COMPUTATIONAL ISSUES IN SENSITIVITY ANALYSIS FOR 1-D INTERFACE PROBLEMS*

Lisa G. Davis

Montana State University, Department of Mathematical Sciences, PO Box 172400,
Bozeman, MT 59717–2400, USA
Email: davis@math.montana.edu

John R. Singler

Missouri University of Science and Technology, Department of Mathematics and Statistics, 400 W.
12th St., Rolla, MO 65409–0020, USA
Email: singlerj@mst.edu

Abstract

This paper is concerned with the construction of accurate and efficient computational algorithms for the numerical approximation of sensitivities with respect to a parameter dependent interface location. Motivated by sensitivity analysis with respect to piezoelectric actuator placement on an Euler-Bernoulli beam, this work illustrates the key concepts related to sensitivity equation formulation for interface problems where the parameter of interest determines the location of the interface. A fourth order model problem is considered, and a homogenization procedure for sensitivity computation is constructed using standard finite element methods. Numerical results show that proper formulation and approximation of the sensitivity interface conditions is critical to obtaining convergent numerical sensitivity approximations. A second order elliptic interface model problem is also mentioned, and the homogenization procedure is outlined briefly for this model.

Mathematics subject classification: 65N06, 65B99.
Key words: Finite element method, Interface Problems Sensitivity Equation.

1. Introduction

Scientists often want to measure how well a mathematical model represents the fundamental behaviors of a physical system, and they are often charged with the task of quantifying some measure of how the uncertainty in the model parameters proliferates into uncertainty in the results of the model simulations. As pointed out in [1], sensitivity analysis and uncertainty analysis combine to produce a systematic approach to developing a comprehensive understanding of a mathematical model, the data it produces, and the way that the data is used to influence the design of many engineering systems. Accurate sensitivity calculations play an important role in this process. The term *Sensitivity Equation Methods* (SEMs) refers to a large class of techniques that attempt to derive, analyze, and solve equations whose solutions are functions referred to as *sensitivities*. Sensitivities are derivatives which describe how small changes in design parameters affect the state variables of a mathematical model. Continuous Sensitivity Equation Methods (CSEMs) are one such technique in this class of methods. CSEMs have been used to compute gradients and greatly improve design cycle times in optimization-based design, see [2–4]. In

addition, they can be used to construct fast solvers for computational fluid dynamics [5] and are essential to quantifying uncertainties in parameter dependent systems [1, 6].

The CSEM approach requires one to first derive the appropriate sensitivity equation, then to show the resulting equation is well posed in an appropriate function space, and finally to develop good numerical schemes for approximating the sensitivities. In certain situations, such as when geometry or shape parameters are considered, the sensitivity equations may have very weak solutions (e.g., only L^2 in space) and require that one develop numerical algorithms that capture these weak solutions, see [7] for an example of this process.

A valid question to ask is whether the development of special numerical methods for approximating the sensitivities is necessary; can simple, "natural," computational methods be used to obtain reasonable sensitivity approximations? Furthermore, is it also essential to analyze the continuous sensitivity equation to show that it is a properly posed mathematical problem? In this work, we show that, in general, the answer to both of these questions is yes. Specifically, we consider a fourth order elliptic problem where the parameter of interest governs the location of a coefficient discontinuity. We summarize our results as follows:

- We give a proper formulation of the continuous sensitivity equation and use this formulation to construct a convergent numerical scheme for approximating the sensitivity.

- We consider a simple, "natural," computational method for approximating the sensitivity and show that it completely fails to yield convergent sensitivity approximations.

- We show that the reason for the failure of this methods is that it fails to recognize a certain property of the sensitivity; furthermore, this property can only be found through a preliminary analysis of the problem.

The main goal of this paper is to illustrate that when applying sensitivity analysis techniques to partial differential equations with discontinuous coefficients, or *interface problems* as they are sometimes called, it is necessary to both analyze the continuous sensitivity equation and to develop a special numerical method to accurately approximate the sensitivity.

The fourth order model problem we consider in this work shares many similarities with the Euler-Bernoulli beam model considered in [8]. Specifically, the model problem contains discontinuous coefficients where the location of the discontinuity serves as the parameter of interest for the sensitivity analysis. Furthermore, the derivation of the sensitivity equation for this simple model exhibits similar issues to that of the original Euler-Bernoulli beam model. However, one can explicitly write down the solution to the model equation and derive an explicit form of the sensitivity variable. This allows us to identify some of the key ideas that are relevant when applying sensitivity analysis to interface problems. We note that our results outlined above for the model problem most certainly apply to the corresponding sensitivity approximations for the more complicated Euler-Bernoulli beam model with patch actuators.

We begin with notation and an outline of the motivating beam model in Section 2. The simplified model problem is presented in Section 3 along with the exact solution of the problem and the corresponding sensitivity. Section 3.1 gives a brief description of the standard finite element formulation for constructing state variable approximations. The continuous sensitivity equation (CSE) is studied in Section 4. A homogenization procedure is used to prove that the CSE is well posed, and a corresponding numerical technique is used to obtain convergent numerical sensitivity approximations. Section 5 uses one type of *Discretize-then-Differentiate* (DD) methodology for deriving a sensitivity equation. Numerical experiments are shown to yield

sensitivity approximations which fail to converge to the true sensitivity for the model problem. In Section 6, we use a formal CSE to explain the failure of the DD sensitivity approximations. Section 7 briefly describes the application of the same type of homogenization procedure to a second order elliptic interface model and its corresponding sensitivity equation.

1.1. Notation

We begin by briefly defining the function spaces and mathematical notation related to the course of this exposition. Let $H^m(\Omega)$ denote the usual Sobolev space of "functions" whose partial derivatives, up to order m, are square integrable. Let $L^2 = L^2(\Omega)$ with inner product defined by

$$(u, v) = \int_\Omega u(x)v(x)dx$$

for all $u(\cdot), v(\cdot) \in L^2$. For the content of this work, $\Omega = (0, \ell)$ where $\ell \in \mathbb{R}$ and $\ell > 0$. This paper makes use of the typical Sobolev spaces for fourth-order equations with the exception of some particular boundary conditions. We use the function space denoted by $V = H_L^2(0, \ell) = \{ v \in H^2(0, \ell) : v(0) = 0, v_x(0) = 0 \}$.

If a function $\phi : \mathbb{R} \times A \to \mathbb{R}$ depends on a spatial variable x and a parameter α, then differentiation with respect to the spatial variable (in the Fréchet sense) is denoted by $\phi_x(x; \alpha)$, and differentiation (also in the Fréchet sense) with respect to the parameter α will be denoted $\phi_\alpha(x; \alpha)$. In the situation where a function has an associated index, the notation $\phi_{i,xx}(x)$ is used to denote $\partial^2/\partial x^2 \phi_i(x)$. As in the previous sentence, the explicit dependence of ϕ on the parameter α may be suppressed at times in order to simplify the notation.

We also remind the reader of Leibniz' rule for differentiating integral terms.

$$\frac{d}{d\alpha} \int_{f(\alpha)}^{g(\alpha)} h(x; \alpha)\, dx = \int_{f(\alpha)}^{g(\alpha)} h_\alpha(x; \alpha)\, dx + h(g(\alpha); \alpha)g_\alpha(\alpha) - h(f(\alpha); \alpha)f_\alpha(\alpha), \qquad (1.1)$$

where it is assumed that each of the functions f, g, and h are differentiable with respect to α.

2. Motivating Problem

The motivation for the model problem considered in this paper is rooted in an investigation of sensitivity approximations for an Euler-Bernoulli beam model with a pair of piezoelectric patch actuators placed on the beam according to a location parameter, see [8]. The voltage input is adjusted to only consider pure bending motion. The partial differential equation contains discontinuous (spatially) material coefficients where the discontinuities correspond to the spatial location of the patch actuators. It is important to note that the discontinuous coefficients in this model are explicitly dependent on the location parameter for the patch actuators, and the mathematical formulation of the partial differential equation can be thought of as an interface problem where the location of the interface is parameter dependent. Moreover, the location parameter is the parameter of interest for the sensitivity analysis.

We begin by presenting the Euler-Bernoulli beam model for the situation when a pair of piezoceramic patches are placed on either side of the beam at the same spatial location, see Fig. 2.1.

The beam has length ℓ, height b, and thickness h. The length of the patch is given by L, and the patch location is given by the interval $[\alpha, \alpha + L]$. We consider the case where the patches
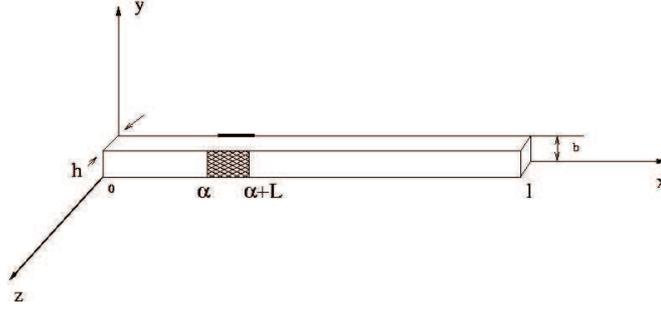
Fig. 2.1. Diagram of beam with a pair of piezoelectric patches of length L located on either side of the beam on the interval $x \in [\alpha, \alpha + L]$.

are excited out-of-phase which results in pure bending of the beam. We account for damping in the beam using both Kelvin-Voigt (material) and linear viscous (air) damping. The model presented below is derived and studied extensively in [9–11].

