation size greatly affects the sensitivity result. Even though it may be necessary to choose a very small perturbation, numerical noise becomes dominant

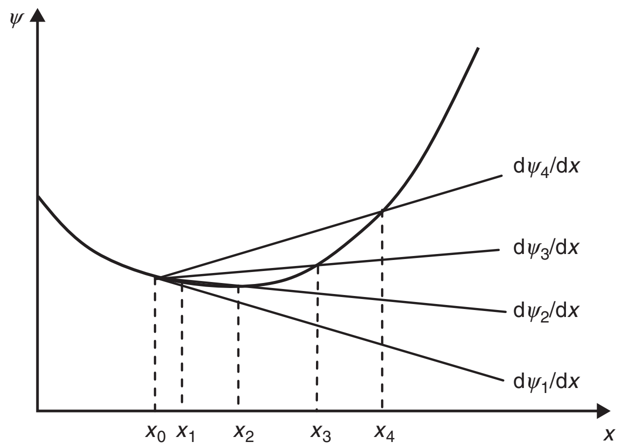


Figure 3. Influence of step-size in forward finite difference method.

for a too small perturbation size. That is, with a too small perturbation, no reliable difference can be found in the analysis results. The most obvious source of this type of error is from computational errors associated with arithmetic operations with finite number of digits and possibly ill-conditioning in the problem. For example, if up to five digits of significant numbers are valid in a structural analysis then any design perturbation in the finite difference that is smaller than the first five significant digits cannot provide meaningful results. As a result, it is very difficult to determine design perturbation sizes that work for all problems. One of the other potential sources of error is the discretization of both the spatial and ~~the~~ temporal domain. A typical example could be numerical noise induced by re-meshing.

Computational efficiency, accuracy and consistency, and implementation effort for global finite differences depend to a large extent on the type of solvers used for the linear system of equations (5). The main issue is whether computational investments associated with solving ~~the~~ equations for the nominal structure can help reduce the effort associated with solving these equations for a perturbed structure.

When the matrix equation (5) is solved by factorizing the matrix $\boldsymbol{K}$, there is an array of methods that provide fast reanalysis of the perturbed structure. A disadvantage of many of these techniques is that accuracy is generally compromised, that is, certain inaccuracies will be introduced. When the perturbation leads to a low rank modification of $\boldsymbol{K}$, for example, because only a single finite element is modified, then an exact analysis of the perturbed structure can be performed using the Sherman–Morrison–Woodbury formulas (Akgün, Garcelon and Haftka, 2001). The main computational cost of this approach is the solution of equation (5) for a number of right-hand sides equal to the rank of the perturbation in $\boldsymbol{K}$. Akgün, Garcelon and Haftka (2001) discuss several variants of this approach including the method of virtual distortions. When the perturbation in the matrix is more extensive, as in shape design, it is still possible to use a binomial series solution (Yoon and Belegundu, 1988) or a similar approximation of the inverse of $\boldsymbol{K}$ using ~~a~~ Neumann series (Oral, 1996).

2.2 Discrete method

A structural problem is often discretized in finite dimensional space in order to solve complex problems. The discrete method computes the performance sensitivity of the discretized problem, where the governing equation is a system of linear equations, as in equation (5). If the explicit form of the stiffness matrix $\boldsymbol{K}(x)$ and the load vector $\boldsymbol{F}(x)$ is known, and if solution $\boldsymbol{q}$ of matrix equation $\boldsymbol{K}(x)\boldsymbol{q} = \boldsymbol{F}(x)$ is obtained, then the sensitivity of the displacement vector can also be obtained from equation (6), as

$$\boldsymbol{K}(x)\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}x} = \boldsymbol{p} \tag{11}$$

where the pseudo-load vector $\boldsymbol{p}$ is defined as

$$\boldsymbol{p} = \frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q} \tag{12}$$

It is clear from equation (11) that the sensitivities require the solution of the same set of equations as solved for the response functions, but for different right-hand sides (compare with equation (5))~~. T~~he latter being the pseudo-load vector, ~~see~~ equation (12).

Once $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$ is calculated, the sensitivity of a performance measure can be obtained from equation (8) as

$$\frac{\mathrm{d}\psi}{\mathrm{d}x} = \frac{\partial\psi}{\partial x} + \frac{\partial\psi^{\mathrm{T}}}{\partial\boldsymbol{q}}\boldsymbol{K}^{-1}\left(\frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q}\right) \tag{13}$$

In practice, it is unnecessary to calculate the inverse of the stiffness matrix, $\boldsymbol{K}^{-1}$. Instead, $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$ is solved from equation (11) and substituted ~~in~~ equation (8).

Since the sensitivity equation (11) uses the same stiffness matrix $\boldsymbol{K}(x)$ ~~with~~ the original structural problem, it can be solved very efficiently if the direct matrix solver is used. The direct matrix solver first decomposes the stiffness matrix into lower- and upper-triangular portions (L-U decomposition), which is independent of the right-hand side. A ~~back~~-substitution process is then employed to solve equation (5). Since the L-U decomposition takes most ~~of~~ computational time and since it is independent of the right-hand side, the sensitivity equation (11) uses the already~~-~~decomposed stiffness matrix to efficiently solve for $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$. This process is similar to the multiple right-hand side solution process in finite element analysis.

