10th World Congress on
Computational Mechanics
8-13 July 2012 • São Paulo • Brazil

NUMERICAL SENSITIVITY COMPUTATION FOR DISCONTINUOUS GRADIENT-ONLY OPTIMIZATION PROBLEMS USING THE COMPLEX-STEP METHOD

D. N. Wilke[1], S. Kok[2]

[1] Department of Mechanical and Aeronautical Engineering, University of Pretoria, South Africa (nico.wilke@up.ac.za)
[2] Modelling and Digital Science, The Council for Scientific and Industrial Research, South Africa

Abstract. *This study considers the numerical sensitivity calculation for discontinuous gradient-only optimization problems using the complex-step method. The complex-step method was initially introduced to differentiate analytical functions in the late 1960s, and is based on a Taylor series expansion using a pure imaginary step. The complex-step method is not subject to subtraction errors as with finite difference approaches when computing first order sensitivities and therefore allows for much smaller step sizes that ultimately yields accurate sensitivities. This study investigates the applicability of the complex-step method to numerically compute first order sensitivity information for discontinuous optimization problems. An attractive feature of the complex-step approach is that no real difference step is taken as with conventional finite difference approaches, since conventional finite differences are problematic when real steps are taken over a discontinuity. We highlight the benefits and disadvantages of the complex-step method in the context of discontinuous gradient-only optimization problems that result from numerically approximated (partial) differential equations.*

Gradient-only optimization is a recently proposed alternative to mathematical programming for solving discontinuous optimization problems. Gradient-only optimization was initially proposed to solve shape optimization problems that utilise remeshing (i.e. the mesh topology is allowed to change) between design updates. Here, changes in mesh topology result in abrupt changes in the discretization error of the computed response. These abrupt changes in turn manifests as discontinuities in the numerically computed objective and constraint functions of an optimization problem. These discontinuities are in particular problematic when they manifest as local minima. Note that these potential issues are not limited to problems in shape optimization but may be present whenever (partial) differential equations are approximated numerically with non-constant discretization methods e.g. remeshing of spatial domains or automatic time stepping in temporal domains.

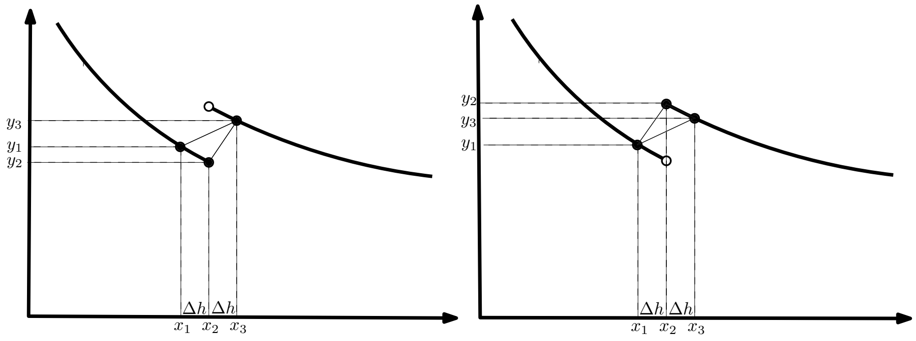
Keywords: *Complex-step derivative, Discontinuous function, Gradient-only optimization.*

1. INTRODUCTION

Numerical sensitivity computation for smooth continuous functions is well established. A number of well-known strategies are available which include (semi)-analytical sensitivity computations using direct and adjoint approaches, numerical finite difference techniques that are prone to cancellation errors for small step sizes and forward and reverse mode automatic differentiation. However, also available are the lesser known complex variable sensitivity methods that were proposed in the late 1960's. The complex-step method [8] and Fourier differentiation [7] are of particular interest. The latter has significant advantages for the computation of higher order derivatives, but it requires steps on the complex plain i.e. steps that have both real and complex components. In turn, the complex step method only requires steps along the imaginary axis. Hence, steps only have a complex component and no real component to compute first order sensitivity information. This is an important distinction that we will explore further in detail. Since gradient-only optimization only requires first order sensitivity information we will focus on the complex step method, which is well suited to compute accurate first order sensitivity information. We note that for the computation of higher order sensitivities, the complex-step method is also prone to cancellation errors (see [5]).

In particular, we consider the computation of first order sensitivity information when dealing with discontinuous objective functions, i.e. functions that are not differentiable at the discontinuities. Non-differentiable objective and constraint functions are not new to engineering optimization problems, with [13] considering continuous functions with discontinuous gradients i.e. C^0 continuity. Since non-differentiable functions are not everywhere differentiable, the concept of subgradients was introduced to allow the gradient field to be defined everywhere. Recently, Wilke et al. [18] proposed gradient-only optimization as an alternative approach to mathematical programming to solve discontinuous optimization problems, which are non-differentiable and *not* continuous. The gradient field is defined everywhere via *associated* gradients, which allows sensitivity information to be defined at a discontinuity. The *associated* gradient is given by the *Calculus* gradient when the gradient exists. Where the *Calculus* gradient does not exist, it is defined by only the left or right sided limit as detailed in [18]. In this study we highlight the connection between the complex-step derivative and the *associated* derivative.