   Let $w(t, x)$ denote the deflection of the beam at time t and position x. For small deflections, the motion of the cantilevered beam is governed by the partial differential equation

$$\rho A w_{tt} + d_{lv} w_t + (d_{kv} I w_{txx})_{xx} + (EI w_{xx})_{xx} = g_{xx}(t, x), \tag{2.1}$$

$$w(t, 0) = w_x(t, 0) = 0, \tag{2.2}$$

$$EI w_{xx}(t, \ell) + d_{kv} I w_{txx}(t, \ell) = 0, \tag{2.3}$$

$$(EI w_{xx})_x(t, \ell) + (d_{kv} I w_{txx})_x(t, \ell) = 0, \tag{2.4}$$

with the initial deflection and initial velocity denoted by

$$w(0, x) = w_0(x), \quad w_t(0, x) = w_1(x).$$

The following coefficients represent material properties of the beam at a certain spatial location x: ρ is the mass density, A is the cross-sectional area, I is the moment of inertia, E is Young's modulus, and d_{lv} and d_{kv} are the coefficients of air and Kelvin-Voigt damping, respectively.

   The presence of the patches results in discontinuities in most of these coefficients, and this can be expressed by

$$\rho A(x) = \rho A_1 + \rho A_2 [H_1(x) - H_2(x)],$$

$$d_{kv} I(x) = d_{kv} I_1 + d_{kv} I_2 [H_1(x) - H_2(x)],$$

$$EI(x) = EI_1 + EI_2 [H_1(x) - H_2(x)],$$

where H_1 and H_2 denote the Heaviside functions with jump discontinuities at the left and right ends of the patch, respectively; i.e.,

$$H_1(x) = \begin{cases} 0, & 0 < x < \alpha, \\ 1, & \alpha < x < \ell, \end{cases} \qquad H_2(x) = \begin{cases} 0, & 0 < x < \alpha + L, \\ 1, & \alpha + L < x < \ell. \end{cases}$$

The constants ρA_1, EI_1, and $d_{kv} I_1$ correspond to the density, flexural rigidity, and Kelvin-Voigt damping properties of the beam, while the constants ρA_2, EI_2, and $d_{kv} I_2$ correspond to those of the patch, respectively. It is natural to assume that the damping due to air is not affected by the patches; therefore, the air damping coefficient, d_{lv}, is assumed to be constant over the length of the beam.

The patches influence the system by exerting a moment force on the section of the beam where they are located. The term $g_{xx}(t,x)$ accounts for this moment, and the spatial influence is described by a difference of Heaviside functions of the form

$$g_{xx}(t,x) = \kappa[H_1(x) - H_2(x)]_{xx}u(t).$$

The constant κ is a parameter describing the patch properties, and $u(t)$ is the voltage applied to the patch at time t. For a more thorough treatment of the model development and the specific forms of the beam and patch parameters, the reader is referred to [11].

One should observe that in (2.1), the damping $(d_{kv}Iw_{txx})_{xx}$, stiffness $(EIw_{xx})_{xx}$, and control term, $g_{xx}(t,x)$, all contain spatial derivatives of the Heaviside functions; consequently, the PDE is naturally interpreted using the variational (or weak) formulation. The weak form is also convenient for the numerical simulations which use the finite element method for approximation in space. After multiplying (2.1) by a function ϕ and integrating twice by parts, the variational formulation of the beam model (2.1)-(2.4) is given by

$$\int_0^\ell \rho A(x;\alpha)w_{tt}(t,x)\phi(x)\,dx + \int_0^\ell d_{lv}w_t(t,x)\phi(x)\,dx + \int_0^\ell d_{kv}I(x;\alpha)w_{txx}(t,x)\phi_{xx}(x)\,dx$$

$$+ \int_0^\ell EI(x;\alpha)w_{xx}(t,x)\phi_{xx}(x)\,dx = \int_0^\ell g(t,x;\alpha)\phi_{xx}(x)\,dx, \qquad \forall \phi \in H_L^2(0,\ell), \ \ (2.5)$$

where

$$g(t,x;\alpha) = \kappa[H_1(x) - H_2(x)]u(t).$$

In [8], sensitivity approximations are numerically computed by deriving a variational sensitivity equation. This equation is obtained by implicitly differentiating the variational form of the state equation with respect to the interface parameter. Numerical calculations for the sensitivity approximations are shown but not verified through any other type of comparison. Questions concerning the validity of those sensitivity approximations serve to motivate the discussion in this paper.

In this work, we consider a related problem that clearly demonstrates the issues in computing sensitivities with respect to interface locations. To simplify the problem, we drop the time dependence and only consider one interface. Specifically, the variational form of the model takes the form of the last two terms in Eq. (2.5), and the piecewise constant coefficient functions EI and g each have one discontinuity.

3. Model Problem: A Fourth-Order Interface Equation

We begin with the strong form of the model problem. Let $\alpha \in (0,\ell)$ be a real-valued parameter. Consider the following interface problem: find $w(x)$ satisfying

$$\big(EI(x;\alpha)w_{xx}(x)\big)_{xx} = 0, \tag{3.1a}$$

$$w(0) = 0, \qquad w_x(0) = 0, \tag{3.1b}$$

$$EI_2 w_{xx}(\ell) = g_2, \qquad EI_2 w_{xxx}(\ell) = 0, \tag{3.1c}$$

$$w(\alpha^-) = w(\alpha^+), \tag{3.1d}$$

$$w_x(\alpha^-) = w_x(\alpha^+), \tag{3.1e}$$

$$EI_1 w_{xx}(\alpha^-) - EI_2 w_{xx}(\alpha^+) = g_1 - g_2, \tag{3.1f}$$

$$EI_1 w_{xxx}(\alpha^-) - EI_2 w_{xxx}(\alpha^+) = 0. \tag{3.1g}$$

This problem contains a differential equation (3.1a), boundary conditions (3.1b)-(3.1c), and conditions at the interface $x = \alpha$ (3.1d)-(3.1g). The coefficient function $EI(x; \alpha)$ is piecewise constant and is given by

$$EI(x; \alpha) = \begin{cases} EI_1, & 0 < x < \alpha, \\ EI_2, & \alpha < x < \ell, \end{cases} \tag{3.2}$$

where EI_1 and EI_2 are positive real constants. The constants g_1 and g_2 appearing in the boundary conditions and interface conditions at $x = \alpha$ are real-valued.

Remark 3.1. This strong form contains interface conditions in (3.1d)-(3.1g) that characterize the smoothness of the state variable at $x = \alpha$. These conditions are often not included in the statement of a PDE of this type (note that we did not include them in the beam model above). However, they are stated explicitly here because we show below that they are useful for determining the regularity of the sensitivity PDE.

Multiplying the differential equation by a test function $\phi \in V = H_L^2(0, \ell)$, integrating over the intervals $(0, \alpha)$ and (α, ℓ), and integrating by parts twice shows that the solution w must satisfy the following variational problem: find $w \in V$ such that

$$\int_0^\ell EI(x; \alpha) w_{xx}(x) \phi_{xx}(x) \, dx = \int_0^\ell g(x; \alpha) \phi_{xx}(x) \, dx \tag{3.3}$$

for all $\phi \in V$, where $g(x; \alpha)$ is the piecewise constant function

$$g(x; \alpha) = \begin{cases} g_1, & 0 < x < \alpha, \\ g_2, & \alpha < x < \ell. \end{cases} \tag{3.4}$$

Since $EI_1, EI_2 > 0$ the variational problem in (3.3) has a unique solution by the Lax-Milgram Theorem (see, e.g., [12]). Moreover, it can be checked that the solution to (3.3) is given by

$$w(x; \alpha) = \begin{cases} c_1 x^2/2, & 0 < x < \alpha, \\ c_2 x^2/2 + \alpha(c_1 - c_2)x - \alpha^2(c_1 - c_2)/2, & \alpha < x < \ell, \end{cases} \tag{3.5}$$

where $c_i = g_i/EI_i$ for $i = 1, 2$.