In calculating the pseudo-load vector, it is unnecessary to differentiate the global load vector and stiffness matrix, but to differentiate only those finite elements that are affected by the design variable. The evaluation of the pseudo-load vector is then carried out by an assembly of all individual nodal points and finite element contributions. These contributions are obtained by differentiating the finite element stiffness matrices with respect to the design variables and following a similar procedure for all load contributions. The fact that the pseudo-load vector only depends on elements that are affected may be exploited to make the computation of the pseudo-load vector more efficient. For shape design variables, this requires some additional attention. For that purpose, one often tries to link the design variables ~~only~~ to boundary elements, which means only ~~a~~ boundary layer of elements is affected by the shape design variables.

The analytical differentiation process may become tedious. This especially holds true for shape design variables. Additional procedures must be implemented for each element used within the sensitivity analysis. The procedure must account for all possible design variables and particularly for shape design variables as they are usually more complex than the original finite element routines. This type of discrete design sensitivities are referred to as *analytical* discrete design sensitivities.

It is not difficult to compute $\mathrm{d}\boldsymbol{F}/\mathrm{d}x$, since the applied force is usually either independent of the design or it has a simple expression. However, the computation of $\mathrm{d}\boldsymbol{K}/\mathrm{d}x$ in equation (11) depends on the type of problem. In addition, modern advances in the finite element method use numerical integration in the computation of $\boldsymbol{K}$. In this case, the explicit expression of $\boldsymbol{K}$ in terms of x may not be available. Moreover, in the case of the shape design variable, computation of the analytical derivative of the stiffness matrix is quite costly. Because of this, approximations are frequently accepted for the pseudo-load vector that reduces this effort. These approximations particularly involve finite difference schemes for evaluation of the pseudo-load vector. Forward and central finite difference schemes are most popular. This type of design sensitivity is commonly denoted as *semi-analytical* discrete design sensitivit~~ies~~. However, Barthelemy and Haftka (1988) show that the semi-analytical method can have serious accuracy problems for shape design variables in structures modeled by beam, plate, truss, frame, and solid elements. They found that accuracy problems occur even for a simple cantilever beam. Moreover, errors in the early stage of approximation multiply during the matrix equation solution phase. As a remedy, Olhoff, Rasmussen and Lund (1993) proposed an exact numerical differentiation method when the analytical form of the element stiffness matrix is available.

For shape design variables, design perturbation involves both the size of the perturbation and its distribution over the domain. For the choice of perturbation size, similar considerations as discussed for global finite differences play a role. Unfortunately, the semi-analytical formulation may be extremely sensitive with respect to this choice. We do not explain this aspect in detail, but we only note here that this drawback may negate all advantages of a semi-analytical formulation and motivates modifications to the semi-analytical method.

The method of calculating sensitivity using equation (13) is called the *direct method* in which the derivatives of the state variables, $\mathrm{d}\boldsymbol{q}/\mathrm{d}x$, are calculated first, ~~and, then,~~ the derivative of the performance measure is calculated using the chain rule of differentiation. There is another way of calculating the sensitivity of performance measure without explicitly calculating the sensitivity of state variable. Note that the term $(\partial\psi/\partial\boldsymbol{q})\boldsymbol{K}^{-1}$ is independent of design in equation (13). Thus, we can solve for this term first by defining the following adjoint equation:

$$\boldsymbol{K}\boldsymbol{\lambda} = \frac{\partial\psi}{\partial\boldsymbol{q}} \tag{14}$$

where the symmetric property of stiffness matrix, $\boldsymbol{K} = \boldsymbol{K}^{\mathrm{T}}$, is used. Note that the above adjoint equation uses the same stiffness matrix ~~with~~ the structural equation. The only difference is the right-hand side term $\partial\psi/\partial\boldsymbol{q}$, which is called the adjoint load. The adjoint solution, $\boldsymbol{\lambda}$, is not dependent on design but the performance measure. Thus, the adjoint solution is required per performance measure. Once the adjoint solution is available, the sensitivity can be calculated from equation (13) as

$$\frac{\mathrm{d}\psi}{\mathrm{d}x} = \frac{\partial\psi}{\partial x} + \boldsymbol{\lambda}^{\mathrm{T}}\left(\frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q}\right) \tag{15}$$

Calculating the sensitivity using equation (15) is called the *adjoint method*. The direct method in equation (13) is more closely related to the design variables, whereas the adjoint method is more closely related to the performance measure. When the number of design variables is much greater than that of performance measures, the adjoint method has a computational advantage.

2.3 Continuum method

In the continuum method, ~~the~~ sensitivity is obtained by differentiating the continuum equations that govern structural behavior. Most commonly, these consist of partial differential equations or an integral form, for example, derived from the principle of virtual work. The differentiation leads to a set of continuum sensitivity equations that are then solved numerically, usually but not necessarily, with the same discretization as used for the original structural response. For shape sensitivities, the two main approaches for continuum derivatives are the material derivative approach (Choi and Kim, 2004) and the control volume approach (Arora, Lee and Cardoso, 1992). Profound mathematical proofs are available regarding the existence and uniqueness of the sensitivity (Haug, Choi and Komkov, 1986).

In the continuum approach, ~~the~~ sensitivity can be understood as a variation of a function. Let us consider that the design variable x is perturbed to $x + \tau\eta$ in which τ is the scalar that measures the perturbation size and η is the direction of design change. For simplicity, it is assumed that the structural design variable x does not affect the domain. The variation of field response $\boldsymbol{u}$ with respect to x can then be defined as

$$\boldsymbol{u}' \equiv \lim_{\tau\to 0}\left\{\frac{\boldsymbol{u}(x+\tau\eta) - \boldsymbol{u}(x)}{\tau}\right\} = \left.\frac{\partial\boldsymbol{u}}{\partial\tau}\right|_{\tau=0}\eta \tag{16}$$

Since the direction of design change η can be arbitrary, equation (16) must be linear with respect to η and the coefficient of η is called the sensitivity of field response $\boldsymbol{u}$, which is equivalent to the derivative in the context of other approaches.

