Local Continuum Sensitivity Method for Shape Design Derivatives Using Spatial Gradient Reconstruction

David M. Cross

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Aerospace Engineering

Robert A. Canfield, Chair
Mayuresh J. Patil
Muhammad R. Hajj
Ned J. Lindsley

April 24, 2014
Blacksburg, Virginia

Keywords: Continuum Sensitivity, Shape Sensitivity, Shape Optimization, Aeroelasticity, Fluid-structure Interaction

Local Continuum Sensitivity Method for Shape Design Derivatives Using Spatial Gradient Reconstruction

David M. Cross

(ABSTRACT)

Novel aircraft configurations tend to be sized by physical phenomena that are largely neglected during conventional fixed wing aircraft design. High-fidelity fluid-structure interaction that accurately models geometric nonlinerity during a transient aeroelastic gust response is critical for sizing the aircraft configuration early in the design process. The primary motivation of this research is to develop a continuum shape sensitivity method that can support gradient-based design optimization of practical and multidisciplinary high-fidelity analyses. A local continuum sensitivity analysis (CSA) that utilizes spatial gradient reconstruction (SGR) and avoids mesh sensitivities is presented for shape design derivative calculations. Current design sensitivity analysis (DSA) methods have shortcomings regarding accuracy, efficiency, and ease of implementation. The local CSA method with SGR is a nonintrusive and element agnostic method that can be used with black box analysis tools, making it relatively easy to implement. Furthermore, it overcomes many of the accuracy issues documented in the current literature. The method is developed to compute design derivatives for a variety of applications, including linear and nonlinear static beam bending, linear and nonlinear transient gust analysis of a 2-D beam structure, linear and nonlinear static bending of rectangular plates, linear and nonlinear static bending of a beam-stiffened plate, and two-dimensional potential flow. The analyses are conducted using general purpose codes. For each example the design derivatives are validated with either analytic or finite difference solutions and practical numerical and modeling considerations are discussed. The local continuum shape sensitivity method with spatial gradient reconstruction is an accurate analytic design sensitivity method that is amenable to general purpose codes and black box tools.

Attribution

This document contains material that appear in thre separate manuscripts co-authored by Robert A. Canfield. His contributions include strategic advice and review of the articales for technical accuracy, completeness, and grammatical correctness.

Dedication

This dissertation is dedicated to my beloved wife, Anna

and to our family and dog

for their endless love, support, and encouragement.

Acknowledgments

First and foremost, I would like to thank my graduate advisor, Dr. Robert Canfield. I am deeply grateful for his support, guidance, encouragement, and always present optimism. The research presented in this dissertation would not have been possible without his knowledge, experience, and enthusiasm. In addition to being a phenominal advisor, he is also a terrific mentor. Working with him has been a wonderful experience, one that has made me not only a better academic student, but a better life student.

I would also like to thank my PhD committee members, Dr. Mayuresh Patil, Dr. Muhammad Hajj, and Dr. Ned Lindsley. The insightful feedback and guidance that they have provided has been invaluable, and I am grateful for all of the time that they have so willingly committed over the years. I feel privelaged to have had such long standing relationships with each of them.

I would like to thank the Aerospace and Ocean Engineering Department and the Virginia Tech community for being my home the last nine years. I am especially thankful to my office-mates and design sensitivity mates, Anthony Ricciardi, Shaobin Liu, and Mandar Kulkarni, for always providing insightful discussion and of course, comic relief.

Lastly, I would like to thank my family for their constant love and support, especially my wife; you are my inspiration.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research Motivation

One of the fundamental objectives of aircraft design is to achieve optimal performance at a minimal cost. Therefore, it is important that the designers maintain an accurate picture of both cost and performance throughout the design process. In fact, decisions made early in the design process can have a dramatic effect on cost and performance. Empirical and low fidelity models are commonly used during the early and conceptual design stages. Critical design decisions, which are made at these early stages of the design process, can be improved by utilizing high fidelity models and design optimization. However, using empirical and low fidelity models often remains more economical, because design optimization that uses high fidelity models can be computationally expensive.

One example of this is NASA's Helios, a high-altitude, long endurance (HALE) configuration depicted in Figure 1.1. After it crashed in 2003, the investigating board "determined that the mishap resulted from the inability to predict, using available analysis methods, the aircraft's increased sensitivity to atmospheric disturbances such as turbulence, following vehicle configuration changes required for the long-duration flight demonstration" (NASA, 2004). The available analyses were unable to properly model the transient aeroelastic gust response as it entered the nonlinear regime. This incident proved that highly flexible aircraft configurations are very sensitive to gust loads. Furthermore, it demonstrated that novel aircraft configurations, which operate in such nonlinear regimes, require the use of high fidelity modeling early in the design process, when critical design decisions are made and preliminary sizing of the configuration takes place. (Ricciardi, A.P. et al., 2011)

Figure 1.1: NASA's Helios prototype (NASA, 2004)

Lucia (2005) states that developing computational tools for fluid-structure interaction (FSI) that accurately model nonlinear behavior is necessary to extend the design envelope to include aircraft configurations which "thrive in nonlinear regimes." A good example is the next generation Sensorcraft (Figure 1.2), which the Air Force is studying as a potential Global Hawk replacement (Johnson, 2001). Studies by the Air Force Research Laboratory (AFRL) (Lucia, 2005; Blair, M. et al., 2005) and The Boeing Company (Omar, 2003) concluded that aeroelastic gust loads are a critical design condition of the Sensorcraft mission. Furthermore, a lightweight, joined-wing configuration, such as the Sensorcraft, can exhibit nonlinear behavior due to gust loads. These effects also exist for configurations other than large HALE aircraft. For instance, micro-air vehicles (MAVs) are also susceptible to large displacements and rigid body motion due to gust loads.

Figure 1.2: Boeing Sensorcraft configuration (Johnson, 2001)

There is a clear necessity to provide a high fidelity FSI capability that can model nonlinear transient aeroelastic gust responses. As previously mentioned, utilizing such a model for design optimization can be computationally expensive. Sobieszczanski-Sobiesk, J. and Haftka, R.T. (1997) point out that multidisciplinary design and shape optimization of FSI is very challenging. They identify sensitivities (design derivatives) of the gust response as a critical component of the optimization, because they are required by gradient-based algorithms and they can help guide design decisions. However, computing the design derivatives can be one of the most computationally expensive steps in shape design optimization. Developing an efficient approach to design sensitivity analysis (DSA) for computational FSI models can go a long way to reducing the design cycle time. This research is motivated by the need to have an accurate and efficient DSA approach that can enable aeroelastic shape design optimization of future flexible aircraft using high fidelity models.

Existing DSA methods have certain limitations when applied to high fidelity models or general purpose analysis codes, summarized in Sections 1.2 and 1.3 and detailed in Chapter 2. The DSA method developed and proposed in this research aims to overcome each of these limitations. In particular, it is specifically formulated as a general approach that can be easily and readily implemented with general purpose codes for low, medium, or high fidelity models with controllable accuracy.

1.2 Overview of Design Sensitivity Analysis Methods

Gradient-based algorithms are popularly used for shape design optimization of high fidelity structures and aeroelastic structures. DSA has been a popular field of research, because design sensitivities, also called design derivatives, play such a vital role in gradient-based optimization. Therefore, many different approaches to DSA have been developed over the years. As depicted in Figure 1.3, they can classified into three basic categories: numeric, analytic, and automatic differentiation (AD).

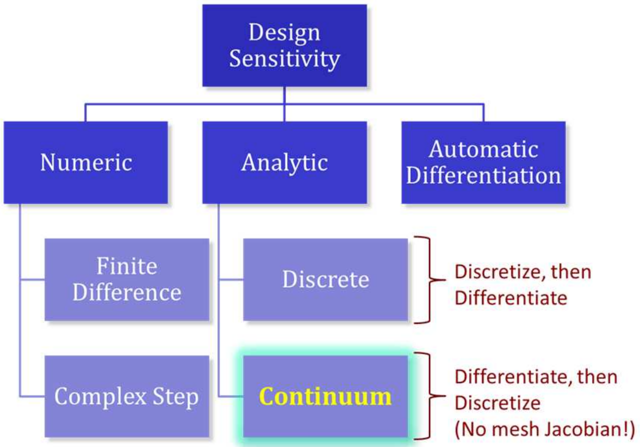


Figure 1.3: A taxonomy of design sensitivity methods

The finite difference and complex step formulations are classified as numerical methods. The finite difference method is a general formulation of the design derivatives, and therefore, quite easy to implement. However, for large systems with many design variables, it becomes computationally expensive. Furthermore, it can be unpredictably inaccurate, because the correct step size must be chosen in order to balance truncation and round-off errors (Borggaard, J. and Burns, J., 1997; Livne, 1994). In contrast, the complex step method is typically very accurate, but it requires that the analysis tool be able to handle complex arguments and arithmetic (Squire, W. and Trapp, G., 1998; Martins, 2003). Many general purpose codes, which are not intended for use with complex inputs, such as Nastran, make implementation of the complex step method infeasible. In addition, both the finite difference and complex step methods can be computationally expensive for nonlinear problems and when there are a large number of design variables.

Automatic differentiation is a very accurate DSA method, but it requires access to all of the source code. In many instances, the source code of general purpose codes is not available, rendering AD infeasible. There have been many successful implementations when access to the source code has been possible. Carle, A. et al. (1998) and Bischof, C.H. et al. (2007) utilized AD to obtain design derivatives of results from general purpose computational fluid dynamics (CFD) codes. It has also been utilized to calculate the design derivatives for flutter and aeroelastic responses (Kapania, R.K. et al., 1993; Issac, J.C. et al., 1995; Issac, J.C. and Kapania, R.K., 1997). Since the numerical methods and AD each have significant limitations regarding accuracy or implementation, a great deal of research has focused on analytic DSA methods. However, they too have limitations of varying degrees.

The numerical methods solve for the design derivatives using approximate equations, whereas the analytical methods aim to solve analytical equations that govern the design derivatives (sensitivity equations). Analytic methods can be further categorized based on how the sensitivity equations are derived and solved. Typically, the original analysis is solved using an approximate method that discretizes the governing equations. For the discrete analytic method, the sensitivity equations are obtained by differentiating the discretized governing equations with respect to the design parameters. The discrete analytic method can be an accurate and effective means to computing design derivatives, but all details of the analysis procedure must be known. That includes the governing equations, the discretization method, integration techniques, and shape functions (Haftka, R. and Gurdal, Z., 1992). It is not reasonable to expect that this information will always be available for general purpose codes. Hence, the discrete analytic method is an intrusive method. Therefore, a popular alternative is the semi-analytic method. This method utilizes the same sensitivity equations as the discrete analytic method, but design derivative terms that appear in the applied loads are approximated using finite differences. This method only requires access to the system matrices after the analysis procedures have taken place. Therefore, the semi-analytic method is a general formulation of the design derivatives and is quite easy to implement. However, Barthelemy, B. and Haftka, R.T. (1988) and Haftka, R. and Gurdal, Z. (1992) demonstrated that the semi-analytic method can be highly inaccurate for shape design derivatives.