The parameter of interest for this study is α, the parameter determining the interface location in the coefficients of the state equation, and we denote the dependence of the state variable on α by $w(x) = w(x; \alpha)$. The sensitivity of the state with respect to the interface location is defined by

$$s(x; \alpha) = \frac{\partial w}{\partial \alpha}(x; \alpha).$$

Differentiating the exact solution, $w(x; \alpha)$, given in (3.5), with respect to α shows that the sensitivity is given by

$$s(x; \alpha) = \begin{cases} 0, & 0 < x < \alpha, \\ (c_1 - c_2)x - \alpha(c_1 - c_2), & \alpha < x < \ell. \end{cases} \tag{3.6}$$

It is important to note that the solution of the model problem w is in $V = H_L^2(0, \ell)$, while the sensitivity s is not in V. Specifically, the sensitivity $s(x; \alpha)$ is continuous and differentiable in x with derivative

$$s_x(x; \alpha) = \begin{cases} 0, & 0 < x < \alpha, \\ c_1 - c_2, & \alpha < x < \ell. \end{cases}$$

If $c_1 \neq c_2$, then s_x is not differentiable with respect to x and therefore s is not in V. Therefore, the sensitivity does not have the same spatial smoothness as the state variable. In more complicated problems, the exact state and sensitivity variables will not be known and it might not be clear whether the sensitivity shares the same regularity as the state. In Section 4 below, we demonstrate that the proper formulation of the continuous sensitivity equation will indicate the smoothness of the sensitivity. This information can be used to guide the numerical approximation of the sensitivity.

Although the main goal of this work is to develop efficient and convergent computational schemes for sensitivity approximations, convergent state variable approximations are required for that process. The next section contains a brief outline of the discretization for the state equation.

3.1. Finite element methods for the state variable approximations

The finite element method is used for the discretization of the variational form of the state equation in (3.3). We consider finite element bases with first cubic B-spline basis functions (see [13,14]) and then Hermite cubic basis functions (see [12,15]). Both types of basis functions belong to $H^2(0,\ell)$; however, the second derivatives of the cubic B-splines are continuous, while the second derivatives of the Hermite cubics are only piecewise continuous with discontinuities at the finite element nodes. As shown in later sections, this has implications for the sensitivity approximations produced using each of the discretization schemes.

Let ϕ_j represent the jth finite element basis function and define $V^N = \text{span}\{\phi_j\}_{j=1}^N$ to be the corresponding finite dimensional subset of V. The finite element method constructs an approximate solution $w^N \in V^N$ of the model problem (3.3). Since $w^N \in V^N$, it can be represented as the linear combination

$$w^N(x) = \sum_{j=1}^N a_j \phi_j(x). \tag{3.7}$$

Substituting this expression into the variational equation (3.3) and taking $\phi = \phi_i$ for $i = 1, \cdots, N$ yields the approximating linear system

$$K^N \mathbf{a}^N = \mathbf{f}^N, \tag{3.8}$$

where $\mathbf{a}^N = [a_1, \cdots, a_N]^T$,

$$K_{ij}^N(\alpha) = \int_0^\ell EI(x;\alpha)\phi_{j,xx}(x)\phi_{i,xx}(x)\,dx, \quad \text{and} \quad f_i^N(\alpha) = \int_0^\ell g(x;\alpha)\phi_{i,xx}(x)\,dx. \tag{3.9}$$

We briefly give a sample of the numerical results for the finite element approximations defined in (3.7)-(3.9). In order to maintain continuity in the exposition, we have chosen a particular set of parameter values that are used for the state variable as well as the sensitivity variable computations shown in this paper. Those parameter values are given in Table 3.1. Recall that the true expression for the state variable given in (3.5) depends on $c_i = g_i/EI_i$ for $i = 1, 2$.

Fig. 3.1 shows the graph of the true state variable (3.5) and its finite element approximations using 33 equally spaced nodes and the parameter value $\alpha = 0.5$. Finite element approximations using cubic B-spline (cbs) basis functions as well as Hermite cubic (hc) basis functions are shown.

Table 3.1: Parameter values.

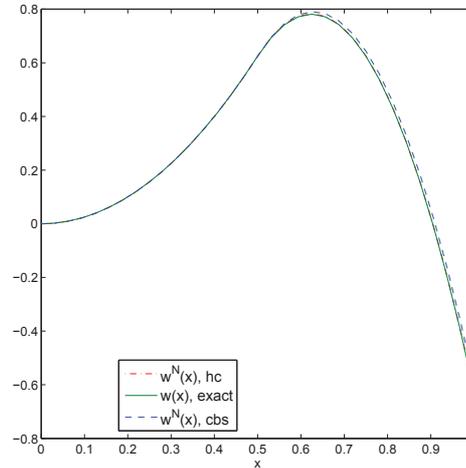| EI_1	EI_2	g_1	g_2	c_1	c_2
0.2	0.1	1	-2	5	-20

Fig. 3.1. The exact solution (3.5) of the model equation (3.3) for $\alpha = 0.5$ compared with cubic B-spline (cbs) and Hermite cubic (hc) finite element approximations using 33 equally spaced nodes.

Note that the choice of 33 nodes in the finite element mesh is used to place a node at the interface location, $x = \alpha = 0.5$, which is the point of discontinuity for the coefficient functions EI and g in the model equation (3.3). When numerical simulations are conducted using the algorithm outlined above, each of these finite element approximations converge to the true solution as $N \to \infty$ clearly, and we note that a more coarse mesh also exhibited very good agreement between the computations and the true state variable.

Remark 3.2. We have taken a simple approach to approximating the solution of the model problem. For more complicated problems, special interface methods may be necessary to obtain accurate state approximations. We comment further on numerical methods for interface problems and sensitivity equations for higher dimensional problems in the conclusion.

Now we move to a discussion of the sensitivity computations.

4. The Continuous Sensitivity Equation

In this section, we show that one can accurately approximate the sensitivity by performing an initial analysis of the smoothness of the parameter dependence in the problem. For interface problems of the type considered in this paper, the state equation does not vary smoothly with respect to the parameter α because it is the parameter that governs the location of a discontinuity in the coefficients of the equation. In order to accurately approximate the sensitivity, one must account for this lack of smoothness. In particular, the interface conditions satisfied by the state variable can be used to derive the appropriate sensitivity interface conditions; these in turn allow us to properly pose the continuous sensitivity equation.

We formally differentiate (with respect to α) the state equation, the boundary conditions, and the interface conditions in Eq. (3.1) to arrive at a strong interface problem for the sensitivity $s(x; \alpha) = (d/d\alpha)w(x; \alpha)$:

$$(EI(x; \alpha)s_{xx}(x; \alpha))_{xx} = 0, \tag{4.1a}$$

$$s(0) = 0, \quad EI_2 s_{xx}(\ell) = 0, \tag{4.1b}$$

$$s_x(0) = 0, \quad EI_2 s_{xxx}(\ell) = 0, \tag{4.1c}$$

$$s(\alpha^-) - s(\alpha^+) = 0, \tag{4.1d}$$

$$s_x(\alpha^-) - s_x(\alpha^+) = d, \tag{4.1e}$$

$$EI_1 s_{xx}(\alpha^-) - EI_2 s_{xx}(\alpha^+) = 0, \tag{4.1f}$$

$$EI_1 s_{xxx}(\alpha^-) - EI_2 s_{xxx}(\alpha^+) = 0. \tag{4.1g}$$

Eq. (4.1d) is a homogeneous interface condition for the sensitivity, and it is obtained by differentiating equation (3.1d) with respect to α, applying the total derivative to the interface condition, and using the continuity of the interface condition for w_x given in (3.1e) to simplify the expression. Eq. (4.1e) is a nonhomogeneous interface condition

$$s_x(\alpha^-) - s_x(\alpha^+) = d,$$

where

$$d = w_{xx}(\alpha^+) - w_{xx}(\alpha^-)$$

is the jump in the second derivative of the state. This is obtained by differentiating equation (3.1e) with respect to α, and again applying the total derivative when differentiating this interface condition.

Recall from Section 3 that the exact state and sensitivity do not possess the same degree of smoothness; the state w is in $V = H_L^2(0, \ell)$, while the sensitivity is not in V. This latter property can be seen directly from the above sensitivity equation. To verify this, note that the nonhomogeneous interface condition

$$s_x(\alpha^-) - s_x(\alpha^+) = d = c_2 - c_1 = g_2/EI_2 - g_1/EI_1 \tag{4.2}$$

guarantees that s has a jump in its first derivative as long as $c_1 \neq c_2$ which implies that $s \notin V$.

For this problem, we achieve two goals. The first is to derive a weak formulation of a sensitivity equation that is well-posed in the original function space V (so that we can make use of the same finite element basis used to approximate the state variable), and the second is to show that the sensitivity variable of interest is indeed the unique solution of that sensitivity equation. The details are given in the following section.

4.1. Homogenization procedure

We use the following homogenization technique in order to develop a continuous sensitivity equation which is well-posed; a change of variables is required, and the resulting sensitivity equation can be shown to have a unique solution with the standard Lax-Milgram theorem. Once that variational equation is solved, one can recover the original sensitivity variable directly.

We establish the well posedness of the interface sensitivity equation by changing variables to homogenize the first derivative interface condition (4.2). This technique is used in [16]

and [17, Section 8.6] to numerically approximate solutions of elliptic interface problems with nonhomogeneous jump conditions. Let h be a function that satisfies the interface conditions

$$h(\alpha^-) - h(\alpha^+) = 0, \quad h_x(\alpha^-) - h_x(\alpha^+) = d,$$

and define the function p by $s(x) = p(x) + h(x)$. Examining the interface conditions in the sensitivity equation (4.1d)-(4.1e), one can check that $p(x)$ satisfies the homogeneous interface conditions

$$p(\alpha^-) - p(\alpha^+) = 0, \quad p_x(\alpha^-) - p_x(\alpha^+) = 0.$$

Once the function h is chosen and this change of variables is defined, one can derive a variational equation with *homogeneous* interface conditions whose unique solution is p. This homogenization procedure is made precise in the following theorem.