Using equation (16), the principle of virtual work in equation (4) can be differentiated to obtain the following continuum sensitivity equation:

$$\iint_V \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{\varepsilon}'\,\mathrm{d}V = \iint_V \delta\boldsymbol{u}^{\mathrm{T}}\boldsymbol{b}'\,\mathrm{d}V + \int_A \delta\boldsymbol{u}^{\mathrm{T}}\boldsymbol{h}'\,\mathrm{d}A - \iint_V \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{C}'\boldsymbol{\varepsilon}\,\mathrm{d}V \tag{17}$$

for all $\delta\boldsymbol{u}$ that belong to the space of kinematically admissible displacements. The left-hand side of equation (17) is the same as that of equation (4) if $\boldsymbol{u}$ is replaced by $\boldsymbol{u}'$. The right-hand side of equation (17) defines a pseudo-load (or fictitious load), which explicitly depends on the design. Thus, solving the sensitivity equation is the same as solving the original structural equilibrium equation with different load terms. The major advantage of the continuum approach is that the sensitivity formulation is independent of discrete model and numerical schemes. Once the continuum sensitivity equation is obtained, it can be discretized in the same manner as the original analysis equations in order to obtain a system of matrix equations similar to equation (5).

When the design variables affect the shape of the domain, the differentiation of the equations of equilibrium is much more complicated because the integral domain depends on the design. Interested readers are referred to Choi and Kim (2004) for the material derivative approach and Arora, Lee and Cardoso (1992) or Phelan and Haber (1989) for the control volume approach.

One frequently asked question is: "Are the discrete and continuum methods equivalent?" This comparison is only possible when the two methods use the same finite element equation. In the case of non-shape designs, it is well known that the two methods yield equivalent sensitivity results if the same discretization (i.e., the same finite element method) is used. However, in the case of shape design variables, the two methods can yield different results based on how the domain is changed according to the shape design variables. Choi and Twu (1989) showed that both methods are equivalent when they have (i) ~~same~~ discretization (shape function), (ii) exact integration (not numerical integration), (iii) analytical (not numerical) finite element solutions, and (iv) linear velocity field and consistent mesh perturbation. It was shown that the sensitivity results of both methods are different when quadratic and cubic design velocity fields are used. In the practical point of view, the requirements in (i) and (iv) are reasonable, but the requirements in (ii) and (iii) can be significant. Akbari and Kim (20~~09~~) further showed that when a linear design velocity is used, the two methods are equivalent without requiring (ii) and (iii). Thus, in many practical cases, both methods may yield the same sensitivity results.

2.4 Automatic differentiation

Even if the finite element programs are composed of many complicated subroutines and functions, they are basically a collection of elementary functions. Automatic differentiation method defines the partial derivatives of these elementary functions from which the derivatives of complicated subroutines and functions are computed using propagation and the chain rule of differentiation. The arguments of elementary functions can be either one or two. Without loss of generality, let us assume that an elementary function has two arguments, defined as

$$a = f_{\text{elem}}(z_i, z_j) \tag{18}$$

where $f_{\text{elem}}(\bullet, \bullet)$ represents ($++$, $\sin(\bullet)$, $\ldots$) operators for ~~the~~ single argument and ($+$, $-$, $*$,/, $\ldots$) operators for ~~the~~ double arguments.

In the direct differentiation method, the derivative of equation (18) can be defined as

$$\frac{\partial a}{\partial x} = \frac{\partial f_{\text{elem}}}{\partial z_i}\frac{\partial z_i}{\partial x} + \frac{\partial f_{\text{elem}}}{\partial z_j}\frac{\partial z_j}{\partial x} \tag{19}$$

This derivative can propagate through complicated functions and subroutines using the chain rule of differentiation. This propagation eventually produces the derivative of the structural response.

In the reverse mode, which corresponds to the adjoint method in the previous sections, the derivatives are computed backward. Due to the reverse procedure, this approach requires saving entire function evaluation, which also requires a significant amount of memory.

Computer programs that calculate the derivatives of output of other computer programs are now available and are applicable to ever-growing programs. The largest program that we found had about 800 000 lines (Bischof *et al.*, 2003). Both first- and higher-order derivatives can be obtained. Application of automatic differentiation to coupled systems is discussed by Wujek and Renaud (1998). This approach was initially called automatic differentiation, but, after a while, it was realized that human intervention in the process is required in many cases in order to obtain a reasonably efficient code. Thus, it is often referred to as computational differentiation.

In order to achieve better performance, automatic differentiation may only be used with certain parts of the program. This consequently leads to higher labor investment ~~as~~ compared to automatic differentiation of the entire program.