Continuum sensitivity analysis (CSA) takes a slightly different approach than the discrete analytic method. The main difference is that the sensitivity equations are derived by differentiating the governing equations before they are discretized. This results in linear continuous (continuum) sensitivity equations (CSEs), which govern the design derivative solution. In past literature, the CSA method has also been called variational shape design (Haug, E.J. et al., 1986), variational sensitivity analysis (Haftka, R. and Gurdal, Z., 1992), and the continuous sensitivity equation method (Borggaard, J. and Burns, J., 1994, 1997). When deriving the CSEs, the governing equations of the analysis can be differentiated using either partial or total differentiation. Using partial or total differentiation yields CSEs that are posed in terms of local or total design derivatives, respectively. As CSA was developed over the years, the fluids camp typically derived and solved local CSEs, because local design derivatives are consistent with an Eulerian reference frame (Borggaard, J. and Burns, J., 1997; Etienne, S. and Pelletier, D., 2005). In contrast, the structures camp typically derived and solved total CSEs, because total design derivatives are consistent with material derivatives and a Lagrangian reference frame (Haug, E.J. et al., 1986; Choi, K. and Kim, N.H., 2005). However, either local or total CSA can be used with fluids and structures applications, because, if necessary, local and total design derivatives can be transformed from one to the other using the definition of the material derivative.

Haftka, R. and Gurdal, Z. (1992) state that the CSEs can be solved using an analysis tool different than the one used to solve the original boundary value problem, which means the CSEs can be solved using a different discretization method and different shape functions. However, using different numerical techniques can result in a design derivative solution that is inconsistent with the analysis solution. To mitigate this, the same numerical techniques should be used, which, according to Borggaard, J. and Burns, J. (1997), results in "asymptotic consistency" with the discrete analysis solution. This is a primary limitation of the local and total CSA methods, because it still requires specific knowledge of the discretization, shape functions, and kinematic assumptions. In particular, for nonlinear problems, it requires specific knowledge of how the nonlinearities are modeled (e.g. what nonlinear terms are included or excluded, how is the system linearized, etc.). When using general purpose codes, this information may not be available, making the implementation of local and total CSA infeasible. Therefore, CSA by nature is also generally an intrusive method. One primary objective of this research is to develop a nonintrusive analytic DSA method. By nonintrusive, it is meant that the method does not need access to the source code, does not make any assumptions about the analysis formulation, and thus, is amenable to general purpose codes and can treat the analysis tool as a black box. Such a method is presented in this research: a nonintrusive local continuum sensitivity method for shape design derivatives using spatial gradient reconstruction.

1.3 Literature on Continuum Sensitivity Analysis

This literature review focuses primarily on CSA, but for additional references, Olhoff, N. and Taylor, J.E. (1983); Haug, E.J. et al. (1986); Haftka, R.T. and Grandhi, R.V. (1986); Haftka, R.T. and Adelman, H.M. (1989) provide broad surveys of all the DSA methods. The development of CSA that has taken place over the last several decades can be separated naturally into three categories of applications: structures, fluids, and fluid-structure interaction. For each application, this section outlines significant research contributions that were made and identifies key points regarding various formulations of CSA.

1.3.1 Structures

CSA was first introduced in the context of structural applications by Haug, E.J. and Arora, J.S. (1978). Shortly afterwards Haug, E.J. and Rousselet, B. (1980a,b, 1982) published a series of papers on DSA that included formulation of CSA for shape design problems, which they called the variational shape design method. More formulations of total CSA were published for structural mechanics problems by Choi, K.K. and Haug, E.J. (1983) and Dems, K. and Mroz, Z. (1983, 1984, 1985). When the design derivatives are formulated as functionals, the domain integrals can be transformed into boundary integrals using an adjoint variable and specific integral identities. This is called the boundary integral method, and it can be computationally less expensive than carrying out traditional total CSA (Choi, K.K. and Haug, E.J., 1983; Chun, Y.W. and Haug, E.J., 1983; Haug, E.J. et al., 1986). However, when using finite element analysis, the results at the boundaries can be inaccurate, which can result in numerical inaccuracies when the boundary integrals are evaluated (Yang, R.J. and Choi, K.K., 1985, 1986; Yang, R.J. and Botkin, M.E., 1986). Braibant, V. and Fleury, C. (1984, 1985) formulated the domain integral method, which is another direct formulation of total CSA. Choi, K.K. and Seong, H.W. (1986) formulated the adjoint version of this. Although the domain integral method avoids the numerical difficulties associated with the boundary integrals, it is computationally more expensive (Liu, S. and Canfield, R.A., 2013b). Choi, K. and Kim, N.H. (2005) documented CSA extensively for structural applications and extended it to nonlinear analysis. They also coined the term "continuum sensitivity analysis" (CSA), which was later adopted by Wickert, D.P. and Canfield, R.A. (2008), Liu, S. and Canfield, R.A. (2011), and this research. Choi, K.K. and Twu, S. (1989) and Akbari, J. et al. (2010) showed that the total CSA and discrete analytic methods produce equivalent design derivative solutions under certain conditions. Choi, K.K. and Twu, S. (1989) showed that the equivalence was achieved by exact analytic integration of element matrices and the exact solution of global discrete finite element equations. Akbari, J. et al. (2010) removed these restrictions and showed that equivalence occurred if the same discretization method, integration technique, shape functions, and linear design velocity fields were used. Liu, S. and Canfield, R.A. (2013a) extended Akbari et al's equivalence proof to general, nonlinear field problems. This begs the question as to what benefits can be gained by using one method over the other.

The design velocity plays a critical role in the difference between local and total CSA. In order to solve the total CSEs, the design velocity must be defined throughout the entire domain. If the deign velocity can not be obtained in a closed form, then it can be computationally expensive to compute (Liu, S. and Canfield, R.A., 2013b). In contrast, solving the local CSEs requires the design velocity to be defined on the boundaries. Therefore, as concluded by Liu, S. and Canfield, R.A. (2013b), local CSA is computationally less expensive than total CSA. To the best knowledge of the author, no implementations of local CSA for structural applications were documented between Dems, K. and Haftka, R. (1989) and Wickert, D.P. and Canfield, R.A. (2008). Over that period, total CSA has traditionally been the more common choice for several reasons: first, shape design optimization requires total design derivatives; second, local CSA requires higher order derivatives on the boundaries, which can be hard to approximate; and lastly, local design derivatives may be discontinuous at structural interfaces, which complicates the local CSA formulation (Choi, K. and Kim, N.H., 2005; Liu, S. et al., 2010; Liu, S. and Canfield, R.A., 2013b). However, if each of these issues can be handled or overcome, then local CSA could possibly provide a more efficient method for calculating shape design derivatives. Wickert, D.P. et al. (2009), Liu, S. et al. (2010), and Cross, D. and Canfield, R.A. (2012a) developed a specific local CSA formulation to handle the aforementioned issues. It turns out that the local CSA method is also well suited towards being implemented with general purpose codes. This is discussed in greater detail in Chapter 2.

1.3.2 Fluids

Jameson (1988) introduced CSA for aerodynamic shape problems with an adjoint formulation. Borggaard, J. and Burns, J. (1994, 1997) developed a direct formulation of CSA that they initially called the "PDE sensitivity" method and later called the "continuous sensitivity equation method". Burns, J.A. and Cliff, E.M. (1998) utilized this method to develop an analytic sensitivity package called SENSE for a CFD code called GASP, an Aerosoft Inc. product. Since those initial developments, CSA for fluids has been well documented. Pelletier and his students have implemented CSA for a variety of fluid sensitivity applications (Turgeon, E. et al., 1999; Hristova, H. et al., 2004; Colin, E. et al., 2005; Duvigneau, R. and Pelletier, D., 2006). These documented applications of CSA for fluids pose the CSEs in terms of local design derivatives, consistent with an Eulerian reference frame. This is advantageous because the local CSA formulation avoids the need to calculate mesh (geometric) sensitivities (design velocity) throughout the domain. The local CSE boundary conditions only require that geometric sensitivities be defined on the boundaries. However, the disadvantage is that local CSE boundary conditions require higher order derivatives to be defined on the boundary. These terms can be difficult to approximate, but Duvigneau, R. and Pelletier, D. (2006) developed a technique called the Taylor series l-patch method to help provide accurate approximations of the local CSE boundary conditions. This technique plays a critical role in the research presented here. Recently, Charlot, L. et al. (2012) investigated the use of total CSA for fluid sensitivities, calling it the continuous Lagrangian sensitivity equation method. They concluded that the method was more efficient than the local (Eulerian) CSA method, because of the added computational cost associated with approximating higher-order derivatives that appear in the boundary conditions. However, the total CSA method is intrusive, a fact that is discussed extensively in later chapters.

1.3.3 Fluid-structure Interaction

CSA for coupled structure and aerodynamic analysis is much less mature than it is for the individual disciplines. Bhatia, K.G. and Rudisill, C.S. (1971) derived flutter derivatives with respect to structural parameters, and analytic sensitivity equations were first derived for aeroelasticity by Barthelemy, J.F. and Bergen, F.D. (1989). DSA for FSI problems was conducted using the discrete analytic method by Ghattas, O. and Li, X. (1998), Lund, E. et al. (2001, 2003), Fernandez, M. and Mouchabir, M. (2001), and many others. Bhaskaran, R. and Berkooz, G. (1997) were the first to present CSA for an aeroelastic problem, but it only included one way, structure to fluid coupling. Etienne, S. and Pelletier, D. (2005) were the first to present CSA for fully coupled FSI problems. Etienne, S. et al. (2007) have since implemented CSA for a variety of FSI problems, primarily solving for flow design derivatives to facilitate shape optimization of aerodynamic flows. Wickert, D.P. et al. (2008, 2009) were the first to formulate and implement CSA for transient gust analysis. This was also the first use of CSA for FSI applications to facilitate structural shape optimization. More recently, Liu, S. and Canfield, R.A. (2011, 2013b) and Cross, D. and Canfield, R.A. (2012b) applied local and total CSA of nonlinear transient gust responses of built-up beam structures.

1.4 Research Contributions

A primary objective of this research is to develop a DSA method that can accurately and efficiently calculate design derivatives of high fidelity models, such as nonlinear transient aeroelastic gust analysis. Furthermore, the DSA method should be nonintrusive, providing the ability to calculate design derivatives of analyses conducted using general purpose codes. A "one size fits all" DSA approach is very appealing, because designers can compute design derivatives for a wide variety of analysis types and fidelity using a single DSA algorithm.

Chapter 2 describes how each of the existing DSA methods fails to meet these criteria. This research presents a specific formulation of the local CSA method that satisfies all of the criteria. The method is called local CSA with spatial gradient reconstruction (SGR). This formulation represents a significant contribution to the current body of literature regarding DSA. Furthermore, an added benefit of using local CSA is that mesh sensitivities are not required. The specific contributions provided by the research presented here are listed next.