We note that gradient-only optimization is not a general strategy for solving discontinuous optimization problems. For the purposes of this discussion we distinguish between two types of discontinuities that may be present in objective and constraint functions. Firstly, discontinuities can be *physical* in a computed response e.g. contact problems or shock-waves. Secondly, discontinuities can be *non-physical* but are present due to errors in a computational strategy. For example, it is well known that abrupt changes in the discretization error of a numerical strategy e.g. remeshing in finite element based shape optimization results in discontinuities (see [2]. Gradient-only optimization is concerned with solving the latter type of optimization problems and therefore we will only consider *non-physical* discontinuities from hereon. In particular we only address problems when changes in candidate designs require changes in the temporal and/or spatial domains during optimization e.g. shape optimization. Here, an analyses is conducted for each candidate design and consequently the objective func-



(a) (b)

Figure 1. Finite difference step over a (a) lower semi-continuous and (b) upper semi-continuous function with an inconsistent step discontinuity.

tion is constructed from multiple independent analyses. The relation of the discretization error between analyses influences the differentiability of the objective function. This is in stark contrast to once-off analyses where temporal and/or spatial domains may also change, but for a single analysis this only influences the solution quality of the current analysis.

Instead of finding the minimum of these *non-physical* discontinuous optimization problems, gradient-only optimization aims to find positive projective points of these problems. Positive projection points are characterized by a positive directional derivative at all points in a small ball around such a point. The direction vector is defined by the point in the ball where the directional derivative is to be computed and the positive projection point itself [18].

After the proposal of the complex-step method by Lyness and Moler [8], it remained unexploited for almost 20 years until the revival paper was written by Squire and Trapp [14]. Since then it has been used in a number of computational fluid dynamics solvers [10,15], non-linear structural analysis [4], pseudo-spectral simulations [1], dynamical systems [3], fatigue analysis [17] and shape sensitivity [16].

As we will now demonstrate, conventional finite differences are severely hampered when the finite difference perturbation occurs within the vicinity of a discontinuity. This occurs when the discretization topology (i.e. number of elements and/or nodal connectivity of elements) changes abruptly during the finite difference perturbations, which results in the computation of inconsistent sensitivity information [12]. To allow for the computation of consistent sensitivity information using non-constant discretization strategies, usually requires the non-constant discretization strategy to be modified to allow for only smooth and continuous variations of discretizations during the sensitivity computations.

2. SENSITIVITIES OF DISCONTINUOUS FUNCTIONS

Consider the lower and upper semi-continuous functions depicted in Figures 1(a) and (b), respectively. Since an n-dimensional gradient vector is defined by it's n 1-dimensional partial derivatives, we will restrict ourselves to a 1-dimensional discussion without loss of generality.

It is clear that the derivative is not defined at the discontinuity. However, the associated derivative [18] is defined everywhere and it is given by the left hand limit for lower semi-continuous functions and by the right hand limit for upper semi-continuous functions at the discontinuity. The benefit of the complex-step derivative is that it computes the associated derivative and allows the computation of sensitivity information even at a discontinuity. As we will consider in detail, the complex-step method achieves this by only taking an imaginary step without the need to take a real step over the discontinuity.

In turn, conventional finite difference schemes requires a real step to be taken, with undesired consequences. Consider the following finite difference computations on the lower semicontinuous function depicted Figure1 (a): forward (FD), backward (BD) and central difference (CD) schemes computes the derivative as:

$$\begin{array}{lcl} \left(\frac{df}{dx}\right)_{FD} & \approx & \frac{y_3-y_2}{x_3-x_2} > 0, \\ \left(\frac{df}{dx}\right)_{BD} & \approx & \frac{y_2-y_1}{x_2-x_1} < 0, \\ \left(\frac{df}{dx}\right)_{CD} & \approx & \frac{y_3-y_1}{x_3-x_1} > 0, \end{array} \tag{1}$$

whereas the upper semicontinuous function in Figure1 (b) results in the following:

$$\begin{array}{lcl} \left(\frac{df}{dx}\right)_{FD} & \approx & \frac{y_3-y_2}{x_3-x_2} < 0, \\ \left(\frac{df}{dx}\right)_{BD} & \approx & \frac{y_2-y_1}{x_2-x_1} > 0, \\ \left(\frac{df}{dx}\right)_{CD} & \approx & \frac{y_3-y_1}{x_3-x_1} > 0. \end{array} \tag{2}$$

It is clear that finite differences in the context of discontinuous functions are severely problematic, with inconsistencies not only in the magnitude but also the direction (sign) of the computed sensitivities.