There are several automatic differentiation tools widely available today, notably ADIFOR (Automatic Differentiation of Fortran (Bischof *et al.*, 1996)) and ADOL-C for C/C++ programs (Griewank, Juedes and Utke, 1996). In terms of implementation, there are two basic approaches to automatic differentiation – source code transformation and operator overloading. Source code transformation can be viewed as a pre-compiler that adds code for computing the derivatives. Operator overloading is available in modern computer languages, such as C++ and Fortran 90, which provide the ability to redefine the meaning of elementary operators (such as multiplication) for various classes of variables. By defining new variable types that have gradient objects associated with them and overloading the elementary operators to also produce gradients, the code can be transformed without increasing its size substantially. ADOL-C and ADOL-F (Shiriaev and Griewank, 1996) are examples of operator-overloading tools for automatic differentiation.

3 EXAMPLES

3.1 Cantilevered beam

Consider a cantilevered beam shown in Figure 4 with $E = 2.9 \times 10^4$ ksi. The cross-sectional dimensions are given as $w = 2.25$ in and $h = 4.47$ in. The sensitivity of the tip deflection is to be determined with respect to the height of the cross section. The discrete method is used to calculate the sensitivity. In this simple example, the analytical expression of the tip deflection is available as

$$v_{\text{tip}} = \frac{4FL^3}{Ewh^3} \tag{20}$$

By differentiating the above expression with respect to the height, the exact sensitivity expression can be obtained as

$$\begin{aligned}\frac{\mathrm{d}v_{\text{tip}}}{\mathrm{d}h} &= -\frac{12FL^3}{Ewh^4} = -\frac{12 \times 2000 \times 100^3}{2.9 \times 10^7 \times 2.25 \times 4.47^4} \\ &= -0.921\end{aligned} \tag{21}$$

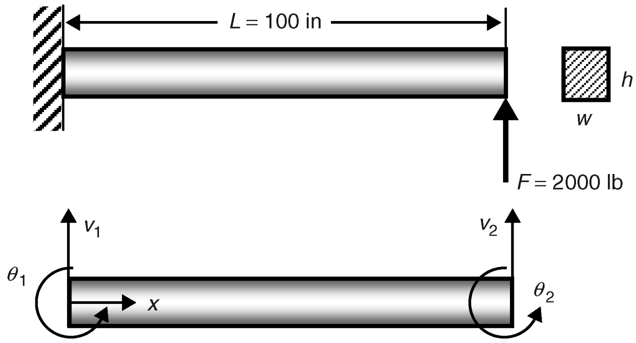


Figure 4. Cantilevered beam and finite element.

Note that the negative sensitivity means that if the height of the cross section increases, the tip deflection decreases.

When one beam finite element is used to model the problem, as shown in Figure 4, the finite element matrix equation before applying boundary conditions can be written as

$$\frac{EI}{L^3}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}\begin{bmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} R_1 \\ C_1 \\ F \\ 0 \end{bmatrix} \tag{22}$$

where R_1 and C_1 are the supporting force and couple at the wall, respectively. In matrix notation, the above equation can be written as

$$\boldsymbol{K}(x)\boldsymbol{q} = \boldsymbol{F}(x) \tag{23}$$

For the cantilevered beam, the deflection and slope at the wall are fixed. In finite element analysis, these boundary conditions can be applied by deleting the first and second columns and rows. Then, the final form of finite element matrix equation becomes

$$\frac{EI}{L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{bmatrix} v_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} \tag{24}$$

The solutions of the above equation become

$$v_2 = \frac{4FL^3}{Ewh^3}, \quad \theta_2 = \frac{6FL^2}{Ewh^3} \tag{25}$$

Note that the tip deflection, v_2, is identical to that in equation (20). Thus, the solution from finite element analysis is exact in this case.

The sensitivity of nodal displacements can be obtained by differentiating the finite element matrix equation (22) with respect to the design. Consider height $x = h$ of the ~~cross-sectional dimension~~ as a design variable. The sensitivity equation is given in equation (6). In order to solve the design sensitivity equation, we need to calculate the RHS of equation (6). Since the applied load $\boldsymbol{F}$ is independent of the design, the first term $\mathrm{d}\boldsymbol{F}/\mathrm{d}x = 0$. The stiffness matrix in equation (22) depends on design through the moment of inertia $I = wh^3/12$. Thus, it can be differentiated with respect to design. After multiplying with nodal degree of freedoms (DOFs) $\boldsymbol{q}$, we have

$$\frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q} = \frac{F}{4Lh}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 4L \\ 6 \end{bmatrix} = \frac{F}{4h}\begin{bmatrix} -12 \\ -12L \\ 12 \\ 0 \end{bmatrix} \tag{26}$$

Thus, the RHS of equation (6) can be computed as

$$\frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q} = \frac{F}{4h}\begin{bmatrix} 12 \\ 12L \\ -12 \\ 0 \end{bmatrix} \tag{27}$$

Then, the sensitivity equation can be obtained as

$$\frac{EI}{L^3}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}\begin{bmatrix} \frac{\mathrm{d}v_1}{\mathrm{d}x} = 0 \\ \frac{\mathrm{d}\theta_1}{\mathrm{d}x} = 0 \\ \frac{\mathrm{d}v_2}{\mathrm{d}x} \\ \frac{\mathrm{d}\theta_2}{\mathrm{d}x} \end{bmatrix} = \frac{F}{4h}\begin{bmatrix} 12 \\ 12L \\ -12 \\ 0 \end{bmatrix} \tag{28}$$

The first two rows and columns are deleted due to zero displacement boundary conditions. Note that this part is the same as the original finite element analysis. When a displacement is fixed, the sensitivity is also zero. After applying displacement boundary conditions, we have

$$\frac{EI}{L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{bmatrix} \frac{\mathrm{d}v_2}{\mathrm{d}x} \\ \frac{\mathrm{d}\theta_2}{\mathrm{d}x} \end{bmatrix} = \frac{F}{4h}\begin{bmatrix} -12 \\ 0 \end{bmatrix} \tag{29}$$