- Local CSA with SGR can be implemented with general purpose codes in a nonintrusive fashion. It can treat the analysis tool as a"black box." The only requirements are for the user to differentiate the boundary conditions to define sensitivity loads and for the analysis tool to provide as output the system response of primary and secondary variables, as well as the system matrices. The method is element agnostic and can be readily implemented for a variety of analysis models with varying degrees of fidelity. Presented in this research are numerous examples, conducted with Nastran analyses, that demonstrate these attributes. The results also demonstrate that the method can be as accurate as other analytic DSA methods and more accurate than most.
- This research marks the first application of local CSA for linear and nonlinear analysis of structural models using plate elements. In addition, it is the first demonstration of implementing local CSA with black box analysis tools without knowledge of finite element shape functions. Furthermore, it is the first application of local CSA to a built-up structure that uses mixed element types (plate and beam elements). Interface boundary conditions were derived for joined mixed elements.
- SGR has been utilized in the context of flow sensitivities, but this is the first use of SGR in the context of structural sensitivities. Previously, SGR was utilized to approximate

high-order derivatives of primary variables to formulate local CSE boundary conditions. The local CSA method with SGR presented here requires only the approximation of first-order derivatives, but of both primary and secondary variables. This is a key ingredient in making the method element-agnostic and amenable to general purpose codes. This research marks the first time that SGR has been employed in this way and for this purpose.

- Guidelines for how to control the convergence of accuracy for local CSA with spatial gradient reconstruction is presented. In general, the convergence rate is largely dependent on the accuracy of the spatial derivative approximations that appear in the local CSE boundary conditions. The accuracy of these approximations is driven by the parameters used for the SGR calculations.

In addition to formulating and discussing DSA methods, Chapter 2 also illustrates the implementation of each DSA method for a simple example problem. This chapter and the results presented in it demonstrate the limitations of existing DSA methods. It also paints a clear picture of how the local CSA method with SGR provides significant benefits and flexibility that other DSA methods do not. Chapter 3 presents several implementations of the local CSA method with SGR for linear systems. The examples presented include linear static beam bending, linear transient gust response of a 2-D beam structure, linear static bending of rectangular plates, and linear static bending of a beam-stiffened rectangular plate. Each of these examples is designed to illustrate the specific advantages and attributes of the local CSA method with SGR. Furthermore, the examples were chosen as a gradual path towards using the method to calculate design derivatives of nonlinear transient gust responses of high fidelity structures. Chapter 4 extends the method to nonlinear systems, including built-up beam and plate models. Chapter 5 investigates several numerical considrations regarding grid convergence and the accuracy of SGR. Furthermore, the method is implemented for 2D potential flow, and an adjoint formulation of the method is also presented. Lastly, Chapter 6 provides a summary and discussion of the research and draws concrete conclusions regarding CSA for aeroelastic shape optimization. Chapter 6 also provides recommendations for future work and research.

Chapter 2

Design Sensitivity Analysis: Methodology and Implementation

This chapter provides details regarding the formulation of the finite difference, complex step, discrete analytic, total continuum, and local continuum design sensitivity methods. Each of the methods is implemented and used to calculate shape design derivatives for a simple benchmark problem. The limitations of the existing DSA methods are clearly illustrated by investigating the formulations and the results of the benchmark problem. Most of the limitations have been documented in the literature, but an overall comparison of the methods has never been demonstrated on a single benchmark problem. With the limitations clearly laid out, the stage is set for the local CSA method with spatial gradient reconstruction (SGR) to be introduced. A formulation is provided and the method is implemented for the benchmark problem. Comparing the results to those of the other DSA methods, provides quantitative evidence that local CSA with SGR is an accurate and effective DSA method.

2.1 Mathematical Foundation for Design Sensitivity Analysis

Let Figure 2.1 represent a continuous domain, Ω, in Cartesian space, with essential (geometric) boundary conditions specified on Γ_e, and non-essential (natural) boundary conditions specified on Γ_n, of a boundary value problem

$$\boldsymbol{A}(\boldsymbol{u}, t; \boldsymbol{b}) = \boldsymbol{f}(\boldsymbol{x}, t; \boldsymbol{b}) \qquad \text{on} \qquad \Omega \tag{2.1}$$

$$\boldsymbol{B}(\boldsymbol{u}, t; \boldsymbol{b}) = \boldsymbol{g}(\boldsymbol{x}, t; \boldsymbol{b}) \qquad \text{on} \qquad \Gamma \tag{2.2}$$

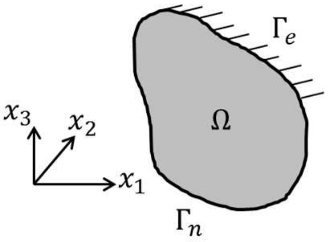


Figure 2.1: Domain, Ω, with boundaries Γ_e and Γ_n

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are the time-space differential operator and the boundary condition operator, respectively. In addition, $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t; \boldsymbol{b})$ is a vector of response variables, $\boldsymbol{b}$ is a vector of design variables, $\boldsymbol{x}$ is a vector of spatial coordinates, and t is a temporal variable. The total design derivative of response variables, $\boldsymbol{u}$ with respect to the i-th design parameter, b_i is equivalent to the material derivative.

$$\frac{D\boldsymbol{u}}{Db_i} = \frac{\partial \boldsymbol{u}}{\partial b_i} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}} \cdot \frac{\partial \boldsymbol{x}}{\partial b_i} \tag{2.3}$$

This material derivative consists of the local design derivative, $\frac{\partial \boldsymbol{u}}{\partial b}$, plus a convective term,

$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}} \cdot \frac{\partial \boldsymbol{x}}{\partial b_i}$. The local design derivative is a measure of the rate of change of the response at a fixed point in the global reference frame due to the rate of change of the design parameter. The convective term accounts for the movement of a material point should the geometric domain, $\boldsymbol{x}$, change with the design parameter (Wickert, D.P. and Canfield, R.A., 2008). The geometric domain is a function of shape design parameters, in which case, the nonzero convective term becomes the difference between the local and total design derivatives. This term contains two gradients: the first, a spatial gradient of the response, the second, a design derivative of the geometric domain. The latter is often referred to as the design velocity (Arora, J. and Haug, E., 1979; Choi, K. and Kim, N.H., 2005). Subsequent derivations use the compact notation

$$\dot{\boldsymbol{u}} = \boldsymbol{u}' + \nabla_{\boldsymbol{x}} \boldsymbol{u} \cdot \boldsymbol{\mathcal{V}} \tag{2.4}$$

The system response, $\boldsymbol{u}$, is obtained by solving the boundary value problem, using either analytic or approximate methods. The examples presented throughout this research use a displacement-based finite element approximation to solve the boundary value problem. After obtaining the system response, the design derivative of the response with respect to design variables can be calculated using any number of DSA methods. For shape design problems, the DSA method can be used to solve for either local or total design derivatives. Once a local or total design derivative solution is known, the other can be obtained using the material derivative definition in Eq. (2.3). For sizing design variables the domain does not change. Therefore, the convective term is zero and the local and total design derivative solutions are equivalent. Sections 2.3 through 2.9 provide details on the formulation and implementation of the DSA methods listed in Figure 1.3, but first, Section 2.2 defines the benchmark problem.

2.2 Benchmark Problem

The benchmark problem was selected from a finite element text by Szabo, B. and Babuska, I. (1991). Figure 2.2 depicts an axial bar of length, $L = 1$, with axial stiffness, $EA = 1$, a distributed load, $T(x) = \sin(\pi x/L)$, an end load, $P = 1/\pi$ at $x = 0$, and a spring of stiffness, $k = 10$ at $x = L$.

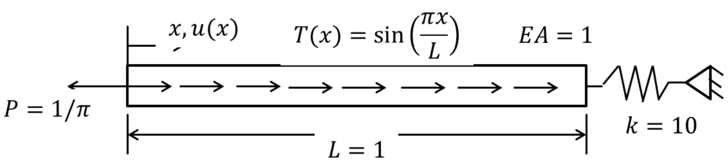


Figure 2.2: Axial bar under a prescribed loading condition (Szabo, B. and Babuska, I., 1991)

The governing equation of the axial bar displacement, leaving L as a design variable, is

$$u_{,xx} + \sin\left(\frac{\pi}{L}x\right) = 0 \tag{2.5}$$

with the following boundary conditions.

$$\begin{aligned} u_{,x}(0) &= \frac{1}{\pi} \\ u_{,x}(L) &= -10u(L) \end{aligned} \tag{2.6}$$

The analytic solution is

$$u(x; L) = \frac{1}{EAk\pi^2}\left(\pi\left(EA\left(2L - P\pi\right) + k\left(L - P\pi\right)\left(L - x\right)\right) + kL^2 \sin\frac{\pi x}{L}\right) \tag{2.7}$$

When $L = 1$, $EA = 1$, $k = 10$, and $P = 1/\pi$, the analytic solution simplifies to

$$u(x; 1) = \frac{1}{\pi^2}\sin(\pi x) + \frac{1}{10\pi} \tag{2.8}$$

For this case only one design variable is being considered. Therefore, $b_1 = L$ and the design velocity can be defined as $\boldsymbol{\mathcal{V}} = x/L$. Substituting this information, along with Eq. (2.7), into the material derivative in Eq. (2.3), yields the analytically derived total design derivative of the response with respect to the length of the beam, L.

$$\dot{u}(x;L) = \frac{\pi\left(2EAL + k\left(2L - P\pi\right)\left(L - x\right)\right) + 2kL^2 \sin\frac{\pi x}{L}}{EAkL\pi^2} \tag{2.9}$$

When $L = 1$, $EA = 1$, $k = 10$, and $P = 1/pi$, the analytic total design derivative is

$$\dot{u}(x;1) = \frac{6\pi - 5\pi x + 10\sin\left(\pi x\right)}{5\pi^2} \tag{2.10}$$

Typically, the boundary value problem cannot be solved analytically, and therefore deriving design derivatives from analytic solutions is not possible. Therefore, DSA methods approximate the design derivatives using approximate solutions to the original governing equations. A finite element solution to Eq. (2.5) is used in the subsequent sections when implementing the various DSA methods. The finite element analysis was conducted using linear shape functions. The mesh convergence of the finite element analysis is depicted in Figure 2.3.

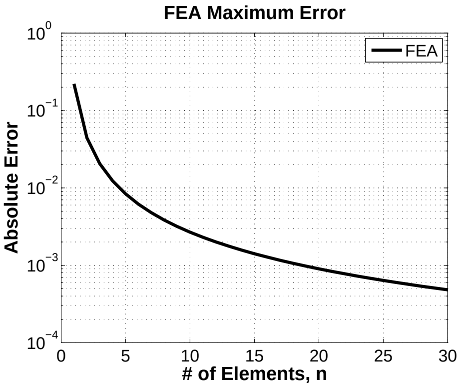


Figure 2.3: Mesh convergence of the finite element results for the axial bar

2.3 Finite Difference Method

2.3.1 Formulation

The finite difference method is classified as a numerical DSA method. The total design derivative in Eq. (2.11) is numerically approximated by a first-order forward finite difference of the analysis response, $\boldsymbol{u}(b_i)$.

$$\frac{D\boldsymbol{u}}{Db_i} \doteq \frac{\boldsymbol{u}\left(b_i + h\right) - \boldsymbol{u}\left(b_i\right)}{h} \tag{2.11}$$

where $h = \delta b_i$, and δ is a small perturbation (step size). If δ is too small, then the approximation becomes poor due to round-off error, and if it is too large, then the approximation becomes poor due to truncation error. One of the main challenges associated with using the finite difference method is determining an appropriate step size. Balancing the round-off and truncation errors can be difficult. Higher order finite difference schemes can be used to improve the accuracy, but such schemes require multiple analyses per design variable. In addition to potential inaccuracies, the finite difference method can be computationally expensive, because it requires additional analyses for each design variable. For an expensive analysis, such as nonlinear transient aeroelastic gust analysis, computing design derivatives with the finite difference method can become very expensive.