In contrast, the complex-step method avoids these problems. Consider the complex Taylor series expansion of an analytic function $f(x)$ using a complex step $i\Delta h$,

$$f(x + i\Delta h) = f(x) + i\Delta h f'(x) - \Delta h^2 \frac{f''(x)}{2} + \text{higher order terms}, \tag{3}$$

By equating the imaginary parts of both sides of the equation, the complex-step derivative approximation $\frac{Im[f(x+i\Delta h)]}{\Delta h}$ is obtained as a second order accurate approximation to $f'(x)$.

The advantage of the complex-step method is that only a complex step $i\Delta h$ on the imaginary axis is required, as opposed to a step along the real axis when using conventional finite differences. Hence, even at the discontinuity the derivative would be computed at that point as if the function was smooth in the vicinity of that point. Therefore, derivative information can always be computed. Similarly, derivative information can also always be computed when using (semi)-analytical or automatic differentiation strategies. The computed sensitivities are the *associated* derivative [18] that can be used for gradient-only optimization.

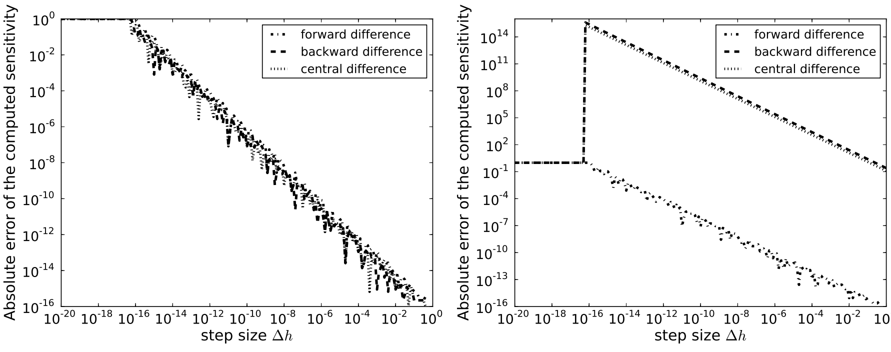
To demonstrate our arguments, consider the following simple piece-wise linear step discontinuous function:

$$f(x) = \left\{ \begin{array}{ll} x < 1: & -2x - 0.5 \\ x \geq 1: & -2x \end{array} \right. . \tag{4}$$

Mathematically, the derivative is not defined at $x = 1$. However, the *associated* derivative of this function is continuous and -2 everywhere, including at $x = 1$. Computing the derivative with the complex step method yields exactly -2 everywhere, including $x = 1$, allowing a full field computation of the derivative of a discontinuous function.

The truncation error is 0 for all finite difference schemes on each section of the piecewise linear function. The choice for a piecewise linear function allows us to isolate the error due to the discontinuity, and not be influenced by rounding errors.

Consider the results presented in Figure 2. The absolute round-off error as a function of step size is presented in Figure 2 (a) for the linear function $-2x$, using forward, backward and central difference schemes. The combined round-off and discontinuity error for the function given in (4) is presented in Figure 2 (b). The derivative at $x = 1$ was approximated using the forward, backward and central difference schemes. The step sizes in both figures are varied between 10^0 and 10^{-20}. The results of the complex-step method is not presented as it exactly recovers a sensitivity of -2, everywhere.



(a) (b)

Figure 2. Finite difference computations for the (a) linear function $f(x) = -2x$ and (b) linear function with discontinuity as given in (4).

The behaviour of the difference schemes corresponds to the anticipated behaviour as described in (2). As expected the discontinuity error dominates and increases as the step size is reduced. It is easy to envisage a simple modification to finite difference schemes from the observation above that may be able to overcome the associated problems. For example, when the signs amongst the three computed derivatives differ, a simple modification may be to choose the derivative that is the most negative when considering minimization problems as given in (1) and (2). However, when considering multidimensional problems the discontinuities may not be isolated as is the case in Figure 1, with forward and backward steps occurring over different discontinuities.

Unfortunately, the complex-step method only allows for direct or forward mode computations of sensitivities. Hence, when dealing with a large number of design variables and only a few functions the benefits of adjoint or reverse mode sensitivities cannot be exploited [11]. In addition, complex number calculations requires additional memory and computational resources. Fortunately, modifications to programs mostly require only the accommodation of complex variables, overloading a few functions and reviewing some branching statements. The authors found the guidelines presented by [9] more than adequate for all examples presented in this paper.

3. SENSITIVITIES OF VARYING DISCRETIZATIONS

As pointed consistent sensitivity information needs to be computed when considering objective functions that rely on solutions of partial differential equations [12], which usually requires a modification to non-constant discretization strategies to avoid any inconsistencies during the sensitivity computations.