The above equation can be solved for the unknown nodal DOFs. Now we have,

$$\frac{\mathrm{d}v_2}{\mathrm{d}x} = -\frac{12FL^3}{Ewh^4}, \quad \frac{\mathrm{d}\theta_2}{\mathrm{d}x} = -\frac{18FL^2}{Ewh^3} \tag{30}$$

Note that $\mathrm{d}v_2/\mathrm{d}x$ is the same as ~~with~~ equation (21). Thus, the sensitivity we calculated is exact. Note that in differentiating the stiffness matrix in equation (26), only the moment of inertia I was differentiated. The basic form of the matrix remains unchanged. This is because the design variable appears as a parameter in the structural equation.

In the adjoint method, the adjoint load must be calculated first. Since the performance measure is the tip deflection, $\psi = v_2$, the adjoint load becomes $\partial\psi/\partial\boldsymbol{q} = \{0\ 0\ 1\ 0\}^{\mathrm{T}}$. Thus, the adjoint load is applying a unit force at the tip. The adjoint equation, $\boldsymbol{K\lambda} = (\partial\psi/\partial\boldsymbol{q})^{\mathrm{T}}$, becomes

$$\frac{EI}{L^3}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix}\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} \bar{R}_1 \\ \bar{C}_1 \\ 1 \\ 0 \end{bmatrix} \tag{31}$$

After applying the same boundary conditions, the reduced adjoint equation becomes

$$\frac{EI}{L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\begin{bmatrix} \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{32}$$

Thus, the adjoint solution becomes

$$\lambda_3 = \frac{4L^3}{Ewh^3}, \quad \lambda_4 = \frac{6L^2}{Ewh^3} \tag{33}$$

From the sensitivity expression in equation (15), there is no explicitly dependent term; that is, $\partial\psi/\partial x = 0$. In addition, the applied load is independent of design; that is, $\mathrm{d}\boldsymbol{F}/\mathrm{d}x = 0$. Thus, the sensitivity expression becomes

$$\begin{aligned} \frac{\mathrm{d}\psi}{\mathrm{d}x} &= -\boldsymbol{\lambda}^{\mathrm{T}}\frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x}\boldsymbol{q} \\ &= -\frac{L^2}{Ewh^3}\left\{4L\ \ 6\right\}\frac{Ewh^3}{12L^3}\begin{bmatrix} 12 & -6L \\ -6L & 4L^2 \end{bmatrix}\frac{FL^2}{Ewh^3}\begin{Bmatrix} 4L \\ 6 \end{Bmatrix} \\ &= -\frac{12FL^3}{Ewh^4} \end{aligned} \tag{34}$$

Note that the result is identical to that of equation (30). It would be a good exercise to calculate the sensitivity of θ_2. In that case, the adjoint load will be $\partial\psi/\partial\boldsymbol{q} = \{0\quad 0\quad 0\quad 1\}^{\mathrm{T}}$.

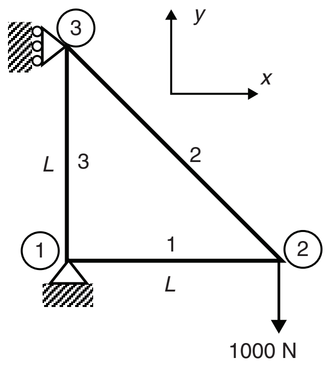


Figure 5. Three-bar truss for fully stressed design.

3.2 Three-bar truss

A three-bar truss with length $L = 1$ m and Young's modulus $E = 80$ GPa in Figure 5 is under a vertical load of $F = 1000$ N at Node 2. Design variables are the cross-sectional areas of each truss member. The current design variables are $x_1 = x_2 = x_3 = 10\ \mathrm{mm}^2$. The discrete method is used to calculate the sensitivity of the vertical displacement (v_2) at Node 2 with respect to x_2. In addition, the accuracy of the calculated sensitivity is compared with the forward finite difference sensitivity with 1.0% perturbation size.

The element table is shown below.

Element	First node i	Second node j	AE/L	l	m
1	1	2	Ex_1	1	0
2	2	3	$Ex_2/\sqrt{2}$	$-1/\sqrt{2}$	$1/\sqrt{2}$
3	1	3	Ex_3	0	1

Using the above element table, the global element equations can be obtained after applying displacement boundary conditions:

$$10^5\begin{bmatrix} 10.83 & -2.83 & 2.83 \\ -2.83 & 2.83 & -2.83 \\ 2.83 & -2.83 & 10.83 \end{bmatrix}\begin{bmatrix} u_2 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ -1000 \\ 0 \end{bmatrix} \tag{35}$$

By solving the above equation, we have the following ~~un-known~~ nodal DOFs:

$$u_2 = -1.25\ \text{mm}, \quad v_2 = -6.04\ \text{mm}, \quad v_3 = -1.25\ \text{mm} \tag{36}$$