2.3.2 Implementation for Axial Bar

The design derivative of the axial bar displacement was calculated using finite differences. The finite element analysis was executed for a bar length, $L = 1$. Additional executions were conducted with perturbations of $\delta = 10^{-1}$, $\delta = 10^{-10}$, and $\delta = 10^{-14}$. A finite difference approximation was calculated for each perturbation. The error results are plotted in Figure 2.4.

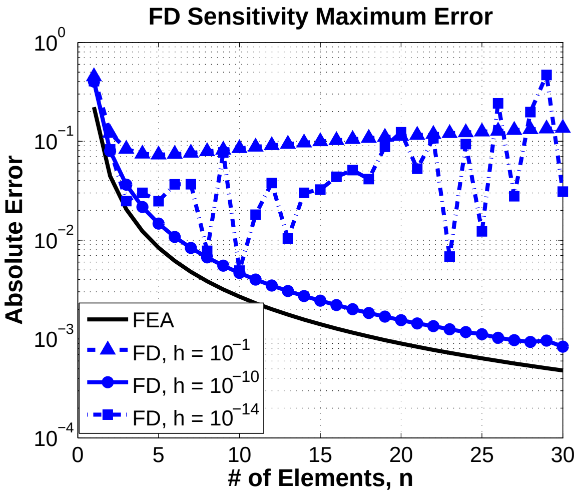


Figure 2.4: Mesh convergence of finite difference design derivatives for the axial bar

The results are consistent with previous statements. For the large perturbation, the finite difference approximation is quite poor due to truncation errors. For the smallest perturbation, the approximation is also poor due to round-off error. The intermediate perturbation of 10^{-10} results in a very good approximation. For this very simple benchmark problem, the displacement at a particular location is linearly proportional to the length of the bar. Therefore, an appropriate step size can be chosen from a relatively wide range of values (10^{-4} to 10^{-12}). However, this is not always the case and is often times unpredictable. Furthermore, if single precision inputs and outputs are being used, then a relatively large step size has to be used, which introduces the truncation errors.

2.4 Complex Step Method

2.4.1 Formulation

The complex step method is also classified as a numerical DSA method. From Squire, W. and Trapp, G. (1998), the total design derivative is numerically approximated as

$$\frac{D\boldsymbol{u}}{Db_i} \doteq \frac{\operatorname{imag}\left(\boldsymbol{u}\left(b_i + hj\right)\right)}{h} \tag{2.12}$$

where j is the imaginary number $\sqrt{-1}$. The complex step method is similar to the finite difference method in that large steps result in truncation error, but differs in that small steps do not result in round-off error, because the approximation does not include a subtractive operation. Therefore, the step size can be chosen arbitrarily small. The difficulty with the complex step method is that the analysis must be able to handle complex input parameters, which is not always feasible (e.g. Nastran). Furthermore, it requires an additional analysis for each design variable beyond the first making it computationally expensive. This is especially true since carrying complex numbers through the analysis requires additional operations.

2.4.2 Implementation for Axial Bar

The complex step method was used to approximate the shape design derivative of the axial bar displacements. A very small perturbation, $\delta = 10^{-100}$, was used. Figure 2.5 shows that both the finite difference and complex step methods provide accurate approximations of the design derivatives.

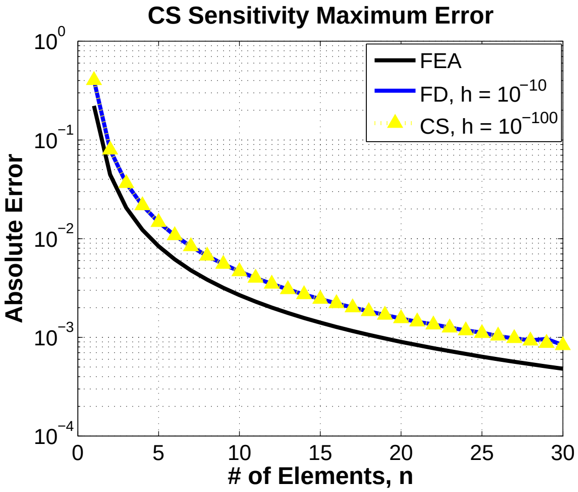


Figure 2.5: Mesh convergence of complex step design derivatives for the axial bar

2.5 Discrete Analytic Method

2.5.1 Formulation

Formulation of the discrete analytic method requires the governing equations, first, to be discretized and, second, to be differentiated. The discretized, linear static version of Eqs. (2.1) and (2.2) is

$$[K]\{u\} = \{F\} \tag{2.13}$$

Differentiating Eq. (2.13) with respect to a shape parameter yields the discrete analytic sensitivity equations

$$[K]\{\dot{u}\} = \{\dot{F}\} - [\dot{K}]\{u\} \tag{2.14}$$

Solving Eq. (2.14) yields a solution, $\{\dot{u}\}$ that contains discrete values of the total design derivatives of the field variables at finite element nodes. The coefficient matrix on the left sides of Eqs. (2.13) and (2.14) are the same. This is only true when the governing system is linear with respect to the response variables. This indicates that the sensitivity system is governed by the same structural model, and can be solved using the same analysis tool by applying new boundary conditions. The boundary conditions, shown on the right side of Eq. (2.14), are referred to as fictitious loads or pseudo loads. (Choi, K. and Kim, N.H., 2005)

The pseudo load, in this case, includes the design derivative of the linear stiffness matrix. Analytic formulation of this term is dependent on the analysis formulation, which makes it an intrusive method. Therefore, implementing the discrete analytic method requires rigorous formulation and "intimate knowledge" of the source code. As pointed out by Haftka, R.T. and Adelman, H.M. (1989), this makes it difficult and often infeasible to implement with general purpose codes. Although highly accurate, the discrete analytic method possesses this unattractive attribute.

2.5.2 Implementation for Axial Bar

A finite element analysis discretized Eqs. (2.5) and (2.6) (the governing equations of the axial bar) with a uniform mesh, linear shape functions, and one point Gauss integration. The linear stiffness matrix, $[K]$ and applied load vector, $\{F\}$ in Eq. (2.13) were assembled using the elemental stiffness matrix and elemental load vector defined in Eqs. (2.15) and (3.19).

$$K_e = \frac{n}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \; e = 1, 2, \ldots, n \tag{2.15}$$

$$F_e = \frac{L}{2n} \sin\left(\frac{(2e-1)\,\pi}{2n}\right) \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}, \; e = 1, 2, \ldots, n \tag{2.16}$$

Here, e represents the e^{th} element of a total of n elements. The design variable, in this case, L, was carried through symbolically, so that the discrete equations could be differentiated analytically. For a simple problem, this is not too difficult, but it is easy to see how this can become very tedious. The design derivatives of the stiffness matrix and load vector were derived by differentiating Eqs. (2.15) and (3.19) with respect to the design variable, L.

$$\dot{K}_e = \frac{DK_e}{DL} = -\frac{n}{L^2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \; e = 1, 2, \ldots, n \tag{2.17}$$

$$\dot{F}_e = \frac{DF_e}{DL} = \frac{1}{2n} \sin\left(\frac{(2e-1)\,\pi}{2n}\right) \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}, \; e = 1, 2, \ldots, n \tag{2.18}$$

Eqs. (2.17) and (2.18) were used to assemble the global stiffness matrix sensitivity, $[\dot{K}]$ and global load vector sensitivity, $\{\dot{F}\}$. These were used to formulate the pseudo loads in Eq. (2.14), which were then solved to obtain the total design derivative solution, $\{\dot{u}\}$. Figure 2.6 shows the discrete analytic results for the axial bar problem.

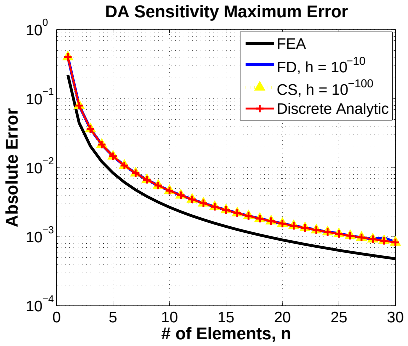


Figure 2.6: Mesh convergence of discrete analytic design derivatives for the axial bar

2.6 Semi-Analytic Method

2.6.1 Formulation

The semi-analytic method approximates Eqs. (2.17) and (2.18) using finite differences. This enables a general formulation of the pseudo loads that is amenable to general purpose codes, but Barthelemy, B. and Haftka, R.T. (1988) have demonstrated that this approach can be highly inaccurate for shape design derivative calculations. They did document several different approaches to improving the method, namely the use of error correction terms. Haftka, R. and Gurdal, Z. (1992) also discuss the inaccuracies associated with the semi-analytic method. Section 2.6.2 illustrates these inaccuracies.

2.6.2 Implementation for Axial Bar

The finite difference terms in the pseudo loads of Eq. (2.14) are approximated using relative step sizes of 10^{-1}, 10^{-10}, and 10^{-14}. Figure 2.7 shows the results, which, as one would expect, have similar trends to the finite difference results. Truncation error dominates for large steps, round-off error dominates for small steps, and in between is a range of step sizes that produce reliable results. Similar to the finite difference method, the range of appropriate step sizes is unpredictable, and even the optimal step size can result in relatively large errors. Some general purpose codes can only take single precision input. For example, when using Nastran to implement the semi-analytic method, single precision input only allows a relative step size as small as 10^{-4}. Since shape problems are dominated by truncation errors (Barthelemy, B. and Haftka, R.T., 1988), the semi-analytic shape design derivatives are largely unreliable. Although, in some instances, the complex step method can be used to approximate the design derivative terms in the pseudo loads.