As a practical example consider a linear elastic finite element based shape optimization problem, where the objective function $\mathcal{F}(\boldsymbol{x})$ is an explicit function of the nodal displacements $\boldsymbol{u}$. The nodal displacements $\boldsymbol{u}(\Lambda)$ in turn is a function of the discretization Λ (computational mesh) defined on the domain Ω with boundary $\partial\Omega$. The computational mesh

$$\Lambda \in \{\boldsymbol{\mathcal{X}} = (\boldsymbol{X}_i)_{i=1,\dots,nn}; T = (T_j^k)_{j=1,\dots,ne;k=1,\dots,nv}\}, \tag{5}$$

describes the position $\boldsymbol{\mathcal{X}} \in \mathbb{R}^3$ of the nn nodes, and gives for each computational element $j = 1, \dots, ne$ the set $T_j^{k=1,\dots,nv}$ of its' nv vertices [6]. For now we limit the nodal positions to two dimensions $\boldsymbol{\mathcal{X}} \in \mathbb{R}^2$. The computational mesh $\Lambda(\boldsymbol{x})$ in turn is a function of the design variables $\boldsymbol{x}$, which controls the discretized geometrical domain.

Following the usual finite element discretization of the linear elastic solid mechanics boundary value problem we obtain

$$\boldsymbol{K}\boldsymbol{u} = \boldsymbol{f}, \tag{6}$$

where $\boldsymbol{K}$ represents the assembled structural stiffness matrix and $\boldsymbol{f}$ the consistent structural loads. The system in Eq. (6) is partitioned along the unknown displacements ($\boldsymbol{u}_{\text{f}}$) and the prescribed displacement ($\boldsymbol{u}_{\text{p}}$), i.e.

$$\boldsymbol{K}\boldsymbol{u} = \begin{bmatrix} \boldsymbol{K}_{\text{ff}} & \boldsymbol{K}_{\text{fp}} \\ \boldsymbol{K}_{\text{pf}} & \boldsymbol{K}_{\text{pp}} \end{bmatrix} \begin{Bmatrix} \boldsymbol{u}_{\text{f}} \\ \boldsymbol{u}_{\text{p}} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{f}_{\text{f}} \\ \boldsymbol{f}_{\text{p}} \end{Bmatrix}, \tag{7}$$

where $\boldsymbol{f}_{\text{f}}$ represents the prescribed forces and $\boldsymbol{f}_{\text{p}}$ the reactions at the nodes with prescribed displacements. The unknown displacements ($\boldsymbol{u}_{\text{f}}$) are obtained from

$$\boldsymbol{K}_{\text{ff}}\boldsymbol{u}_{\text{f}} = \boldsymbol{f}_{\text{f}} - \boldsymbol{K}_{\text{fp}}\boldsymbol{u}_{\text{p}}. \tag{8}$$

Recall that the objective function $\mathcal{F}(\boldsymbol{x})$ is an explicit function of the nodal displacements $\boldsymbol{u}$. Using gradient based optimisation algorithms, we therefore require the sensitivity of the structural response $\boldsymbol{u}$ w.r.t. the design variables (control variables) $\boldsymbol{x}$. In general, the stiffness partition matrices $\boldsymbol{K}_{\text{ff}}$ and $\boldsymbol{K}_{\text{fp}}$, the nodal displacement vector $\boldsymbol{u}_{\text{f}}$ and the load vector $\boldsymbol{f}_{\text{f}}$ in Eq. (8) depend on the design variables $\boldsymbol{x}$, i.e. $\boldsymbol{K}_{\text{ff}}(\boldsymbol{x})\boldsymbol{u}_{\text{f}}(\boldsymbol{x}) = \boldsymbol{f}_{\text{f}}(\boldsymbol{x}) - \boldsymbol{K}_{\text{fp}}(\boldsymbol{x})\boldsymbol{u}_{\text{p}}(\boldsymbol{x})$.

The analytical gradient $\frac{d\boldsymbol{u}_{\text{f}}}{d\boldsymbol{x}}$ is obtained by differentiating Eq. (8) w.r.t. the control variables $\boldsymbol{x}$, i.e.

$$\boldsymbol{K}_{\text{ff}}\frac{d\boldsymbol{u}_{\text{f}}}{d\boldsymbol{x}} = \frac{d\boldsymbol{f}_{\text{f}}}{d\boldsymbol{x}} - \frac{d\boldsymbol{K}_{\text{fp}}}{d\boldsymbol{x}}\boldsymbol{u}_{\text{p}} - \boldsymbol{K}_{\text{fp}}\frac{d\boldsymbol{u}_{\text{p}}}{d\boldsymbol{x}} - \frac{d\boldsymbol{K}_{\text{ff}}}{d\boldsymbol{x}}\boldsymbol{u}_{\text{f}}. \tag{9}$$

In this study the load vector $\boldsymbol{f}_{\text{f}}$ is assumed to be independent of the control variables $\boldsymbol{x}$, hence $\frac{d\boldsymbol{f}_{\text{f}}}{d\boldsymbol{x}} = \boldsymbol{0}$. For Dirichlet boundary conditions, $\boldsymbol{u}_{\text{p}} = \boldsymbol{0}$, and Eq. (9) reduces to

$$\boldsymbol{K}_{\text{ff}}\frac{d\boldsymbol{u}_{\text{f}}}{d\boldsymbol{x}} = -\frac{d\boldsymbol{K}_{\text{ff}}}{d\boldsymbol{x}}\boldsymbol{u}_{\text{f}}. \tag{10}$$