Theorem 4.1. *Let d be any real number. There exists a unique solution $s \in L^2(0,\ell)$ of the continuous sensitivity equation (4.1) in the following sense. Choose a function h defined on the interval $(0,\ell)$ satisfying the following:*

1. *h is H^2 away from the interface, i.e., $h(\cdot)|_{(0,\alpha)} \in H^2(0,\alpha)$ and $h(\cdot)|_{(\alpha,\ell)} \in H^2(\alpha,\ell)$;*
2. *h satisfies the essential boundary conditions $h(0) = 0$, $h_x(0) = 0$;*
3. *h meets the interface conditions*

$$h(\alpha^-) - h(\alpha^+) = 0, \quad h_x(\alpha^-) - h_x(\alpha^+) = d. \tag{4.3}$$

Then $p = s - h \in V = H_L^2(0,\ell)$ is uniquely determined as the solution of the following variational equation: find $p \in V$ such that

$$\int_0^\ell EI p_{xx} \phi_{xx}\, dx = - \int_0^\alpha EI_1 h_{xx}\phi_{xx}\, dx - \int_\alpha^\ell EI_2 h_{xx}\phi_{xx}\, dx, \tag{4.4}$$

for all $\phi \in V$. The solution $s(x;\alpha)$ is independent of the function h; the choice of h simply governs the resulting function $p \in H_L^2(0,\ell)$.

Proof. Let h be a function as described in the theorem. Multiplying the strong interface problem (4.1a) by a test function $\phi \in V$, integrating over the intervals $(0,\alpha)$ and (α,ℓ), substituting $s = p + h$, and integrating by parts gives the variational problem (4.4) for p. The Lax-Milgram Theorem shows that this equation has a unique solution p in V.

It remains to prove that using two different functions h results in the same sensitivity. Let h_1 and h_2 be two homogenizing functions as defined above and let $s_1 = p_1 + h_1$ and $s_2 = p_2 + h_2$, where each p_i is the unique solution of the variational equation (4.4) with $h = h_i$. Since each h_i satisfies the nonhomogeneous interface conditions (4.3), we have

$$h_1 - h_2 \in V = H_L^2(0,\ell).$$

Thus, subtracting the variational equations (4.4) for p_1 and p_2 shows

$$\int_0^\ell EI[p_1 + h_1 - p_2 - h_2]_{xx}\phi_{xx}\, dx = 0$$

for all $\phi \in V$. Since $h_1 - h_2 \in V$, $p_1 + h_1 - p_2 - h_2$ is also in V. The integral operator as a mapping from V to V' (the dual space of V) is invertible; therefore, we have

$$p_1 + h_1 - p_2 - h_2 = 0, \quad \text{or} \quad s_1 = s_2.$$

Thus, s is uniquely determined and the sensitivity equation (4.1) is well posed. $\qquad \square$

Remark 4.1. There are many choices for a "homogenizing" function h described in the theorem, and one such function is given by

$$h(x) = \begin{cases} d\,\alpha^{-1}x^2, & 0 < x < \alpha, \\ d\,x, & \alpha < x < \ell. \end{cases} \tag{4.5}$$

For this particular choice of h, it can be checked that the unique solution of the homogenized variational problem (4.4) is given by

$$p(x) = \begin{cases} -d\,\alpha^{-1}x^2, & 0 < x < \alpha, \\ -2\,d\,x + d\,\alpha, & \alpha < x < \ell. \end{cases}$$

Since $d = w_{xx}(\alpha^+) - w_{xx}(\alpha^-) = c_2 - c_1$, we recover the true sensitivity (3.6):

$$s(x) = p(x) + h(x) = \begin{cases} 0, & 0 < x < \alpha, \\ (c_1 - c_2)x - \alpha(c_1 - c_2), & \alpha < x < \ell. \end{cases}$$

The homogenization procedure given here is one approach to deriving a well-posed sensitivity equation. Alternatively, one may follow the techniques applied in [7] and use a very weak formulation of the continuous sensitivity equation (4.1) to directly show the well posedness of the problem. Regardless of the technique, it is important to recognize that the key piece of information required is the nature of the interface conditions that hold for the state variable. Once those are known, then the corresponding interface conditions satisfied by the sensitivity variable can be derived. It is the interface condition information which allows us to properly pose a variational equation from which the true sensitivity variable can be recovered. The homogenization procedure used here is also a convenient formulation for numerical computation of the sensitivity $s(x;\alpha)$, and an algorithm for the numerical calculation is discussed briefly in the following section.

4.2. Numerical results

The preceding analysis of the sensitivity equation can be used to guide the choice of a numerical algorithm to accurately approximate the sensitivity. Since the homogenization technique led to the well posedness of the sensitivity equation, we follow that procedure for our approximations.

Algorithm: Homogenization Procedure to Approximate the Interface Sensitivity

1. Obtain a Hermite cubic finite element approximation $w^N(x)$ of the state as described in Section 3.1.

2. Approximate the jump $d = w_{xx}(\alpha^+) - w_{xx}(\alpha^-)$ by $d^N = w_{xx}^N(\alpha^+) - w_{xx}^N(\alpha^-)$. That is, one uses the second derivative of the finite element approximation of w to approximate the second derivative information needed for the interface jump condition.

3. Replace d in (4.5) with the computed jump d^N to form an approximate homogenization function $h^N(x)$.

---

4. Obtain a finite element approximation $p^N(x)$ to the solution $p(x)$ of the "homogenized" variational problem (4.4) with $h^N(x)$ in place of $h(x)$.

5. Form the approximate sensitivity $s^N(x) = p^N(x) + h^N(x)$.

In order to validate the algorithm given above, we present a numerical result using the parameter set given in Table 3.1. Fig. 4.1 shows that the pointwise sensitivity error for the homogenization technique is on the order of 10^{-13} with only $N = 7$ equally spaced finite element nodes. Hence, the scheme yields very accurate sensitivity approximations even for a coarse grid size. Similar results are obtained for other parameter values.
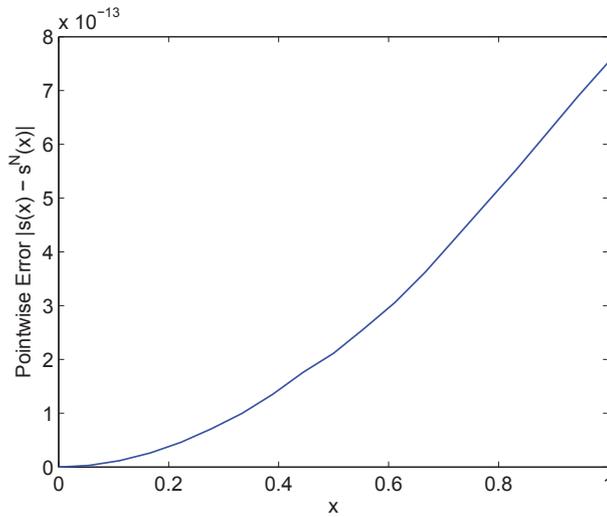

Fig. 4.1. The approximate pointwise sensitivity error for the homogenization approach using Hermite cubic finite elements with $N = 7$ equally spaced nodes and $\alpha = 0.5$. Note the scale on the vertical axis.

The cubic B-splines *should not* be used to approximate the state variable *if one is using this procedure for sensitivity calculations*. Since these basis functions have continuous second derivatives, the computed jump $d^N = w^N_{xx}(\alpha^+) - w^N_{xx}(\alpha^-)$ in step 2 of this process will always be zero and the homogenization method will fail.