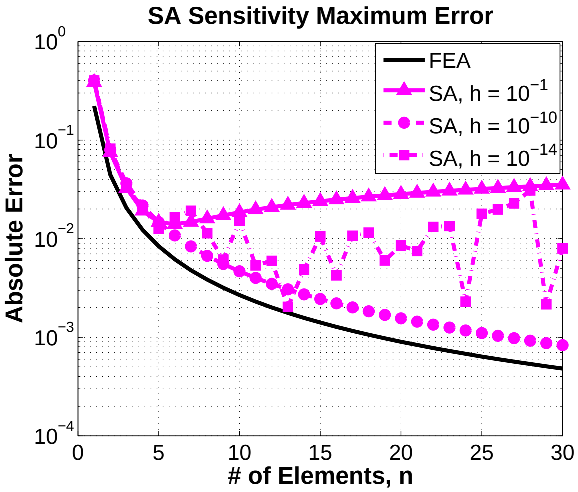


Figure 2.7: Mesh convergence of semi-analytic design derivatives for the axial bar

2.7 Total Continuum Method

For a linear system Eqs. (2.1) and (2.2) can be further defined as

$$\boldsymbol{A}(\boldsymbol{u},t;\boldsymbol{b}) = \boldsymbol{A}_L\boldsymbol{u} = \boldsymbol{f}(\boldsymbol{x},t;\boldsymbol{b}) \qquad \text{on} \qquad \Omega \tag{2.19}$$

$$\boldsymbol{B}(\boldsymbol{u},t;\boldsymbol{b}) = \boldsymbol{B}_L\boldsymbol{u} = \boldsymbol{g}(\boldsymbol{x},t;\boldsymbol{b}) \qquad \text{on} \qquad \Gamma \tag{2.20}$$

where $\boldsymbol{A}_L$ and $\boldsymbol{B}_L$ are constant linear differential operators. Total differentiation of Eqs. (2.19) and (2.20) with respect to a design variable yields the total CSEs.

$$\boldsymbol{A}_L\dot{\boldsymbol{u}} = \dot{\boldsymbol{f}}(\boldsymbol{x},t;\boldsymbol{b}) - \dot{\boldsymbol{A}}_L\boldsymbol{u} \qquad \text{on} \qquad \Omega \tag{2.21}$$

$$\boldsymbol{B}_L\dot{\boldsymbol{u}} = \dot{\boldsymbol{g}}(\boldsymbol{x},t;\boldsymbol{b}) - \dot{\boldsymbol{B}}_L\boldsymbol{u} \qquad \text{on} \qquad \Gamma \tag{2.22}$$

Akbari, J. et al. (2010) and Liu, S. and Canfield, R.A. (2013a) have shown that under certain conditions, once the total CSEs are discretized, they are equivalent to the discrete analytic sensitivity equations in Eq. (2.14). For all intents and purposes, the discrete analytic and total continuum methods are equivalent. For more details on the specific formulation for the total continuum method see Liu, S. and Canfield, R.A. (2013b).

The discrete analytic and total CSA methods are the most common analytic sensitivity methods for structural applications, because structural optimization requires total design derivatives and no special attention needs to be given to structural interfaces (Choi, K. and Kim, N.H., 2005). However, formulation of the sensitivity boundary conditions (pseudo loads) requires specific knowledge of the analysis formulation and numerical techniques used to conduct the original analysis. Therefore, total CSA is also an intrusive method. Although, local CSA requires extra boundary conditions to be imposed at structural interfaces (Liu, S. et al., 2010) and it requires higher order derivatives on the boundaries (Liu, S. and Canfield, R.A., 2013b), it does provide some benefits, which are discussed in the next section.

2.8 Local Continuum Method

2.8.1 Formulation

For a linear system, the local CSEs are derived by partial differentiation of Eq. (2.19) and total differentiation of Eq. (2.20). Total differentiation must be used on the boundary in order to capture the shape change. The differentiation of the domain equation yields

$$\boldsymbol{A}_L\boldsymbol{u}' = \boldsymbol{f}'(\boldsymbol{x}, t; \boldsymbol{b}) - \boldsymbol{A}_L'\boldsymbol{u} \qquad \text{on} \qquad \Omega \tag{2.23}$$

and for the boundary equation

$$\boldsymbol{B}_L\dot{\boldsymbol{u}} = \dot{\boldsymbol{g}}(\boldsymbol{x}, t; \boldsymbol{b}) - \dot{\boldsymbol{B}}_L\boldsymbol{u} \qquad \text{on} \qquad \Gamma \tag{2.24}$$

The total derivatives of $\dot{\boldsymbol{u}}$ and $\dot{\boldsymbol{B}}_L$ can be expanded, and Eq. (2.24) can be rearranged to define the local design derivative boundary condition.

$$\boldsymbol{B}_L\boldsymbol{u}' = \dot{\boldsymbol{g}}(\boldsymbol{x}, t; \boldsymbol{b}) - \boldsymbol{B}'_L\boldsymbol{u} - \nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u}) \cdot \boldsymbol{\mathcal{V}} \qquad \text{on} \qquad \Gamma \tag{2.25}$$

Eqs. (2.23) and (2.25) are the local CSEs. Discretizing the static version (neglecting time) of the linear local CSEs yields

$$[K]\{u'\} = \{F_{Local}\} \tag{2.26}$$

Because the coefficient matrix is equal to $[K]$, the local CSE system can be solved using the same analysis tool used to solve the original system by applying new boundary conditions. This is a property possessed by each of the analytic sensitivity methods. The local CSE pseudo load, $\{F_{Local}\}$ is the result of discretizing the right-hand-sides of Eqs. (2.23) and (2.25). From Eqs. (2.23) and (2.25), $\boldsymbol{f}'$ and $\dot{\boldsymbol{g}}$, can be derived analytically from the definition of the applied loads on the original system. Appendix A.1 elaborates on how to handle concentrated loads. Typically, for structural elasticity problems, $\boldsymbol{A}'_L$ and $\boldsymbol{B}'_L$ are zero. This is not gauranteed to be true, but Appendix A.2 elaborates on this issue. The design velocity, $\boldsymbol{\mathcal{V}}$ can be derived analytically from the geometry definition. This leaves, $\nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u})$, yet to be addressed.

Using the definition of the boundary operator, the $\nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u})$ term can be expanded as a function of spatial derivatives of the primary variables. The analysis shape functions and the analysis response can be used to approximate these derivatives. Once the derivatives are approximated, the entirety of the local CSE boundary conditions (pseudo load) can be formulated, and the local design derivative solution can be obtained. The local design derivative solution must then be transformed into a total design derivative solution via the convective term. One of the primary advantages of local CSA is that the design velocity is only required on the boundaries, and therefore mesh sensitivities, which are an expensive component of the discrete analytic method, are avoided. In addition, as detailed in Section 2.9, the local CSA method can be manipulated to yield a general formulation of the local CSEs that is nonintrusive and can be used with general purpose codes.

2.8.2 Implementation for Axial Bar

Partial differentiation of Eq. (2.5) (governing equation of axial bar) yields the local CSE

$$u'{,_{xx}} - \frac{\pi x}{L^2} \cos\left(\frac{\pi}{L} x\right) = 0 \tag{2.27}$$

Following the formulation in Eq. (2.25), and knowing that $\mathcal{V} = x/L$, the local CSE boundary conditions are

$$\begin{aligned} u'{,_x}(0) &= 0 - u{,_{xx}}(0)\mathcal{V}(0) = 0 \\ u'{,_x}(L) &= -10\dot{u}(L) - u{,_{xx}}(L)\mathcal{V}(L) = -10u'(L) - 10u{,_x}(L) - u{,_{xx}}(L) \end{aligned} \tag{2.28}$$

Figure 2.8 shows two results for the local CSA method. For the solid line, the spatial derivatives in Eq. (2.28) are formulated from the analytic solution. The spatial derivative in the convective term was also formulated using an analytic solution, when transforming the local design derivative solution into a total design derivative solution. For the dashed line, the spatial derivatives were approximated using the finite element solution and the linear shape functions. The latter result is the one traditionally used, simply because analytic solutions are not available and the derivatives must be approximated. The linear shape functions do a poor job of approximating the derivatives, which leads to relatively large errors (Choi, K. and Kim, N.H., 2005).

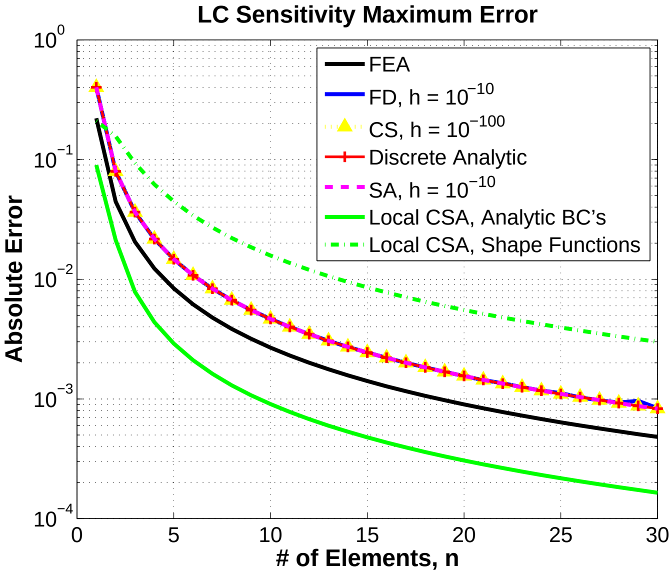


Figure 2.8: Mesh convergence of local continuum design derivatives for the axial bar

This result clearly demonstrates one of the primary reasons that local CSA is not commonly implemented for structural applications. However, the result which unfairly uses analytic derivatives indicates that, if a better approximation of the derivatives can be obtained, then the local CSA result can be dramatically improved. In order to formulate the pseudo loads completely, this method still requires the definition of the boundary operator and the shape functions to be known. The next section introduces a numerical technique that can be used to better approximate the spatial derivatives that appear in the local CSE boundary conditions.

2.9 Local Continuum Method with Spatial Gradient Reconstruction (SGR)

2.9.1 Formulation

As previously mentioned, the local CSE system in Eqs. (2.23) and (2.25) can be solved by applying the sensitivity boundary conditions to the structural model. Assuming that $\boldsymbol{B}'_L$ equals zero, all terms in the sensitivity boundary conditions, except for $\nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u})$, can be formulated without knowledge of the analysis formulation. If this term can be sufficiently approximated without regard to the analysis formulation, then a nonintrusive, general formulation of the local CSE system is feasible.

An essential (geometric) boundary condition is a special case of Eq. (2.19).

$$(\boldsymbol{B}_L\boldsymbol{u})\,|_{\Gamma_e} = \boldsymbol{u}|_{\Gamma_e} = \boldsymbol{g}(\boldsymbol{x},t;\boldsymbol{b})|_{\Gamma_e} \tag{2.29}$$

Taking the material derivative of Eq. (2.29) and rearranging terms according to Eq. (2.25), the essential local sensitivity boundary condition becomes

$$(\boldsymbol{B}_L\boldsymbol{u}')\,|_{\Gamma_e} = \boldsymbol{u}'|_{\Gamma_e} = (\dot{\boldsymbol{g}}(\boldsymbol{x},t;\boldsymbol{b}) - \nabla_{\boldsymbol{x}}\boldsymbol{u}\cdot\boldsymbol{\mathcal{V}})\,|_{\Gamma_e} \tag{2.30}$$

A similar formulation is used for a non-essential (natural) boundary condition.

$$(\boldsymbol{B}_L\boldsymbol{u})\,|_{\Gamma_n} = \boldsymbol{Q}|_{\Gamma_n} = \boldsymbol{g}(\boldsymbol{x},t;\boldsymbol{b})|_{\Gamma_n} \tag{2.31}$$

$$(\boldsymbol{B}_L\boldsymbol{u}')\,|_{\Gamma_n} = \boldsymbol{Q}'|_{\Gamma_e} = (\dot{\boldsymbol{g}}(\boldsymbol{x},t;\boldsymbol{b}) - \nabla_{\boldsymbol{x}}\boldsymbol{Q}\cdot\boldsymbol{\mathcal{V}})\,|_{\Gamma_n} \tag{2.32}$$

For an essential boundary condition $\nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u}) = \nabla_{\boldsymbol{x}}\boldsymbol{u}$, and for a natural boundary condition $\nabla_{\boldsymbol{x}}(\boldsymbol{B}_L\boldsymbol{u}) = \nabla_{\boldsymbol{x}}\boldsymbol{Q}$. For a displacement-based finite element analysis, $\boldsymbol{u}$ is a vector of displacement component functions and $\boldsymbol{Q}$ are internal forces. Displacements, rotations, and internal forces and moments are typical outputs of finite element solvers. Therefore, it is reasonable to expect that the spatial gradients of $\boldsymbol{u}$ and $\boldsymbol{Q}$ on the boundaries, Γ_e and Γ_n, respectively, can be numerically approximated from the analysis output. If this is the case, by simply using output of the primary and secondary variables, the sensitivity boundary conditions in Eq. (2.32) can be formulated regardless of how the analysis tool discretizes $\boldsymbol{B}$.