Eq. (10) is solved to obtain $\frac{d\boldsymbol{u}_{\mathrm{f}}}{d\boldsymbol{x}}$, using the factored stiffness matrix $\boldsymbol{K}_{\mathrm{ff}}$, available from the primary analysis when solving Eq. (8). The unknown $\frac{d\boldsymbol{K}_{\mathrm{ff}}}{d\boldsymbol{x}}$ is computed from

$$\frac{d\boldsymbol{K}_{\mathrm{ff}}}{d\boldsymbol{x}} = \frac{d\boldsymbol{K}_{\mathrm{ff}}}{d\boldsymbol{\mathcal{X}}}\frac{d\boldsymbol{\mathcal{X}}}{d\boldsymbol{x}}, \tag{11}$$

where $\frac{d\boldsymbol{K}_{\mathrm{ff}}}{d\boldsymbol{\mathcal{X}}}$ is obtained by differentiating the stiffness matrix analytically with respect to the nodal coordinates $\boldsymbol{\mathcal{X}}$. This is done on the element level and then assembled into the global system.

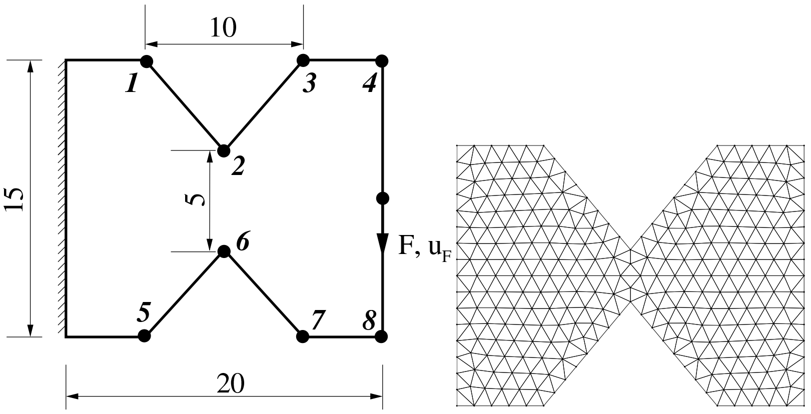
To complete the sensitivity analysis, we still need to evaluate $\frac{d\boldsymbol{\mathcal{X}}}{d\boldsymbol{x}}$ present in Eq. (11). To compute consistent sensitivity information the current number of nodes nn and element nodal connectivity T as defined in (5) needs to remain unchanged and any change in nodal positions needs to be smooth and continuous [2,12]. In particular, when semi-analytical sensitivities are used to compute $\frac{d\boldsymbol{\mathcal{X}}}{d\boldsymbol{x}}$ using conventional finite differences.

However, computing the sensitivity analytically yields consistent sensitivity information as no perturbation is required that may change the computational mesh Λ. Similarly, the benefit of the complex-step method is that the computational mesh Λ remains unchanged on the real axis as only a pure imaginary step is required to compute the sensitivity information.

Again, we highlight that accurate *associated* sensitivity information is computable everywhere although the functions is not everywhere differentiable. This can be achieved by analytical sensitivities, consistent semi-analytical sensitivities, automatic differentiation or the complex step method.

4. NUMERICAL EXPERIMENTS

Consider the linear elastic bow-tie structure depicted in 3(a), with eight control variables $x_i = 1, 2, \ldots, 8$, that define the geometry as indicated. A force F is applied as shown and u_F indicates the displacement in the vertical direction at the point where the force is applied.



(a)

(b)

Figure 3. (a) Bow-tie structure used to compare the various sensitivity strategies and (b) corresponding discretized geometry.

We compute the sensitivity of u_F w.r.t. $x_i = 1, 2, \ldots, 8$ using a remeshing strategy. We compute both *consistent* sensitivities i.e. no changes in mesh topology is allowed as well as *inconsistent* sensitivities i.e. the mesh topology is allowed to change during the sensitivity computation.

The absolute error for both the *consistent* and *inconsistent* sensitivity computations for the forward difference, backward difference, central difference and complex-step method are depicted in Figures 4 and 5. The *inconsistent* sensitivities for points 1-4 are depicted in Figures 4 (a), (c), (e) and (g) respectively, while the absolute sensitivity error for points 5-8 are depicted in Figures 5 (a), (c), (e) and (g). In turn, the absolute error for the *consistent* sensitivities are depicted in Figures 4 (b), (d), (f) and (h) and in Figures 5 (b), (d), (f) and (h) for respectively points 1-4 and 5-8 respectively.