In order to build the sensitivity equation, the RHS of equation (6) needs to be calculated. Since the applied load $\boldsymbol{F}$

is independent of the design, the first term vanishes: that is, $\mathrm{d}\boldsymbol{F}/\mathrm{d}\boldsymbol{x} = 0$. Out of three element stiffness matrices, only $\boldsymbol{k}^{(2)}$ depends on design x_2. Thus,

$$\frac{\mathrm{d}\boldsymbol{k}^{(1)}}{\mathrm{d}x_2} = \frac{\mathrm{d}\boldsymbol{k}^{(3)}}{\mathrm{d}x_2} = 0, \quad \frac{\mathrm{d}\boldsymbol{k}^{(2)}}{\mathrm{d}x_2} = \frac{E}{2L^{(2)}} \begin{array}{cc} \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} & \begin{matrix} u_2 \\ v_2 \\ u_3 \\ v_3 \end{matrix} \end{array} \tag{37}$$

These three matrices are assembled in the same way ~~with~~ the stiffness matrix and then multiplied by the vector of nodal displacements to obtain

$$\frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}x_2} - \frac{\mathrm{d}\boldsymbol{K}}{\mathrm{d}x_2}\boldsymbol{q} = -2.828 \times 10^{10} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 & -1 & 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ -.0013 \\ -.0060 \\ 0 \\ -.0013 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -10^8 \\ 10^8 \\ 10^8 \\ -10^8 \end{bmatrix} \tag{38}$$

Then, the design sensitivity equation, after applying displacement boundary conditions, becomes

$$10^5 \begin{bmatrix} 10.83 & -2.83 & 2.83 \\ -2.83 & 2.83 & -2.83 \\ 2.83 & -2.83 & 10.83 \end{bmatrix} \begin{bmatrix} \dfrac{\mathrm{d}u_2}{\mathrm{d}x_2} \\ \dfrac{\mathrm{d}v_2}{\mathrm{d}x_2} \\ \dfrac{\mathrm{d}v_3}{\mathrm{d}x_2} \end{bmatrix} = \begin{bmatrix} -10^8 \\ 10^8 \\ -10^8 \end{bmatrix} \tag{39}$$

The solution of the above sensitivity equation yields

$$\frac{\mathrm{d}u_2}{\mathrm{d}x_2} = 0, \quad \frac{\mathrm{d}v_2}{\mathrm{d}x_2} = 353.55, \quad \frac{\mathrm{d}v_3}{\mathrm{d}x_2} = 0 \tag{40}$$

Thus, the change in the cross-sectional area of member 2 will ~~only~~ change the vertical displacement of Node 2.

Let us compute the sensitivity of v_2 by using the finite difference method. The original displacements in equation (36) are saved as $v_2(x_2) = -6.036$ mm. And then, design x_2 is perturbed by 1.0%; that is, $x_2 + \Delta x_2 = 10.1\ \mathrm{mm}^2$. A new global matrix equation is produced with new design, as

$$10^5 \begin{bmatrix} 10.86 & -2.86 & 2.86 \\ -2.86 & 2.86 & -2.86 \\ 2.86 & -2.86 & 10.86 \end{bmatrix} \begin{bmatrix} u_2 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 \\ -1000 \\ 0 \end{bmatrix} \tag{41}$$

Note that the matrix is slightly different from the one in equation (35). By solving the above equation, we have the following unknown nodal DOFs:

$$u_2 = -1.25\ \mathrm{mm}, \quad v_2 = -6.00\ \mathrm{mm}, \quad v_3 = -1.25\ \mathrm{mm} \tag{42}$$

Note that u_2 and v_3 did not change, which is consistent with the zero sensitivity in equation (40). With the vertical displacement at Node 2, we have the performance at the perturbed design, $v_2(x_2 + \Delta x_2) = -6.001$ mm. From the finite difference sensitivity formula in equation (9), we have

$$\begin{aligned} \frac{\mathrm{d}v_2}{\mathrm{d}x_2} &\approx \frac{v_2(x_2 + \Delta x_2) - v_2(x_2)}{\Delta x_2} \\ &= \frac{-6.001 \times 10^{-3} + 6.306 \times 10^{-3}}{0.1 \times 10^{-5}} = 350.05 \end{aligned} \tag{43}$$

Note that the finite difference sensitivity in equation (43) is slightly different from the one in equation (40). This is due to the influence of finite perturbation size. When 0.1% perturbation is used, the finite difference sensitivity becomes 353.2, which is much closer to the one in equation (40). However, as we can see in equations (36) and (41), the difference in stiffness matrix is small. Thus, it is required to maintain high accuracy in matrix solution in order to have a small perturbation size.

3.3 Road arm model

A road arm structure, as shown in Figure 6, transfers a force and torque from a road wheel to a suspension unit for a combat vehicle. The road arm model is discretized with 4365 DOFs. The road arm is made of steel with $E = 206$ GPa, and $\nu = 0.3$. At the center of the right hole, a vertical force of 3736 N and a torque of 44 516 N m are applied, while the displacement on the left hole is fixed. As was illustrated in Figure 6, the stress concentration appears in the left corner of the road arm. If the highest stress level in the left corner is considered as a reference value, then the dimension of the right corner cross section can be reduced because this region has a large ~~amount of~~ safety margin.

Since two holes are connected to the road wheel and torsion bar, the dimension and geometry of the holes are fixed.