Upon solving the local CSEs, the local design derivative solution must be transformed into a total design derivative solution via the convective term in Eq. (2.4). This requires approximation of the spatial gradient of $\boldsymbol{u}$ at any point in the domain, Ω, where the total design derivative is desired. The next section details how the aforementioned gradients are approximated without the use of the finite element shape functions.

2.9.2 Spatial Gradient Reconstruction (SGR)

The method used to approximate the gradients stems from the work of Duvigneau, R. and Pelletier, D. (2006). Zienkiewicz, O.C. and Zhu, J.Z. (1992) present a patch technique to recover spatial derivatives of response data. Following this work, Duvigneau and Pelletier presented a method to recover high-order spatial derivatives of primary response variables, which appear in local CSE boundary conditions, like those in Eq. (2.28). The research presented here utilizes their method to recover only first-order derivatives but of both primary ($\nabla\boldsymbol{u}$) and secondary ($\nabla\boldsymbol{Q}$) variables. The advantage this provides is that the differential relationship of the boundary operator to the primary variables does not need to be known.

The user must simply be able to understand which primary and secondary variables are associated with each of the essential and nonessential boundary conditions.

A 2-D formulation of the method begins with a Taylor series expansion of an arbitrary response variable φ about a boundary node P.

$$\begin{aligned}\varphi(x,y) = \varphi(x_P,y_P) + (x-x_P)\varphi_x + (y-y_P)\varphi_y + \frac{1}{2}(x-x_P)^2\varphi_{xx} + ...\\ ... + (x-x_P)(y-y_P)\varphi_{xy} + \frac{1}{2}(y-y_P)^2\varphi_{yy} + ...\end{aligned} \tag{2.33}$$

The response variable φ and its derivatives can be determined through a least squares match of the Taylor series to the analysis response at a set of nodes, $N(x_n, y_n)$, within a patch, $\mathcal{P}$, surrounding node P. That is, solve

$$\min \mathcal{J}(\mathbf{y}_\varphi) = \frac{1}{2}\sum_{N\in\mathcal{P}}(\mathbf{B}^T\mathbf{y}_\varphi - \varphi_N)^2 \tag{2.34}$$

where,

$$\varphi_N = \varphi(x_n, y_n) \tag{2.35}$$

$$\mathbf{B}^T = \{1, x_n - x_P, y_n - y_P, \frac{1}{2}(x_n-x_P)^2, (x_n-x_P)(y_n-y_P), \frac{1}{2}(y_n-y_P)^2, ...\} \tag{2.36}$$

$$\mathbf{y}_\varphi^T = \{\varphi, \varphi_x, \varphi_y, \varphi_{xx}, \varphi_{xy}, \varphi_{yy}, ...\} \tag{2.37}$$

The coordinate dimensions of the Taylor series should be adjusted as necessary to match the dimensions of the problem. Duvigneau and Pelletier define the element patch, $\mathcal{P}$, by layers. A 1-layer patch includes elements that share the node P. A 2-layer patch includes elements adjacent to a 1-layer patch. A 3-layer patch includes elements adjacent to a 2-layer patch, and so forth. Figure 2.9 illustrates various patch definitions. The accuracy of the method depends on the node density of the patch. More specifically, it depends on the mesh size, the number of layers in the patch, and the order of the Taylor series expansion.

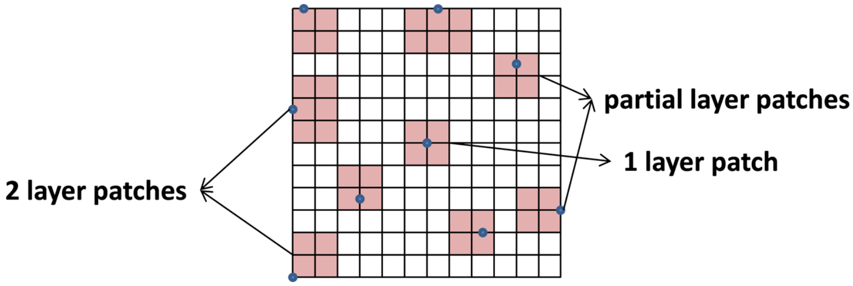


Figure 2.9: Illustration of various patch definitions for SGR

2.9.3 Implementation for Axial Bar

The same local CSE in Eq. (2.27) is used, but formulation of the local CSE boundary conditions changes slightly. Both end conditions of the axial bar are natural boundary conditions. Therefore, let the boundary conditions of the original system be rewritten in terms of the axial force, N.

$$\begin{aligned} N(0) &= 1/\pi \\ N(L) &= -10u(L) \end{aligned} \tag{2.38}$$

Applying Eq. (2.32) yields the local CSE boundary conditions

$$\begin{aligned} N'(0) &= 0 - N_{,x}(0)\mathcal{V}(0) = 0 \\ N'(L) &= -10\dot{u}(L) - N_{,x}(L)\mathcal{V}(L) = -10u'(L) - 10u_{,x}(L) - N_{,x}(L) \end{aligned} \tag{2.39}$$

Eq. (2.39) contains first-order derivatives of the primary and secondary variable, whereas the traditional local CSE formulation in Eq. (2.28), contains first- and second- order derivatives of only the primary variable. Using the displacement and force output of the finite element analysis, SGR can be employed to approximate the spatial derivatives that appear in Eq. (2.39). Figure 2.10 compares the local CSA with SGR results to the results of the other DSA methods.

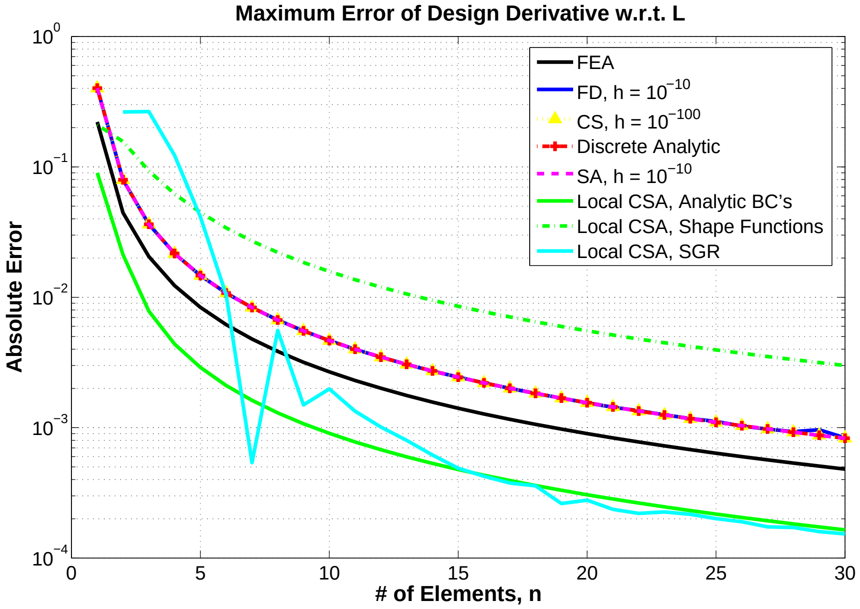


Figure 2.10: Mesh convergence of axial bar design derivatives for all DSA methods

SGR yields more accurate derivative approximations than the shape functions when a sufficient number of elements are present. For n elements, the SGR calculations used n-layer patches and n-order Taylor series expansions for $n \leq 6$ and seven-layer patches and sixth-order Taylor series expansions for $n > 6$. This accounts for the kink in the result at $n = 6$ and $n = 7$. This was to some extent an arbitrary choice made by basic intuition that the sixth-order Taylor series expansions would demonstrate a significant improvement in accuracy. The results indicate that local CSA can outperform all other DSA methods if accurate boundary conditions are applied. With a sufficient number of nodes, SGR is the vehicle by which this accuracy is achieved. Furthermore, it can be done in a nonintrusive manner, which can also be said only for the error prone numerical methods.

One numerical consideration for the local CSA method with SGR is the mesh refinement that is required to achieve accurate approximations of the spatial derivatives on the boundary. The user has some control over the trade-off between accuracy and efficiency by adjusting the node density of the patches and the order of the Taylor series expansions. This topic is investigated further in Chapter 5.

2.10 On Implementation and Computational Effort

Figure 2.11 is a detailed flow chart of the method that outlines the steps and required computations. First, after conducting the original analysis, certain output must be parsed. The method requires global system matrices, displacement and element force data, and the element connectivity. If structural interfaces are present, then the method requires elemental system matrices as well. Typically, this output is obtainable, but in some cases it may not be. During implementation some computational effort must be devoted to parsing this output. Second, the spatial derivatives that appear in the boundary conditions of the local CSE system and that are needed for the local to total design derivative conversion must be approximated by means of the previously defined SGR routine. The computational effort required here depends on the number of derivatives required, the node density of the patches, and the order of the Taylor series expansions. Lastly, a linear solve is required in order to solve the local CSE system.