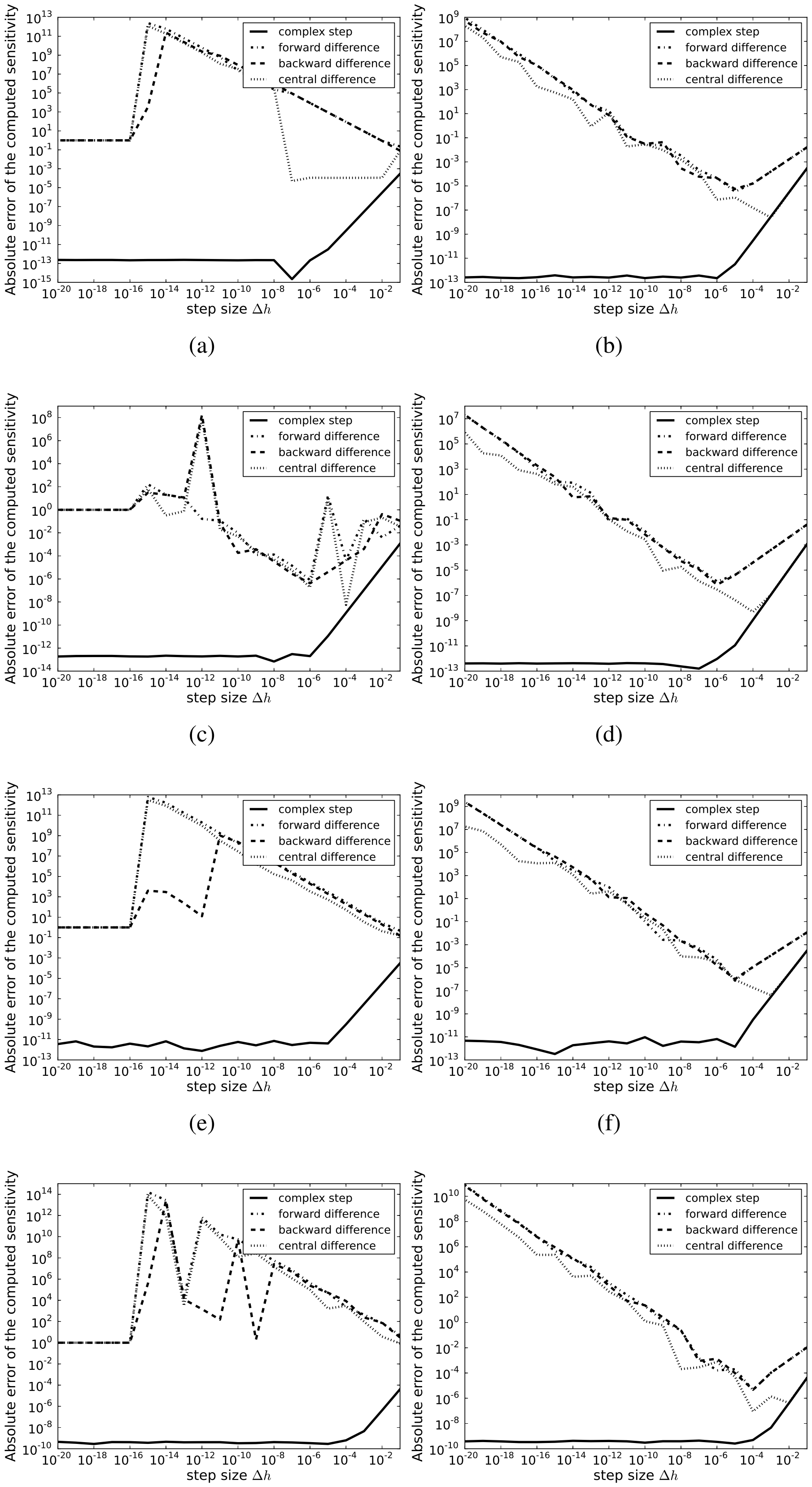
Note that the absolute error of the complex-step sensitivities are unaffected whether the non-constant discretization strategy preserves the mesh topology or not. This is in stark contrast to the finite difference strategies who's absolute error severely degrades for *inconsistent* sensitivities as opposed to *consistent* sensitivities. Consider the inconsistent sensitivity computation of point 3, as depicted in Figure 4(e). It is clear that neither the forward, backward, or central difference schemes compute appropriate sensitivities and hence even a modified difference strategy as discussed in Section 2 would fail to give accurate sensitivities, independent of the step size.

Consider the absolute error of the backward difference computed sensitivity error depicted in Figure 5(e). As indicated, a jump in the absolute error of the sensitivity occurs for a step size change from 10^{-8} to 10^{-9}, with the latter significantly more accurate. The jump in error is a result of a change in the mesh topology from being inconsistent for the 10^{-8} step size to being consistent for the 10^{-9} step size. This is clearly depicted in Figure 6, where the initial mesh (in dashed lines) is superimposed onto the finite difference perturbed mesh for step sizes 10^{-8} and 10^{-9} respectively in Figures 6(a) and (b). Changes in the perturbed mesh is evident in the lower left part of the bow-tie structure. Although these changes in mesh topology are removed from point 7 where the sensitivity is computed as well as the point where the load is applied, the discretization error in the stiffness of the structure is significant enough to adversely affect sensitivity computations.