Furthermore, recall that the second derivative of the state variable is discontinuous at $x = \alpha$ as long as $c_1 \neq c_2$, see Eq. (3.5). In particular,

$$w_{xx}(x; \alpha) = \begin{cases} c_1, & 0 < x < \alpha, \\ c_2, & \alpha < x < \ell. \end{cases} \tag{4.6}$$

The Hermite cubics capture these one-sided limits in the second derivative very accurately while the cubic B-splines do not. Recalling the computations for the state variable given in Section 3.1 and examining those calculations in greater detail, Fig. 4.2(a) shows the graph of the true state variable (3.5) and its cubic B-spline and Hermite cubic finite element approximations using 33 equally spaced nodes and the parameter value $\alpha = 0.5$. Also shown in Fig. 4.2(b) and
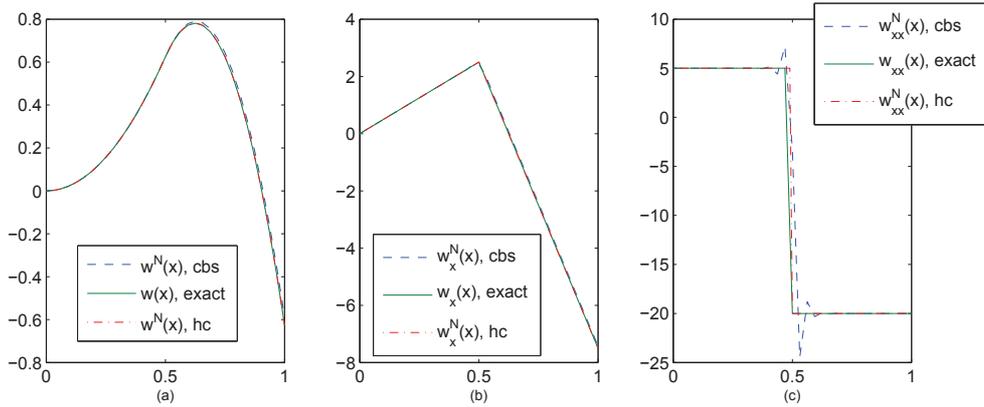
Fig. 4.2. *The exact solution* (3.5) *of the model equation* (3.3) *for* $\alpha = 0.5$ *compared with cubic B-spline* (cbs) *and Hermite cubic* (hc) *finite element approximations using 33 equally spaced nodes. Figure* (a) *compares the solution, and figures* (b) *and* (c) *compare the first and second derivatives, respectively.*
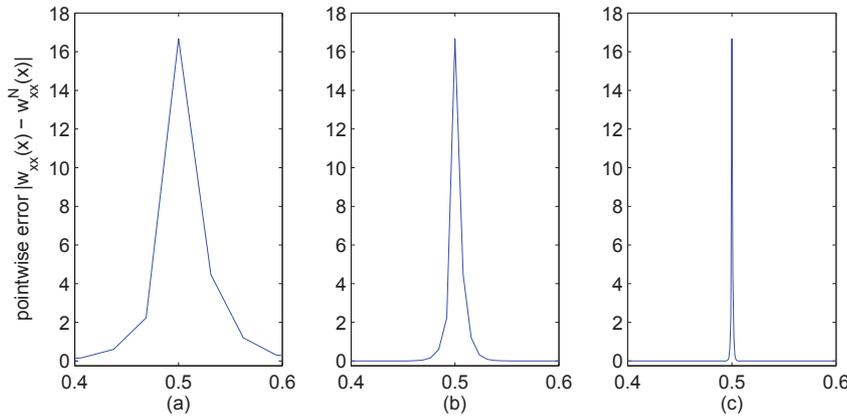

Fig. 4.3. *Pointwise error in the cubic B-spline finite element approximation of the second derivative of the exact solution* $(|w_{xx}(x; 0.5) - w_{xx}^N(x; 0.5)|)$ *for* (a) $N = 33$, (b) $N = 129$, *and* (c) $N = 1025$ *equally spaced finite element nodes. The error is essentially zero for all values of* x *not shown in the figures.*

Fig. 4.2(c) are the first and second derivatives, $w_x(x; \alpha)$ and $w_{xx}(x; \alpha)$, and their respective finite element approximations using 33 equally spaced nodes and $\alpha = 0.5$.

Fig. 4.2(a) shows that regardless of the choice of basis elements (cubic B-spline or Hermite cubic), each numerical approximation, $w^N(x; \alpha)$, converges to the true solution, $w(x; \alpha)$, as $N \to \infty$. Now notice that the discontinuity in the second derivative, $w_{xx}(x; 0.5)$, is large in magnitude compared to the size of the solution. As to be expected, the cubic B-spline finite element derivative approximation, $w_{xx}^N(x; 0.5)$, fails to accurately approximate the second derivative, $w_{xx}(x; 0.5)$, at the point of discontinuity since the second derivatives of the cubic B-splines are continuous. The typical *Gibbs* phenomena can be seen in Fig. 4.2(c). This can be seen more clearly if one examines the pointwise error. Fig. 4.3 shows a thin error "spike" remaining at the point of discontinuity even as the mesh is refined significantly. In contrast, the second derivatives of the Hermite cubic basis functions are discontinuous across adjacent

elements of the mesh, and this property allows one to capture that discontinuity in w_{xx} at $x = \alpha = 0.5$ very accurately. That is, $w_{xx}^N(\alpha^+) \to w_{xx}(\alpha^+)$ and $w_{xx}^N(\alpha^-) \to w_{xx}(\alpha^-)$ as $N \to \infty$, and the magnitude of the jump discontinuity, d, in step 2 of the numerical algorithm can be very accurately recovered by examining the jump discontinuity in w_{xx}^N at $x = \alpha = 0.5$ using the Hermite cubic elements.

Remark 4.2. In step 4 of the above algorithm, the finite element basis and corresponding system matrix from the state variable calculation can be re-used for the computation of p^N. One can define

$$p^N(x) = \sum_{j=1}^{N} b_j \phi_j(x), \tag{4.7}$$

and the linear system corresponding to (4.4) is given by

$$K^N \mathbf{b}^N = \mathbf{z}^N, \tag{4.8}$$

where $\mathbf{b}^N = [b_1, \cdots, b_N]^T$, the matrix K^N is given in (3.9), and the load vector on the right side of the equation is defined by

$$\mathbf{z}_i^N = -\int_0^\alpha EI_1 h_{xx}^N(x;\alpha)\phi_{i,xx}(x)\,dx - \int_\alpha^\ell EI_2 h_{xx}^N(x;\alpha)\phi_{i,xx}(x)\,dx, \tag{4.9}$$

for $i = 1, \cdots, N$. Only the load vector $\mathbf{z}$ must be constructed separately from the computations involved in setting up the linear system for the original model equation. This allows efficient computation of the sensitivity approximation. Although computational efficiency is not an important issue for the model problem considered here, these same ideas translate to large scale systems where computational efficiency is a requirement.

For more complex interface problems, approximating the solution of the continuous sensitivity equation (a PDE with nonhomogeneous interface conditions) may be a challenging task. It is natural to attempt simpler methods to compute the sensitivity. Below, we give an example of a "natural" method to approximate an interface sensitivity, and show that it fails to produce convergent approximations to the sensitivity with respect to the interface location for the model problem.

5. A Discretize-then-Differentiate Methodology for Sensitivity Computation

Applying one standard *Discretize-then-Differentiate* (DD) scheme, we implicitly differentiate the discretized state equations in (4.8) to obtain a linear finite dimensional equation for the approximate sensitivity. This approach assumes that it is reasonable and efficient to reuse the same finite dimensional subspace, V^N, to construct both a state variable approximation as well as a sensitivity approximation. This is a practical assumption to make if the practitioner is solving a large-scale problem where the underlying state equation represents a system of PDEs for which significant time and effort has been devoted to its discretization, or where one is using an existing legacy code.

We proceed as follows. Recall that the original state variable, $w = w(x;\alpha)$, depends explicitly on α. In the finite element approximations of the state, however, the basis functions were not constructed to depend on the parameter α. This is reasonable since, as was shown in

Section 3.1, the state can be accurately approximated without the basis functions depending on the parameter α. Therefore, it is natural to assume that the parameter dependence of the finite element approximation to the state is reflected in the sense that Eq. (3.7) is understood as

$$w^N(x; \alpha) = \sum_{j=1}^{N} a_j(\alpha)\phi_j(x). \tag{5.1}$$

Hence, we assume that the finite element basis functions are independent of α and that it is the coefficients, $a_j = a_j(\alpha)$, in the finite element discretization (5.1) which are influenced by small changes in the parameter value. This results in a sensitivity of the discretized state variable having the following form

$$s^N(x; \alpha) = \frac{\partial}{\partial \alpha} w^N(x; \alpha) = \sum_{j=1}^{N} \left[\frac{\partial}{\partial \alpha} a_j(\alpha) \right] \phi_j(x) = \sum_{j=1}^{N} b_j(\alpha)\phi_j(x), \tag{5.2}$$

where $b_j = \partial/\partial\alpha\, a_j$, for $j = 1, 2, \cdots, N$.

Recall that the coefficients $a_j(\alpha)$ in the state approximation (5.1) satisfy the linear system (4.8)

$$K^N(\alpha)\, \mathbf{a}^N(\alpha) = \mathbf{f}^N(\alpha),$$

where K^N and $\mathbf{f}^N$ are defined in (3.9) and we now emphasize the dependence of the terms on α. We implicitly differentiate through this equation with respect to α to derive an equation for $\mathbf{b}^N = \mathbf{a}^N_\alpha$. The entries of K^N and $\mathbf{f}^N$ are integrals that depend on the parameter α. These integrals are partitioned into integrals over $(0, \alpha)$ and (α, ℓ) and Leibniz' rule in (1.1) is applied to differentiate through these terms. We obtain the linear system

$$K^N \mathbf{b}^N + K^N_\alpha \mathbf{a}^N = \mathbf{f}^N_\alpha, \tag{5.3}$$

where K^N is defined in (3.9) and

$$[K^N_\alpha]_{ij}(\alpha) = EI_1\, \phi_{j,xx}(\alpha^-)\phi_{i,xx}(\alpha^-) - EI_2\, \phi_{j,xx}(\alpha^+)\phi_{i,xx}(\alpha^+), \tag{5.4}$$

$$\left[f^N_\alpha \right]_i(\alpha) = g_1\, \phi_{i,xx}(\alpha^-) - g_2\, \phi_{i,xx}(\alpha^+). \tag{5.5}$$

The discrete set of sensitivity equations takes the form of a linear system of equations where the unknowns are given by the coefficients b_j for $j = 1, \cdots, N$. Once the original approximating system (4.8) is solved for $\mathbf{a}^N$, the DD sensitivity equation (5.3) can be solved for $\mathbf{b}^N$. (Or, if desired, the two equations can be coupled together and solved simultaneously.)