The finite difference method requires $n + 1$ linear analyses for n design variables. The local CSA method with SGR can utilize a saved matrix decomposition, which means it only requires 2 total linear solves for each of the design variables plus the computations made dur-

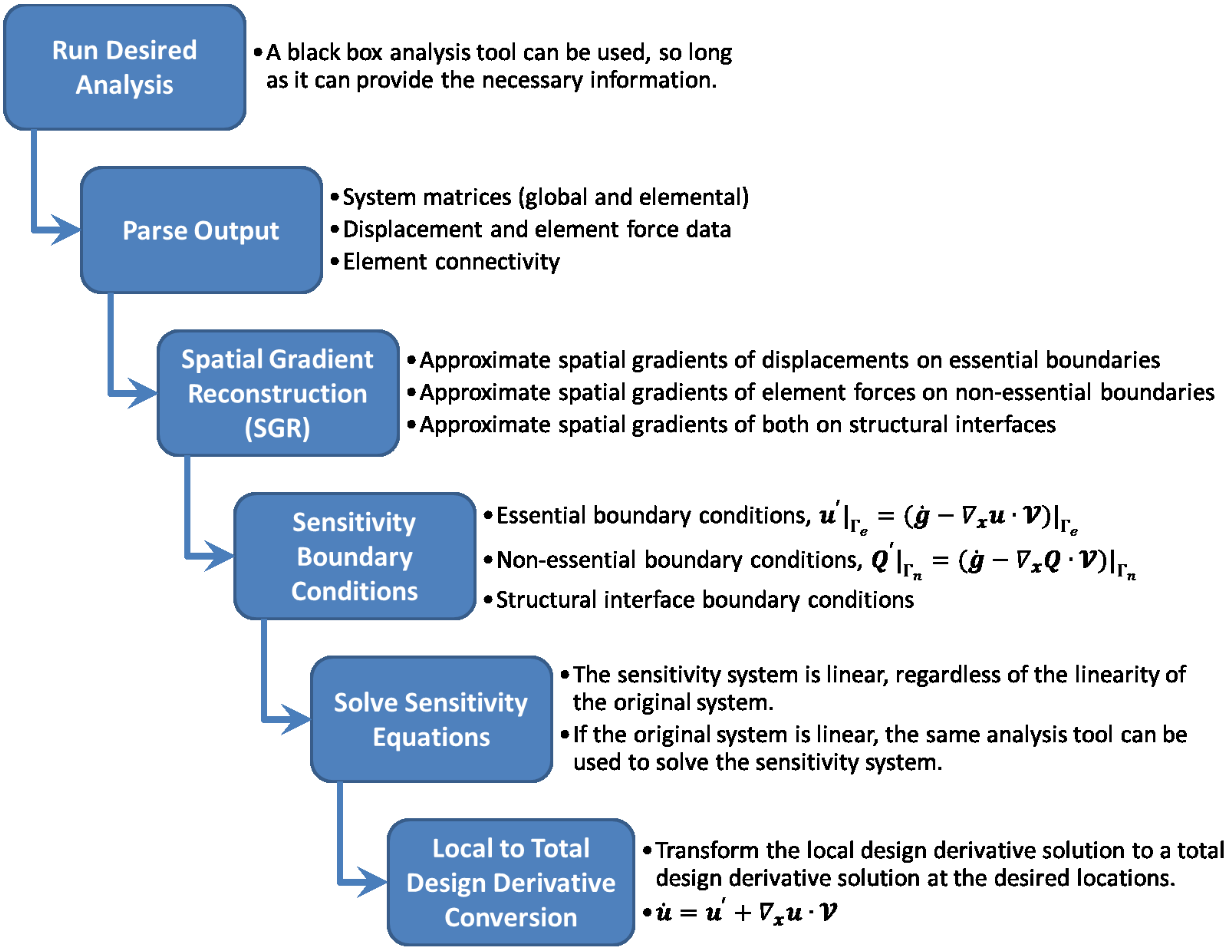


Figure 2.11: Detailed flow chart of local continuum shape sensitivity method with spatial gradient reconstruction

ing SGR. Although approximating the spatial derivatives using SGR seems computationally expensive, the same derivative approximations are required by each of the n design variables. For practical design problems that have many design variables, the local CSA method with SGR will often require less computational effort than the finite difference method. Due to SGR, this method will typically be more expensive than conventional continuum sensitivity methods, but they too require computations to formulate the boundary conditions. These methods can have a major advantage in computational cost if they can be successfully implemented for nonlinear problems.

Chapter 3

Local Continuum Sensitivity Analysis with Spatial Gradient Reconstruction: Linear Applications

In this chapter, local CSA with SGR is implemented for several beam and plate models using linear static and linear transient analysis. The problems presented are specifically designed to illustrate the primary advantages of the method and to demonstrate how certain limitations of the method can be addressed.

3.1 Application to Beam Models

3.1.1 A Comparison of Sensitivity Methods

A cantilevered beam with a non-uniform distributed transverse load is shown in Figure 3.1. This simple model is used to illustrate quantitative comparisons between various sensitivity methods. The model is analysed using a personal code for linear static Euler-Bernoulli beam theory.

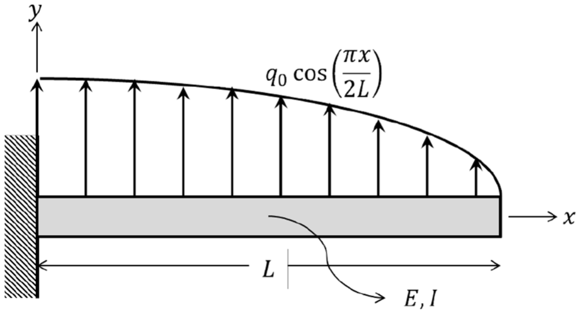


Figure 3.1: Cantilevered beam with a non-uniform distributed transverse load

Finite difference, semi-analytic, discrete analytic (total continuum), and local continuum design derivatives of the transverse displacement, $\dot{w}$, with respect to the length of the beam, L are calculated. The true relative errors of the total design derivative solutions, $\epsilon_{\dot{w}}$, are calculated using the infinity norm of the difference between the approximate and analytic solutions at the finite element nodes, i.e. $\dot{w}_{FE}$ and $\dot{w}_a$, respectively. These errors are plotted in Figure 3.2.

$$\epsilon_{\dot{w}} = \frac{\|\dot{w}_{FE} - \dot{w}_a\|_\infty}{\|\dot{w}_a\|_\infty} \tag{3.1}$$

The error trend for the finite element analysis (solid line with no marker) is an increase with the number of elements because the error is already approaching machine precision for ten elements. The discrete analytic or total CSA (red circles) and local CSA (red crosses) methods are nearly as accurate as the finite element analysis. This high degree of accuracy comes at the cost of difficult implementation that requires knowledge of the source code. The local CSA result comes from using analytically derived boundary condition terms. The hermite-cubic shape functions used in the finite element analysis are not able to recover the fourth-order derivative of displacement that appears in the local CSE boundary conditions. Therefore, conventional implementation of local CSA is infeasible for this problem. The finite

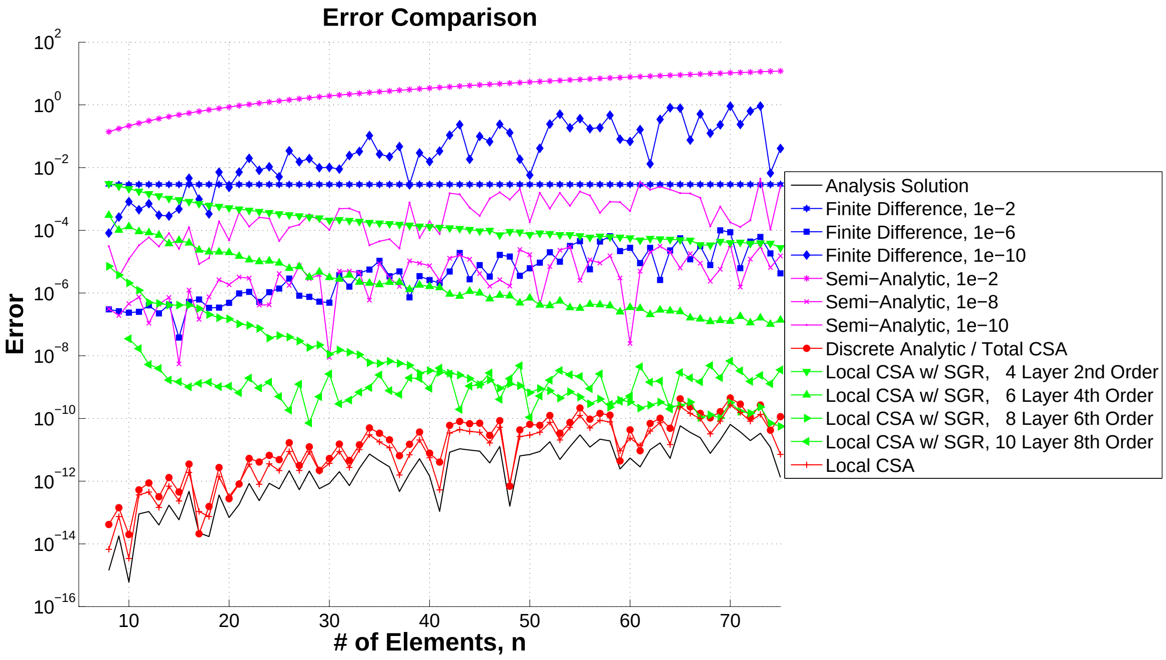


Figure 3.2: Absolute relative errors associated with various sensitivities methods

difference (blue hexagrams, squares, and diamonds) and semi-analytic (magenta asterisks, exes, and dots) methods are much less accurate. Both methods suffer from truncation and roundoff errors, at large and small finite difference step sizes, respectively. Somewhere in between, each method has an optimal step size, which can still result in relatively large errors.

Using SGR to approximate the boundary conditions of the local CSA formulation is an appealing alternative to the other methods. These results (green triangles) illustrate strong convergence of the design derivative calculation. The accuracy of the method is improved by increasing the node density of the patches and the order of the Taylor series expansions used for SGR. Figure 3.2 demonstrates that the method can be nearly as accurate as the pure analytic methods, but, because it does not require knowledge of the element theory, it can be implemented with a black box and a variety of analysis models. Although, some efficiency may be sacrificed in order to converge the SGR solution, the user does have unique control of the tradeoff between accuracy and efficiency.

3.1.2 Linear Transient Gust Response of 2-D Beam Model

Liu, S. et al. (2010) introduced the aeroelastic model shown in Figure 3.3 in the context of CSA. The model consists of a long and slender cantilevered beam with a typical section airfoil mounted at the free end. An angled bracing member supports the cantilevered beam. The aeroelastic structure is submerged in a free stream flow and during a specified period of time encounters a one-minus-cosine vertical gust load. The system is aeroelastic because the beam deformation and the aerodynamic load are coupled. The objective is to compute the design derivatives of the transient gust response with respect to the length of the cantilevered beam, L. The geometry is parameterized such that the angle, β, remains constant for a perturbation in L. The angle, β has been treated as a shape variable by Liu, S. and Canfield, R.A. (2013b).

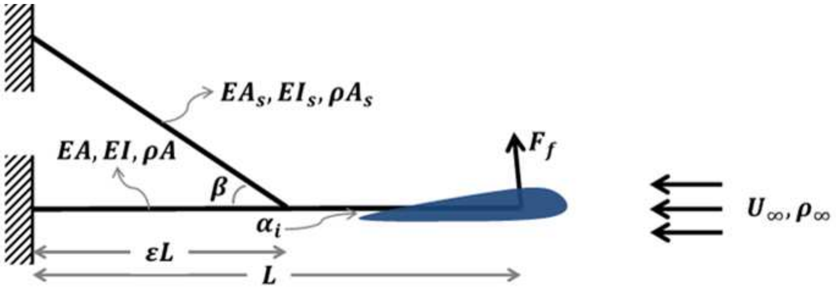


Figure 3.3: Typical section airfoil mounted on the free end of a braced cantilever beam

The Structural Parameters

The cantilevered beam has the following properties: the length of the beam, $L = 6m$, the axial stiffness, $EA = 1.0556 \times 10^8 N$, the bending stiffness, $EI = 3.0023 \times 10^6 Nm^2$, the shear stiffness, $GA = 3.9208 \times 10^7 N$, the shear correction factor, $k = 5/6$, the mass per unit length, $\rho A = 307kg/m$, and the mass moment of inertia, $I_m = 52.4\ kg\ m^2$. The properties of the bracing member (strut) are: the strut angle, $\beta = 45^o$, the strut location, $\epsilon = 1/2$, the axial stiffness, $EA_s = 7 \times 10^6 N$, the bending stiffness, $EI_s = 6.5023 \times 10^4 Nm^2$, the mass per unit length, $\rho A_s = 34N/m$, and the mass moment of inertia, $I_{m_s} = 1.3\ kg\ m^2$.