5. CONCLUSION

We demonstrated the benefits of the complex-step method to compute accurate sensitivity information for discontinuous functions. The complex-step method allows for the computation of sensitivity information at a discontinuity where the derivative is not defined. We showed that the complex-step method is a viable numerical strategy to compute *associated* derivatives or *associated* gradients, as required by gradient-only optimization.

We highlighted the well known need for consistent sensitivity computations when using non-constant discretization strategies during objective or constraint computations. The complex-step method allows for the computation of consistent sensitivity information with similar ease as for conventional finite differences, without having to modify non-constant discretization strategies to preserve discretization topology during sensitivity calculations. Only minor modifications are required, which include the handling of complex variables, the overloading of some functions and reviewing some branching statements.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 4. Absolute error of the finite difference sensitivities for points 1-4 on the bow-tie structure computed inconsistently in respectively (a), (c), (e) and (g) and computed consistently in respectively (b), (d), (f) and (h).

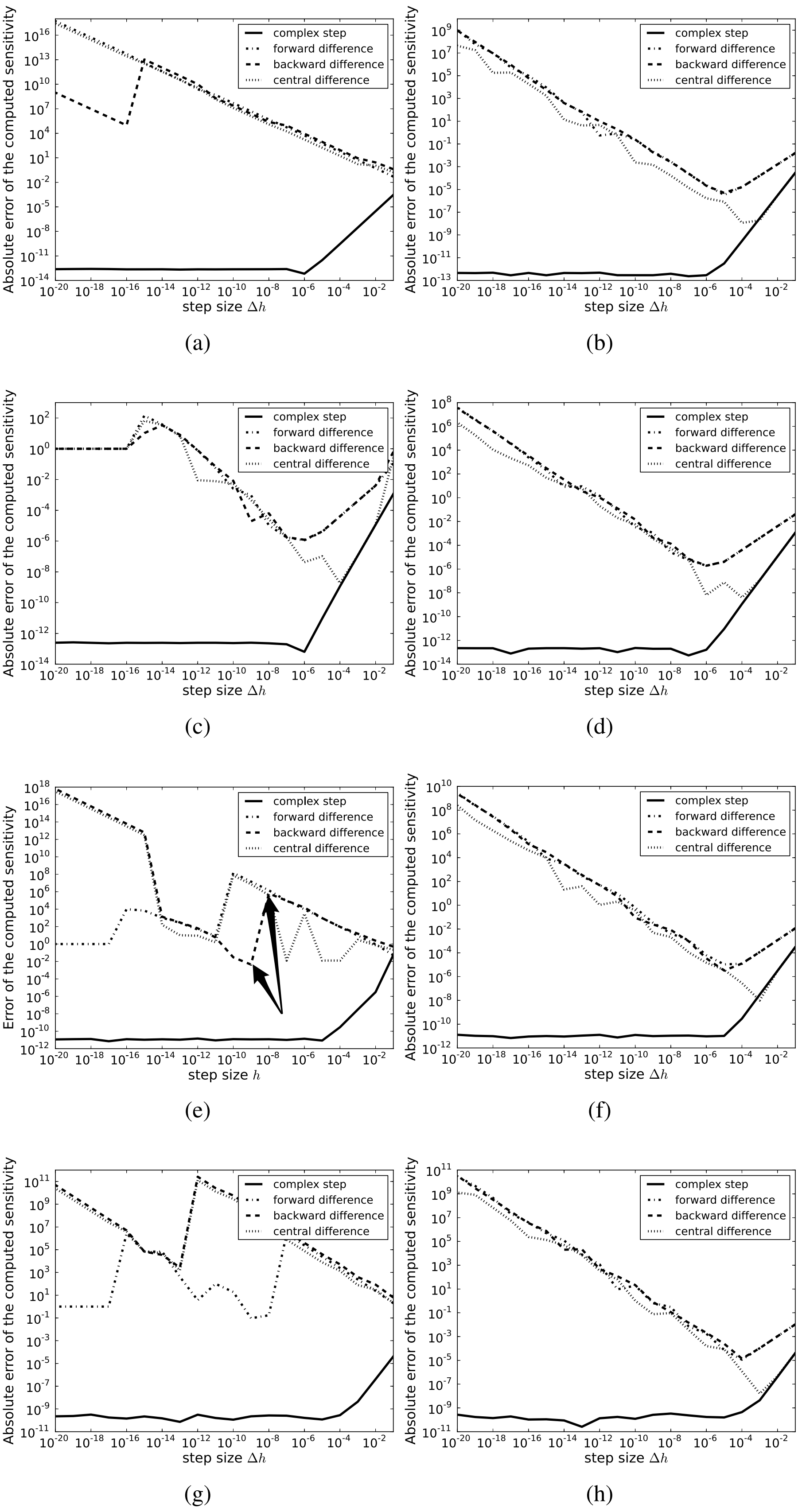
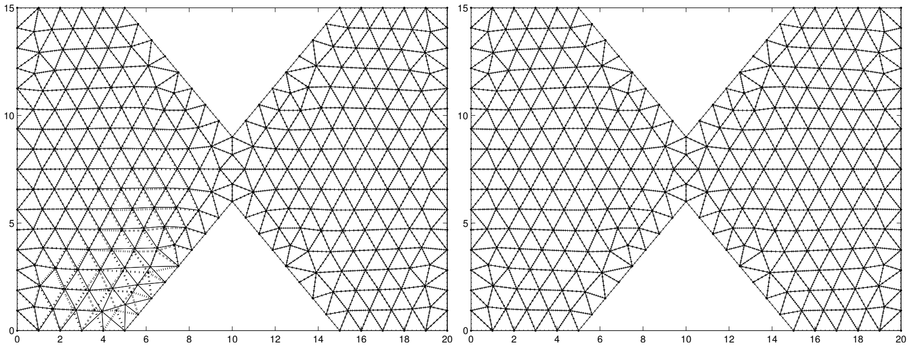