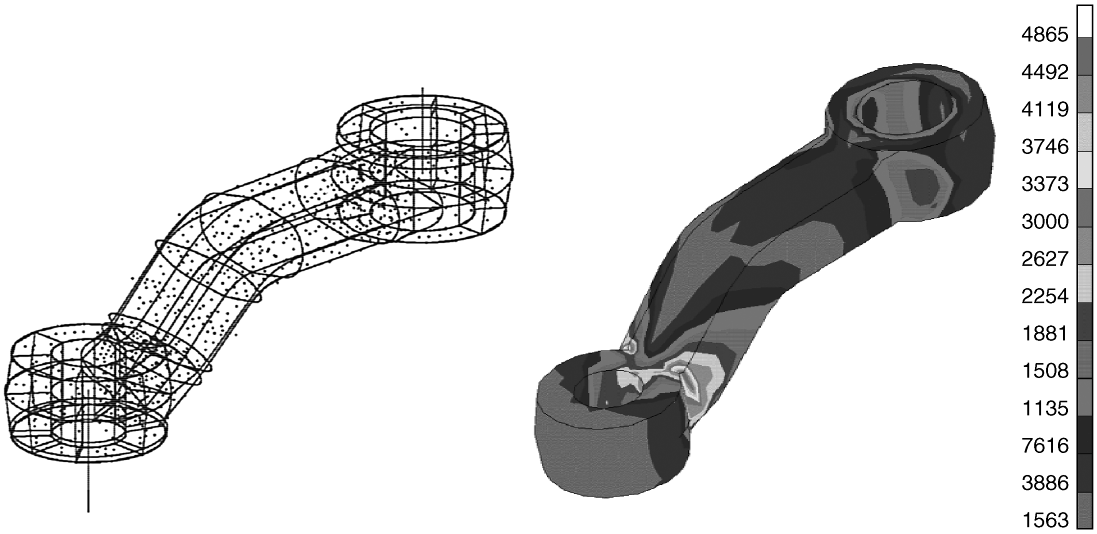


Figure 6. Discrete road arm model and simulation results (stress).

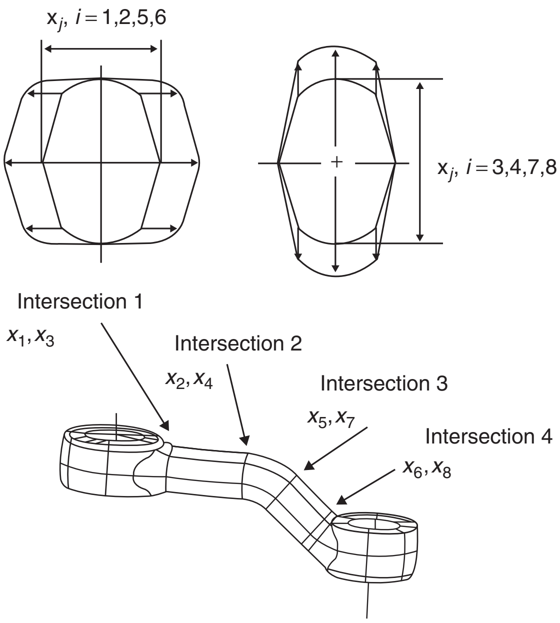


Figure 7. Design parameters for the road arm model.

The design goal is to determine the dimension of the cross sections of the arm. The heights and widths of four sections are selected as design parameters (see Figure 7). Thus, a total of eight design variables are considered in this example.

As the design variables vary, the boundary surface of the structure changes. At the same time, the discrete model also needs to be moved according to the design variable's change. Even if the design variable changes the boundary surface, it is recommended to move the interior nodes too. Otherwise, the accuracy of the perturbed model may deteriorate. The relation between a design variable and the motion of each node is denoted by the *design velocity field*. Figure 8 shows the design velocity field for two different design variables x_2 and x_4, respectively. The arrows denote the magnitude and direction of nodal movement according to the corresponding design variable's change.

For a given design variable, the design sensitivity coefficients of various performance measures can be calculated using the design velocity field. Table 1 shows the design sensitivity coefficients compared ~~with~~ the global finite difference

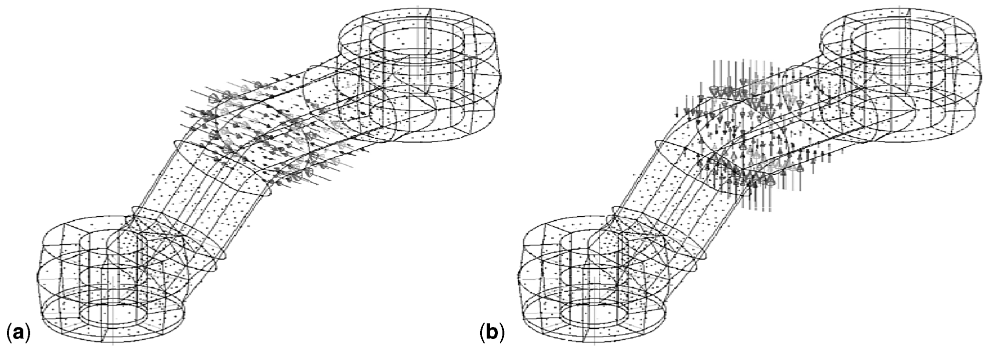


Figure 8. Design velocity field vectors: (a) design x_2; (b) design x_4.

Table 1. Design sensitivity results compared with finite difference method.