As previously mentioned, the second derivatives of the Hermite cubic basis functions are discontinuous at the nodes; however, the second derivatives of the cubic B-spline functions are continuous across elements of the mesh. Hence, the one-sided limits in equations (5.4)−(5.5) must be handled carefully according to which basis functions are being used in the computations.

Cubic B-Splines: If each $\phi_{i,xx}$ is continuous across elements (as with the cubic B-splines), then $\phi_{i,xx}(\alpha^+) = \phi_{i,xx}(\alpha^-)$ for all i and the matrices in (5.4) - (5.5) take the form

$$[K^N_\alpha]_{ij}(\alpha) = (EI_1 - EI_2)\phi_{j,xx}(\alpha)\phi_{i,xx}(\alpha),$$
$$\left[f^N_\alpha \right]_i(\alpha) = (g_1 - g_2)\phi_{i,xx}(\alpha).$$

Hermite Cubics: Since the second derivatives of the Hermite cubic basis functions are discontinuous across the element nodes, then the one-sided limits reflected in Eqs. (5.4)−(5.5)

are evaluated using the one-sided limits of the second derivatives of the Hermite cubic basis functions.

With this in mind, one might expect to obtain different sensitivity approximations depending upon the type of basis function used for the computations. This is the case as we see in the following discussion.

5.1. DD sensitivity approximations

In this section, we discuss the sensitivity approximations defined in (5.2) and computed using Eqs. (5.3)-(5.5). The parameters used for the computations in this section are given in Table 3.1 of Section 3.1. Fig. 5.1 compares the exact sensitivity, $s(x; \alpha)$ with $\alpha = 0.5$, given by (3.6) with the approximate sensitivities produced by the DD scheme using both cubic B-spline and Hermite cubic basis functions (again with 33 equally spaced nodes and $\alpha = 0.5$). The numerical sensitivities for both types of basis functions fail to accurately approximate the true sensitivity. Table 5.1 reports the approximate L^{∞} error between the DD sensitivities and the exact sensitivity as the number of equally spaced finite element nodes is increased. As the mesh is refined the error does not decrease in either case and the DD sensitivities each converge to erroneous sensitivities. Similar results were observed for a wide variety of parameter values.

The graph of the cubic B-spline DD sensitivity in Fig. 5.1 has the same character as that of the graph of the true sensitivity; however, the approximation error is significant and $s^N(x; \alpha) \nrightarrow s(x; \alpha)$ as $N \to \infty$. The Hermite cubic DD sensitivity is zero everywhere (to finite precision arithmetic) and completely fails both to capture the general behavior of the true sensitivity and to converge to the true sensitivity.

It is interesting to note the vastly different results produced by the DD approach for two similar approximation schemes where the only difference is in the choice of finite element basis functions. Recall that the main difference in the basis functions is that the cubic B-splines possess second derivatives which are continuous across elements while the Hermite cubics do not. Noting that the second derivatives of the basis functions appear in the right side of the
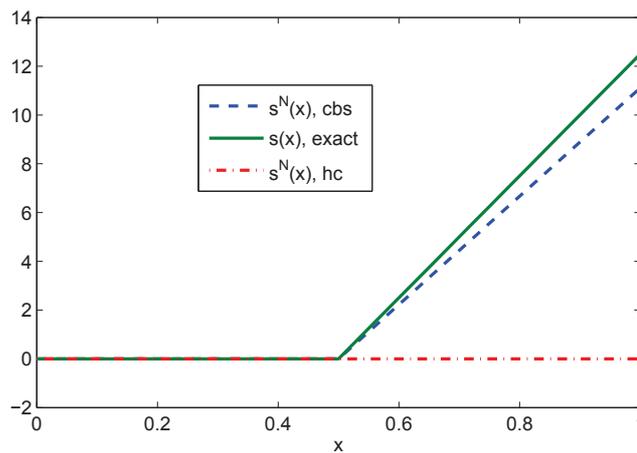

Fig. 5.1. The exact sensitivity given in (3.6) for $\alpha = 0.5$ compared with the approximate sensitivities produced by DD using both cubic B-spline (cbs) and Hermite cubic (hc) basis functions.

Table 5.1: The approximate L^∞ error for the DD sensitivity approximations using the cubic B-spline and Hermite cubic basis functions at various values of N, the number of equally spaced finite element nodes.

N	9	17	33	65	129	1025
L^∞ error (cubic B-spline)	1.3889	1.3889	1.3889	1.3889	1.3889	1.3889
L^∞ error (Hermite cubic)	12.5	12.5	12.5	12.5	12.5	12.5

sensitivity equation in (5.3)-(5.5), it is natural to assume that the sensitivity computations obtained using the different types of basis functions should be different. However, it may not be obvious why both schemes fail to converge to the true sensitivity as the mesh is refined. Also, one might expect to incur some computational errors due to the appearance of the approximate state solution coefficient vector, $\mathbf{a}^N$, in the DD sensitivity equation (5.3). The cause of the failure of the DD approach for this problem becomes more apparent and is thoroughly discussed once we examine a formal continuous sensitivity equation for the model problem.

6. A Formal Continuous Sensitivity Equation

This section presents a formal Continuous Sensitivity Equation Method (CSEM) approach to computing the sensitivity of interest for the model problem. This approach is presented to contrast with the DD approach given in Section 5. Instead of first discretizing the variational problem and then differentiating the resulting set of linear equations, we now formally differentiate the variational form of the model problem with respect to the parameter α in order to derive the formal continuous sensitivity equation. We show the resulting equation is ill-posed, and use it to explain the failure of the discretize-then-differentiate approach.

Recall the weak form of the model problem (3.3): find $w \in V = H_L^2(0,\ell)$ satisfying

$$\int_0^\ell EI(x;\alpha)w_{xx}(x)\phi_{xx}(x)\,dx = \int_0^\ell g(x;\alpha)\phi_{xx}(x)\,dx$$

for all $\phi \in V$. We differentiate this equation with respect to α by partitioning the integrals over $(0,\ell)$ into integrals over $(0,\alpha)$ and (α,ℓ) and *formally* applying Leibniz' rule to obtain the following variational sensitivity equation: find $s \in V = H_L^2(0,\ell)$ such that

$$\int_0^\ell EI(x;\alpha)s_{xx}(x)\phi_{xx}(x)\,dx + EI_1 w_{xx}(\alpha^-)\phi_{xx}(\alpha^-) - EI_2 w_{xx}(\alpha^+)\phi_{xx}(\alpha^+)$$
$$= g_1\phi_{xx}(\alpha^-) - g_2\phi_{xx}(\alpha^+) \tag{6.1}$$

for all $\phi \in V$. Again, we have replaced the function evaluations at α appearing due to Leibniz' rule with the one-sided limits to account for possible discontinuities.

The test functions ϕ are in $V = H_L^2(0,\ell)$, and so ϕ_{xx} is only guaranteed to be in $L^2(0,\ell)$; thus the pointwise evaluations given by the one-sided limits, $\phi_{xx}(\alpha^+)$ and $\phi_{xx}(\alpha^-)$, are not well-defined for a general $\phi \in V$. Therefore, this formal continuous sensitivity equation is not properly posed over $V = H_L^2(0,\ell)$ even though the variational form of the state equation (3.3) is well posed over V. This makes sense if we remember that the exact solution of the state equation (3.5) is in V, yet the exact sensitivity (3.6) is not in V.

We may further explain the ill-posedness of this equation as follows. If we ignore the formal nature of the sensitivity equation and substitute the true values of $w_{xx}(\alpha^+)$ and $w_{xx}(\alpha^-)$ into

the variational form, then the equation has a solution; however, the solution is actually the zero function and NOT the true sensitivity $s(x; \alpha)$ given in Eq. (3.6). This can be seen by substituting the exact solution of the state equation, given by (3.5), into (6.1). Group the like terms in the sensitivity equation to obtain

$$\int_0^\ell EI(x; \alpha) s_{xx}(x) \phi_{xx}(x) \, dx$$
$$= \Big[g_1 - EI_1 w_{xx}(\alpha^-) \Big] \phi_{xx}(\alpha^-) - \Big[g_2 - EI_2 w_{xx}(\alpha^+) \Big] \phi_{xx}(\alpha^+). \tag{6.2}$$

Using the exact values, $w_{xx}(\alpha^-) = c_1 = g_1/EI_1$ and $w_{xx}(\alpha^+) = c_2 = g_2/EI_2$, this equation reduces to one of finding $s \in V$ satisfying

$$\int_0^\ell EI(x; \alpha) s_{xx}(x) \phi_{xx}(x) \, dx = 0,$$

for all $\phi \in V$. Since the associated linear operator corresponding to the left side of this equation is invertible, the sensitivity $s(x)$ must be identically zero. Hence, this does not capture the true sensitivity given in Eq. (3.6).