Figure 5. Absolute error of the finite difference sensitivities for points 5-8 on the bow-tie structure computed inconsistently in respectively (a), (c), (e) and (g) and computed consistently in respectively (b), (d), (f) and (h).



(a) (b)

Figure 6. Consider the inconsistent backward difference sensitivity computation of point 7 with absolute error depicted in Figure 5(e). Depicted are the initial mesh (dashed line) superimposed onto the backward difference mesh (solid line) with step sizes of (a) 10^{-8} and (b) 10^{-9}, respectively. The change is mesh topology is evident in the lower left of (a) as opposed to no change in mesh topology in (b).

6. REFERENCES

[1] Cerviño L., Bewley T., “On the extension of the complex-step derivative technique to pseudospectral algorithms”. *J. Comput. Phys.* 187, 544-549, 2003.

[2] van Keulen F., Haftka R., Kim N., “Review of options for structural design sensitivity analysis Part 1 : Linear systems”. *Comput. Method. Appl. M.* 194, 3213-3243, 2005.

[3] Kim J., Bates D., Postlethwaite I., “Nonlinear robust performance analysis using complex-step gradient approximation”. *Automatica* 42, 177-182, 2006.

[4] Kim S., Ryu J., Cho M., “Numerically generated tangent stiffness matrices using the complex variable derivative method for nonlinear structural analysiss”. *Comput. Method. Appl. M.* 200, 403-413, 2011.

[5] Lai K.L., Crassidis J., ‘Generalizations of the complex-step derivative approximation”. *Coll. Tech. Papers: AIAA Guid. Nav. Contrl. Conf.* 4, 2540-2564, 2006.

[6] Laporte E., Tallec P.L., Numerical Methods in Sensitivity Analysis and Shape Optimization. Modeling and Simulation in Science, Birkhäuser, 2002.

[7] Lyness J., “Differentiation formulas for analytic functions”. *Math. Comput.* 22, 352-362, 1968.

[8] Lyness J., Moler C., “Numerical Differentiation of Analytic Functions”. *SIAM J. Numer. Anal.* 4, 202-210,1967.

[9] Martins J., Sturdza P., Alonso J.J., “The Complex-Step Derivative Approximation”. *ACM T. Math. Software* 29, 245-262, 2003.

[10] Martins J., Alonso J., Reuther J., “A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design”. *Optim. Eng.* 6, 33-62, 2005.

[11] Martins J.R.R.A., Sturdza P., Alonso J.J., “The connection between the complex-step derivative approximation and algorithmic differentiation”. *AIAA Paper* 921, 1-11, 2001.

[12] Schleupen A., Maute K., Ramm E., “Adaptive FE-procedures in shape optimization”. *Struct. Multidiscip. O.* 19, 282-302, 2000.

[13] Shor N., “New development trends in nondifferentiable optimization”. *Cybern. Syst. Anal+* 13, 881-886, 1977.

[14] Squire M., Trapp G., “Using Complex Variables to estimate derivatives of real functions”. *SIAM Rev.* 40, 110-112, 1998.

[15] Vatsa V., “Computation of sensitivity derivatives of Navier-Stokes equations using complex variables”. *Adv. Eng. Softw.* 31, 655-659, 2000.

[16] Voorhees A., Millwater H., Bagley R., “Complex variable methods for shape sensitivity of finite element methods”. *Finite Elem. Anal. Des*. 47, 1146-1156, 2011.

[17] Voorhees A., Millwater H., Bagley R., Golden P., “Fatigue sensitivity analysis using complex variable methods”. *Int. J. Fatigue* 40, 61-73, 2012.

[18] Wilke D.N., Kok S., Snyman J.A., Groenwold A.A., “Gradient-only approaches to avoid spurious local minima in unconstrained optimization”. *Opt. Eng.* available online, 2012.