Design	Performance		Finite difference	Continuum method	Ratio (%)
	u_{x4}	7.985E − 04	6.640E − 11	6.640E − 11	100.00
	u_{y8}	−1.740E − 04	−1.505E − 11	−1.505E − 11	100.00
x_1	σ_{x10}	1.384E + 01	−3.251E − 06	−3.260E − 06	99.72
	σ_{y14}	1.255E + 01	−1.408E − 05	−1.408E − 05	100.00
	σ_{z18}	−3.113E + 00	−1.798E − 06	−1.800E − 06	99.85
	Volume	4.689E + 02	−4.538E − 05	−4.538E − 05	100.00
	u_{x4}	7.985E − 04	2.136E − 10	2.136E − 10	100.00
	u_{y8}	−1.740E − 04	−7.769E − 11	−7.769E − 11	100.00
x_5	σ_{x10}	1.384E + 01	4.835E − 07	4.839E − 07	99.91
	σ_{y14}	1.255E + 01	−4.811E − 06	−4.811E − 06	100.00
	σ_{z18}	−3.113E + 00	−4.004E − 06	−4.006E − 06	99.95
	Volume	4.689E + 02	−7.249E − 06	−7.249E − 06	100.00

results. Displacement, stress, and volume of the structure are considered as performance measures. For example, u_{x4} represents the x-directional displacement at node 4 and σ_{x10} is the x-directional normal stress at element 10. Good agreements are observed between the continuum and global finite difference sensitivities. However, several trial and errors are required to find appropriate perturbation size, $\Delta x = 10^{-4}$, for the global finite difference method. In addition, the structural analysis requires very accurate numerical calculation in order to capture ~~very~~ small changes in the responses. This sensitivity information will be provided to the design optimization algorithm to obtain the optimum design for given constraints.

NOMENCLATURE

A	cross-sectional area
$\boldsymbol{b}$	body force vector
$\boldsymbol{C}$	elasticity matrix
E	Young's modulus
$f_{\text{elem}}(\bullet, \bullet)$	elementary operator with single/double arguments
$\boldsymbol{F}(x)$	load vector
$\boldsymbol{h}$	surface traction
$\boldsymbol{K}(x)$	stiffness matrix
L	length of a beam
$\boldsymbol{p}$	pseudo-load vector
$\boldsymbol{q}$	vector of nodal DOFs
r	radius of a circular cross section
$\boldsymbol{u}$	continuous displacement field
x	design variable
v_i	nodal deflection at node i of a beam element
W	weight
ε	strain
λ	adjoint variable
ρ	density
θ_i	nodal rotation at node i of a beam element
σ	stress
ψ	performance measure

REFERENCES

Akbari, J. and Kim, N.H. (2009) Shape sensitivity analysis with design-dependent loadings: Equivalence between continuum and discrete derivatives. *Struct. Multidiscip. Optim.*, ~~published online~~ doi: 10.1007 ≠ 500158-00158-009-0374-4.

Akgün, M.A., Garcelon, J.H. and Haftka, R.T. (2001) Fast exact linear and nonlinear structural reanalysis and the Sherman–Morrison–Woodbury formulas. *Int. J. Numer. Methods Eng.*, **50**(7), 1587–1606.

Arora, J.S., Lee, T.-H. and Cardoso J.B. (1992) Structural shape sensitivity analysis: relationship between material derivative and control volume approaches. *AIAA J.*, **30**(6), 1638–1648.

Barthelemy, B. and Haftka, R.T. (1988) Accuracy analysis of the semi-analytical method for shape sensitivity. Proceedings of AIAA/ASME/ASCE/ASC 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, 18–20 April, pp. 88–2284.

Bischof, C.H., Bücker, H.M., Lang, B.*et al.* (2003) Extending the functionality of the general-purpose finite element package SEPRAN by automatic differentiation. *Int. J. Numer. Methods Eng.*, **58**, 2225–2238.

Bischof, C.H., Carle, A., Khademi, P. and Mauer, A. (1996) The ADIFOR 2.0 system for the automatic differentiation of Fortran 77 programs. *IEEE Comput. Sci. Eng.*, **3**, 18–32.

Choi, K.K. and Kim, N.H. (2004) *Structural Sensitivity Analysis and Optimization 1: Linear Systems*, Springer-Verlag, New York.

Choi, K.K. and Twu, S.-L. (1989) Equivalence of continuum and discrete methods of shape design sensitivity analysis. *AIAA J.*, **27**(10), 1418–1424.

Griewank, A., Juedes, D. and Utke, J. (1996) ADOL-C, a package for the automatic differentiation of algorithms written in C/C++. *TOMS*, **22**(2), 131–167.

Haug, E.J., Choi, K.K. and Komkov, V. (1986) *Design Sensitivity Analysis of Structural Systems*, Academic Press, New York.

Olhoff, N., Rasmussen, J. and Lund, E. (1993) A method of "exact" numerical differentiation for error elimination in finite-element-based semi-analytical shape sensitivity analyses. *Mech. Struct. Mach.*, **21**, 1–66.

Oral, S. (1996) An improved semianalytical method for sensitivity analysis. *Struct. Optim.*, **11**, 67–69.

Phelan, D.G. and Haber, R.B. (1989) Sensitivity analysis of linear elastic systems using domain parametrization and a mixed mutual energy principle. *Comput. Methods Appl. Mech. Eng.*, **77**, 31–59.

Shiriaev, D. and Griewank, A. (1996) ADOL-F: automatic differentiation of Fortran codes. *Comput. Diff. Tech. Appl. Tools*, **1**, 375–384.

Wujek, B.A. and Renaud, J.E. (1998) Automatic differentiation for more efficient system analysis and optimization. *Eng. Optim.*, **31**, 101–139.

Yoon, B.G. and Belegundu, A.D. (1988) Iterative methods for design sensitivity analysis. *AIAA J.*, **26**(11), 1413–1415.