Although the pointwise evaluations $\phi_{xx}(\alpha^+)$ and $\phi_{xx}(\alpha^-)$ do not make sense for a general $\phi \in V$, they are well defined for the cubic B-spline and Hermite cubic finite element basis functions discussed above. Below, we proceed with a numerical approximation scheme for this ill-posed problem to explain the failure of the discretize-then-differentiate approach.

6.1. Numerical approximation schemes

The discretization for the formal continuous sensitivity equation is given in this section. First, we clearly see from the left side of Eq. (6.1) that an approximation to the second derivative of the state variable, w_{xx}, is required. A natural approach is to use the finite element approximations given in (3.7) and (4.8) to obtain these approximations. The one-sided limits in (6.1) are then approximated by

$$w_{xx}(\alpha^+) \approx w_{xx}^N(\alpha^+) = \sum_{j=1}^N a_j \phi_{j,xx}(\alpha^+), \quad \text{and} \quad w_{xx}(\alpha^-) \approx w_{xx}^N(\alpha^-) = \sum_{j=1}^N a_j \phi_{j,xx}(\alpha^-). \tag{6.3}$$

Here, $\phi_j \in V^N$ is a basis function from the finite element discretization of the original model problem. In order to parallel the approximation schemes of Section 5, we discretize the sensitivity equation using both cubic B-spline and Hermite cubic basis functions. In particular, we make use of $V^N = \text{span}\{\phi_j\}_{j=1}^N$, where ϕ_j is either the jth cubic B-spline or Hermite cubic finite element basis function. Note that if cubic B-splines are used, then $\phi_{j,xx}(\alpha^+) = \phi_{j,xx}(\alpha^-)$ for all $j = 1, \cdots, N$ leading to $w_{xx}^N(\alpha^+) = w_{xx}^N(\alpha^-)$.

For the sensitivity approximation, we again look for an approximate sensitivity $s^N \in V^N \subset V$ defined by the linear combination

$$s^N(x) = \sum_{j=1}^N \tilde{b}_j \phi_j(x).$$

Substituting this into (6.1) and taking $\phi = \phi_i$ for $i = 1, \cdots, N$ yields the approximating linear system

$$K^N \tilde{\mathbf{b}}^N + \mathbf{h}_1^N w_{xx}^N(\alpha^-) + \mathbf{h}_2^N w_{xx}^N(\alpha^+) = \mathbf{f}_\alpha^N, \tag{6.4}$$

where $\tilde{\mathbf{b}}^N = [\tilde{b}_1, \cdots, \tilde{b}_N]^T$, K^N and $\mathbf{f}_\alpha^N$ are defined in (3.9) and (5.5), respectively, and

$$[\mathbf{h}_1^N]_i = EI_1\, \phi_{i,xx}(\alpha^-), \quad [\mathbf{h}_2^N]_i = -EI_2\, \phi_{i,xx}(\alpha^+).$$

Hence, Eq. (6.4) yields a system of linear equations that can be solved in order to compute sensitivity approximations.

We use the same number of finite element basis functions to approximate both the second order derivative of the state and the solution of the formal continuous sensitivity equation. One could use a different number of finite element basis functions or a different basis in the sensitivity solves; however, we use the same basis for simplicity.

It is important to note that substituting in the approximations to the second derivative of the state in (6.3) shows that $\tilde{\mathbf{b}}^N$ satisfies

$$K^N \tilde{\mathbf{b}}^N + K_\alpha^N \mathbf{a}^N = \mathbf{f}_\alpha^N,$$

where K_α^N is defined in (5.5). That is, this equation is identical to the DD sensitivity equation in (5.3). Therefore in this case, the approximate sensitivities produced by the DD and the formal CSEM approach **are the same**.

6.2. An explanation of the failure of the DD scheme

We may now explain the failure of the discretize-then-differentiate (DD) scheme, and also point out why using two different finite element basis functions gave such different results. We showed above that a particular discretization of the formal continuous sensitivity equation resulted in the same sensitivity approximations produced by the Discretize-then-Differentiate method. It follows that the DD sensitivity approximations are actually approximations to the "solution" of the formal continuous sensitivity equation. Since the formal continuous sensitivity equation is not well posed, the DD approximations produced erroneous sensitivities. *Thus, an ill posed formally derived continuous sensitivity equation is a warning that approximations produced by a Discretize-then-Differentiate procedure are most likely completely false.*

Furthermore, the discretization of the formal sensitivity equation required approximations of the one-sided limits of the second derivative of the state at the interface. As discussed in Section 4.2, accurate finite element approximations to these quantities can be obtained using Hermite cubic basis functions, while cubic B-splines give extremely poor approximations. Since the discretization of the formal sensitivity equation gives the same results as the DD sensitivity approximations, the Hermite cubic DD sensitivity approximates the "exact solution" of the formal continuous sensitivity equation (the zero function), while the cubic B-spline DD sensitivity is a bad approximation to this "exact solution" due to the large error in the second derivative state approximation. Of course, in the end both approximations converge to incorrect sensitivities as $N \to \infty$.

7. A Second Order Example Problem

The fourth order model problem considered above was motivated by the beam equation in Section 2. Many other applications give rise to second order interface problems. In this section, we briefly consider the homogenization method for computing the interface sensitivity for a second order model problem.

Consider the 1D model interface problem from [7]: find $w(x)$ satisfying

$$- (\kappa(x; \alpha)w_x(x))_x = 0, \tag{7.1a}$$

$$w(0) = 0, \quad w(1) = 1, \tag{7.1b}$$

$$w(\alpha^-) = w(\alpha^+), \tag{7.1c}$$

$$(\kappa w_x)(\alpha^-) = (\kappa w_x)(\alpha^+). \tag{7.1d}$$

As before, the coefficient function $\kappa(x; \alpha)$ is piecewise constant and is given by

$$\kappa(x; \alpha) = \begin{cases} \kappa_1, & 0 < x < \alpha, \\ \kappa_2, & \alpha < x < 1, \end{cases} \tag{7.2}$$

where κ_1 and κ_2 are positive real constants.

Differentiating through this problem with respect to α yields the following interface problem for the sensitivity $s(x; \alpha) = (d/d\alpha)w(x; \alpha)$:

$$- (\kappa(x; \alpha)s_x(x; \alpha))_x = 0, \tag{7.3a}$$

$$s(0) = 0, \quad s(1) = 0, \tag{7.3b}$$

$$s(\alpha^-) - s(\alpha^+) = d, \tag{7.3c}$$

$$(\kappa s_x)(\alpha^-) - (\kappa s_x)(\alpha^+) = 0, \tag{7.3d}$$

where d is the jump in the first derivative of w, i.e., $d = w_x(\alpha^+) - w_x(\alpha^-)$.

In [7], this problem was formulated in a very weak sense in order to obtain approximate solutions in $L^2(0, 1)$ that satisfied the nonhomogeneous jump condition (7.3c). Here, we apply the homogenization approach used previously for the fourth order problem. Similar to the approach in Section 4.1, we select any function $h(x)$ that is H^1 on the intervals $0 < x < \alpha$ and $\alpha < x < 1$ satisfying (1) the essential boundary conditions $h(0) = 0$ and $h(1) = 0$, and (2) the essential interface condition $h(\alpha^-) - h(\alpha^+) = d$. Then we make the change of variable $s = p + h$. The unknown function p satisfies the zero boundary conditions, and it also satisfies the homogeneous interface condition $p(\alpha^-) = p(\alpha^+)$.

Next, multiplying the strong interface problem (7.3a) by a test function $\phi \in V = H_0^1(0, 1)$, integrating over the intervals $(0, \alpha)$ and (α, ℓ), substituting $s = p + h$, and integrating by parts gives the variational problem: find $p \in V$ such that

$$\int_0^1 \kappa p_x \phi_x \, dx = - \int_0^\alpha \kappa_1 h_x \phi_x \, dx - \int_\alpha^1 \kappa_2 h_x \phi_x \, dx,$$

for all $\phi \in V$. To approximate the sensitivity s, we proceed as with the fourth order problem: first approximate w and the jump d, then approximate p (e.g., using finite elements), and lastly set $s = p + h$.

As in our previous model problem, this method succeeds in recovering the exact sensitivity as long as the true value of the jump parameter d is known and used explicitly in the calculation. For example, let $\kappa_1 = 1$ and $\kappa_2 = 2$. Taking $h(x) = (d/\alpha)x$ for $0 < x < \alpha$ and $h(x) = 0$ for $\alpha < x < 1$, it can be checked that the exact solution of the above variational problem is

$$p(x) = \begin{cases} \left(\frac{2d}{\alpha+1} - \frac{d}{\alpha}\right)x, & 0 < x < \alpha, \\ \frac{d}{\alpha+1}(x - 1), & \alpha < x < 1. \end{cases}$$

For the exact value of the jump, $d = -1/(\alpha + 1)$, we recover the exact sensitivity given in [7]:

$$s(x) = p(x) + h(x) = \begin{cases} -2x/(\alpha+1)^2, & 0 < x < \alpha, \\ -(x-1)/(\alpha+1)^2, & \alpha < x < 1. \end{cases}$$

8. Conclusion

This paper uses a simple fourth order interface model problem to illustrate the computational issues that can arise when computing sensitivities with respect to a parameter that determines an interface location. Since accurate numerical approximations to the original state variable can be obtained using standard finite element techniques, one may be inclined to apply those same techniques to the sensitivity approximations. The example in this paper should serve as a cautionary tale that points out the pitfalls that can arise when approaching the problem of sensitivity computation in an ad hoc manner as well as the benefits of performing a rigorous mathematical analysis to identify the smoothness properties of the state and sensitivity variables prior to constructing an algorithm for sensitivity computation. We emphasize again that the key to the success of the true CSEM discussed in Section 4 is that it accounts for the lack of smoothness of the governing equations with respect to the parameter of interest. The regularity of the sensitivity equation is not the same as that of the original state equation. The Discretize-then-Differentiate method does not take such regularity issues into account, and it completely fails to accurately approximate the true sensitivity.

Moreover, the coupling between the interface conditions of the sensitivity equation and those of the state equation must be treated carefully. For both the second order and fourth order models discussed here, one of the sensitivity interface conditions depends explicitly on an interface *jump* condition that the original state variable satisfies. (For these examples, the jump conditions involve information from w_x and w_{xx} at the interface.) The value of the jump, labeled d in this paper, must be known analytically or must be accurately approximated in order to construct an accurate homogenization function h. Even for the simple model problem discussed in Section 4, we observe that the choice of basis functions in the finite element computation of the state variable drastically affects accuracy and convergence of sensitivity approximations through the accuracy (or lack thereof) in the approximation of the value for the jump condition d. As noted in Section 4.2, the numerical scheme one chooses for solving the original state equation also must be able to yield accurate interface **jump** conditions for state gradients when sensitivity calculations are to follow. This refers to Step 2 of the algorithm outlined at the beginning of Section 4.2 and the comments directly following the algorithm. From a numerical perspective, the fourth order model demonstrates that accurate sensitivity calculations obtained by using the homogenization procedure rely on the convergence of the numerical approximations of the state variable in an appropriate sense; more specifically, the second derivative approximation must converge pointwise, except at the interface. This allows one to accurately approximate the jump condition d that occurs at the interface. Although it isn't explicitly stated in Section 7, pointwise convergence except at the interface is required in the approximations of the first derivative of the state variable, when the approximate value of the jump d is used for the homogenization procedure. In any case, the convergence requirements for the numerical scheme chosen for state variable approximation are most clearly identified once the correct sensitivity equation, including all relevant interface conditions, is formulated.

A problem to be considered in future work is to compute sensitivities with respect to interface locations in higher dimensional problems. Again, we expect the proper formulation of

the sensitivity equation to be crucial to obtaining accurate approximations. However, there are many issues to be addressed. First, obtaining accurate simulations of higher dimensional interface problems is not a simple task. For second order elliptic interface problems, it is well known that standard finite element approximations can converge slowly [18]. Optimal convergence rates can be restored if the finite element mesh is aligned to the interface [19–21]. For complicated interface geometries the construction of such a mesh may be very difficult; therefore, many researchers have developed numerical methods to treat interface problems. For examples, see [17] and the references therein; see also [22] for a Hermite cubic immersed finite element method for fourth order problems.

Next, we expect that the sensitivity equation will again contain nonhomogeneous interface conditions. It is important to note that the homogenization method applied here to such a problem can be much more difficult in higher spatial dimensions. In many cases, the homogenization function h will not be known analytically and must be constructed as described in [16, 17]. As noted earlier, the construction of h may require accurate state gradient information along the interface, and the numerical scheme for state variable approximation must be chosen with that in mind. Also, for time varying problems, h will be time dependent and so the forcing function for the homogenized sensitivity equation will have to be reconstructed at each time step. It may be beneficial to use other special numerical methods for nonhomogeneous interface problems, e.g., see [17] and the references therein. Some of these methods also may require an initial homogenization step. Nitsche's method (as in [23, 24]) may be a viable alternative method since no homogenization is necessary; for this method the nonhomogeneous interface conditions are satisfied weakly. Regardless of the chosen technique, proper formulation of the sensitivity equation as an interface problem will be the key to constructing an accurate, efficient, and convergent numerical scheme.

References

[1] D.G. Cacuci, Sensitivity and uncertainty analysis. Vol. I, Chapman & Hall/CRC, Boca Raton, FL, 2003.

[2] J. Borggaard and J. Burns, A PDE sensitivity equation method for optimal aerodynamic design, *J. Comput. Phys.*, **136** (1997), 366–384.

[3] L.G. Stanley, Computational Methods for Sensitivity Analysis with Applications to Elliptic Boundary Value Problems, PhD thesis, Department of Mathematics, Virginia Polytechnic Institute and State University, 1999.

[4] D.L. Stewart, Numerical Methods for Accurate Computation of Design Sensitivities, PhD thesis, Department of Mathematics, Virginia Polytechnic Institute and State University, 1998.

[5] A.G. Godfrey, Using sensitivities for flow analysis, Computational Methods for Optimal Design and Control, pages 181–196, Birkhäuser, Boston, MA, 1998.

[6] J. Borggaard, D. Pelletier and É. Turgeon, Parametric uncertainty analysis for thermal fluid calculations, Proceedings of the Third World Congress of Nonlinear Analysts, Part 7 (Catania, 2000), volume 47, pages 4533–4543, 2001.

[7] J.A. Burns, T. Lin and L.G. Stanley, A Petrov Galerkin finite-element method for interface problems arising in sensitivity computations, *Comput. Math. Appl.*, **49**:11-12 (2005), 1889–1903.

[8] L.G. Stanley, Sensitivity analysis for actuator placement on an Euler-Bernoulli beam, Proceedings of the 43rd IEEE Conference on Decision and Control, pages 1532 – 1537, 2004.

[9] H.T. Banks, K. Ito and B.B. King, Theoretical and computational aspects of feedback in structural systems with piezoceramic controllers, Technical Report CRSC-TR93-2, Center for Research in Scientific Computation, North Carolina State University, 1993.

[10] H.T. Banks, R.C. Smith and Y. Wang, The modeling of piezoceramic patch interactions with shells, plates, and beams, *Quarterly of Applied Mathematics*, **53**:2 (1995), 353–381.

[11] H.T. Banks, R.C. Smith and Y. Wang, Smart Material Structures: Modeling, Estimation and Control, John Wiley & Sons, New York, 1996.

[12] P. Šolín, Partial Differential Equations and the Finite Element Method, Wiley, Hoboken, NJ, 2006.

[13] K. Höllig, Finite Element Methods with B-Splines, SIAM, Philadelphia, PA, 2003.

[14] P.M. Prenter, Splines and Variational Methods, John Wiley & Sons, New York, 1989.

[15] G.J. Fix and G. Strang, An Analysis of the Finite Element Method, Prentice-Hall, Englewood Cliffs, N.J., 1973.

[16] Y. Gong, B. Li and Z. Li, Immersed-interface finite-element methods for elliptic interface problems with nonhomogeneous jump conditions, *SIAM J. Numer. Anal.*, **46**:1 (2007/08), 472–495.

[17] Z. Li and K. Ito, The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains, SIAM, Philadelphia, PA, 2006.

[18] V.J. Rivkind, On an estimate of the rapidity of convergence of homogeneous difference schemes for elliptic and parabolic equations with discontinuous coefficients, Problems in Mathematical Analysis. Vol. 1. Boundary Value Problems and Integral Equations, Edited by V. I. Smirnov, translated from the Russian, pages 102–110, Consultants Bureau, New York, 1968.

[19] J.H. Bramble and J.T. King, A finite element method for interface problems in domains with smooth boundaries and interfaces, *Adv. Comput. Math.*, **6**:2 (1996), 109–138 (1997).

[20] M. Plum and C. Wieners, Optimal a priori estimates for interface problems, *Numer. Math.*, **95**:4 (2003), 735–759.

[21] R.K. Sinha and B. Deka, On the convergence of finite element method for second order elliptic interface problems, *Numer. Funct. Anal. Opt.*, **27**:1 (2006), 99–115.

[22] T.S. Wang, A Hermite Cubic Immersed Finite Element Space for Beam Design Problems, Master's thesis, Department of Mathematics, Virginia Polytechnic Institute and State University, 2005.

[23] A. Hansbo and P. Hansbo, An unfitted finite element method, based on Nitsche's method, for elliptic interface problems, *Comput. Methods Appl. M.*, **191**:47-48 (2002), 5537–5552.

[24] P. Hansbo, Nitsche's method for interface problems in computational mechanics, *GAMM-Mitt.*, **28**:2 (2005), 183–206.