The Typical Section Aerodynamics

The lift force, $F_f(t)$, is modeled using typical section, quasi-steady aerodynamics.

$$F_f(t) = q_\infty s C_{l_\alpha} \alpha_{eff}(t) \tag{3.2}$$

In Eq. (3.2), q_∞ is the dynamic pressure, $s = 6m^2$ is the effective lifting area (the chord multiplied by the span of the airfoil section), C_{l_α} is the lift curve slope, and α_{eff} is the effective angle of attack. The dynamic pressure is defined as

$$q_\infty = \frac{1}{2}\rho_\infty U_\infty^2 \tag{3.3}$$

where, ρ_∞ is the air density and U_∞ is the free stream velocity. The effective angle of attack is defined as

$$\alpha_{eff} = \alpha_i + \theta_{tip} + \frac{U_g}{U_\infty} + \frac{h_{,t}}{U_\infty} \tag{3.4}$$

In Eq. (3.4), α_i is the incidence angle of attack (orientation of the airfoil relative to the beam frame), θ_{tip} is the rotation of the beam at the free end, U_g is the gust velocity, and $h_{,t}$ is the plunge velocity at the free end of the beam. The plunge velocity has a damping effect, which causes the gust response to dissipate, returning the system to the steady state solution. The gust velocity is defined in Eq. (3.5).

$$U_g(t) = \begin{cases} \frac{1}{2}U_{g,max}\left(1 - \cos\left(\frac{2\pi(t-\tau_i)}{\tau_f}\right)\right) & \tau_i \leq t \leq \tau_f \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

The aerodynamic parameters are: the air density, $\rho_\infty = 1.2kg/m^3$, the free stream velocity, $U_\infty = 70m/s$, the effective lifting area, $s = 6m$, the lift curve slope, $C_{l_\alpha} = 2\pi$, the incidence angle of attack, $\alpha_i = 3^o$, the peak gust velocity, $U_{g,max} = 20m/s$, and the gust period, $\tau_i = 3s$ and $\tau_f = 4s$.

Analysis and Local Continuum Sensitivity Equations

This analysis model is of interest for two reasons in particular. First, the bracing member simulates some of the physical effects seen in joined-wing configurations. The bracing member is subject to buckling during the gust response, which can be a critical design load condition. Second, stress and strain discontinuities can be present at structural interfaces, which creates discontinuities in the local design derivative variables. Solving the local CSE system for a model with structural interfaces requires special treatment (Liu, S. and Canfield, R.A., 2013a). This is one reason local CSA is not commonly used with built-up structures. Analysis of this particular model demonstrates that local CSA with SGR can be implemented with built-up structures.

At the initial time, $t = 0$, the beam is stationary and undeformed, but combination of the free stream flow and the incidence angle of attack produces a positive lift force. Gravity is neglected and there are no other body forces included. The beam is cantilevered at $x = 0$, and the only applied load is the aerodynamic force, $F_f(t)$. The response vector, $\boldsymbol{u}(x, t; \boldsymbol{b})$ contains axial displacement, $u(x, t; \boldsymbol{b})$, transverse displacement, $w(x, t; \boldsymbol{b})$, and rotation, $\psi(x, t; \boldsymbol{b})$, so that $\boldsymbol{u} = \{u, w, \psi\}^T$. The internal force vector, $\boldsymbol{Q}(x, t; \boldsymbol{b})$ contains axial force, $N(x, t; \boldsymbol{b})$, transverse shear force, $V(x, t; \boldsymbol{b})$, and bending moment, $M(x, t; \boldsymbol{b})$, so that $\boldsymbol{Q} = \{N, V, M\}^T$. The general form of the governing equations is shown in Eqs. (1) through (3.8).

$$\boldsymbol{A}_L \boldsymbol{u} = \boldsymbol{f}(x, t; \boldsymbol{b}) = 0 \qquad on \qquad \Omega \tag{3.6}$$

$$\boldsymbol{B}_L \boldsymbol{u}|_{\Gamma_e} = \boldsymbol{u}|_{\Gamma_e} = \boldsymbol{g}(x, t; \boldsymbol{b})|_{\Gamma_e} = \{0, 0, 0\}^T \tag{3.7}$$

$$\boldsymbol{B}_L \boldsymbol{u}|_{\Gamma_n} = \boldsymbol{Q}|_{\Gamma_n} = \boldsymbol{g}(x, t; \boldsymbol{b})|_{\Gamma_n} = \{0, F_f(t), 0\}^T \tag{3.8}$$

From Eqs. (2.23), (2.30), and (2.32), the local CSEs with respect to the length of the beam, L, simplify to

$$\boldsymbol{A}_L \boldsymbol{u}' = \boldsymbol{f}'(x, t; \boldsymbol{b}) - \boldsymbol{A}'_L \boldsymbol{u} = 0 \qquad on \qquad \Omega \tag{3.9}$$

$$\begin{aligned} \boldsymbol{B}_L \boldsymbol{u}'|_{\Gamma_e} = \boldsymbol{u}'|_{\Gamma_e} &= \left(\dot{\boldsymbol{g}}(x, t; \boldsymbol{b}) - \nabla_x \boldsymbol{u} \cdot \boldsymbol{\mathcal{V}}\right)|_{\Gamma_e} \\ &= \{0, 0, 0\}^T - \boldsymbol{u}_{,x} \cdot \{0, 0, 0\}^T \\ &= \{0, 0, 0\}^T \end{aligned} \tag{3.10}$$

$$\begin{aligned} \boldsymbol{B}_L \boldsymbol{u}'|_{\Gamma_n} = \boldsymbol{Q}'|_{\Gamma_n} &= \left(\dot{\boldsymbol{g}}(x, t; \boldsymbol{b}) - \nabla_x \boldsymbol{Q} \cdot \boldsymbol{\mathcal{V}}\right)|_{\Gamma_n} \\ &= \{0, \dot{F}_f(t), 0\}^T - \boldsymbol{Q}_{,x}|_{x=L} \cdot 1 \\ &= \{-N_{,x}, \dot{F}_f(t) - V_{,x}, -M_{,x}\}^T|_{x=L} \end{aligned} \tag{3.11}$$

In Eqs. (3.10) and (3.11), the design velocity, $\boldsymbol{\mathcal{V}}$, is defined as

$$\Omega \equiv \{x(\xi) = \xi L \qquad | \qquad \xi \in [0,1]\} \tag{3.12}$$

$$\boldsymbol{\mathcal{V}}(\xi) = \frac{\partial x(\xi)}{\partial L}\hat{x} = \xi \hat{x} \tag{3.13}$$

The total design derivative of the aerodynamic force, $\dot{F}_f(t)$, with respect to the length of the beam, $\boldsymbol{b} = L$, is defined in Eq. (4.21).

$$\begin{aligned} \dot{F}_f\,(t) &= q_\infty s C_{l_\alpha}\left(\dot{\theta}_{tip} + \frac{1}{U_\infty}\dot{h}_{,t}\right) \\ &= q_\infty s C_{l_\alpha}\left(\psi' + \psi_{,x} + \frac{1}{U_\infty}\left(-w_{,t}' - w_{,tx}\right)\right)\Bigg|_{x=L} \end{aligned} \tag{3.14}$$

The local CSE boundary conditions require $\boldsymbol{Q}_{,x}$, $\psi_{,x}$, and $w_{,tx}$ at $x = L$. Each of these terms is obtained using SGR. The local design derivatives of tip rotation, $\psi'(L,t;\boldsymbol{b})$, and plunge velocity, $-w_{,t}'(L,t;\boldsymbol{b})$, that appear in Eq. (4.21), make the local CSE system aeroelastic. This is noteworthy, because it illustrates that aeroelastic effects carry over from the original analysis to the CSA. Special treatment of the joint requires additional CSE boundary conditions to be formulated (Liu, S. and Canfield, R.A., 2013a).

$$\boldsymbol{u}'(x_j^+) = \boldsymbol{u}'(x_j^-) - \left(\boldsymbol{u}_{,x}(x_j^+) - \boldsymbol{u}_{,x}(x_j^-)\right)\boldsymbol{\mathcal{V}}(x_j) \tag{3.15}$$

$$\boldsymbol{Q}'(x_j) = -\sum \boldsymbol{Q}_{,x}(x_j)\boldsymbol{\mathcal{V}}(x_j) \tag{3.16}$$

where, x_j is the joint location and is equal to ϵL. A $()^+$ indicates the evlaution is from the right, and a $()^-$ indicates the evaluation is from the left. The joint boundary conditions require SGR of $\boldsymbol{u}$ and $\boldsymbol{Q}$ at the joint. In particular, Eq. (4.22) requires that $\boldsymbol{u}_{,x}$ be reconstructed from both the left and the right sides of the joint, and Eq. (4.23) requires that $\boldsymbol{Q}_{,x}$ be reconstructed in each of the three structural members adjacent to the joint. When conducting SGR to approximate these terms, it is essential that the patches are adjacent to the joint, but do not cross over it. This is necessary to capture any discontinuities that may be present.

Aeroelastic Analysis Tool

Nastran's OpenFSI solver is used to solve the aeroelastic gust response. OpenFSI allows the user to define an external fluid solver which can update the loads throughout the dynamic solution. Nastran communicates with the external solver in order to pass system information. Nastran passes displacement, velocity, and acceleration data to the external solver. The external solver is responsible for specifying the updated loads applied to the structural nodes. OpenFSI is capable of explicit or implicit time integration. An explicit method was utilized for the results presented here. More details can be found in the MSC User Defined Services Manual. (MSC Software Corporation, 2010b)

OpenFSI is executed to solve the original aeroelastic system. SGR is conducted using the time history of the solution to generate time histories of gradient terms that appear in the local CSE boundary conditions. These boundary conditions are applied during a second execution of OpenFSI. The local CSE solution is then transformed into a total sensitivity solution by adding the convective term, as shown in Eq. (2.4). This requires SGR of the displacement response throughout the domain, Ω, evaluated at discrete nodes of interest.

Gust Response and Design Derivative Results

Figure 3.4 shows the time histories of vertical displacement at the free end of the beam and at the joint. It also shows the design derivative of these responses with respect to the length of the beam, L. The design derivatives calculated using the local continuum method with SGR are compared to finite difference calculations which utilized a 1e-3 step size. Both beams in the structure were modeled with 20 finite elements. Both the analysis and sensitivity analysis were sufficiently converged using 20 finite elements per beam and 1000 time steps. Often finer spatial and temporal meshes are required to obtain the same degree of accuracy for the CSA as for the original analysis. Three-layer patches and second-order Taylor series expansions were used when employing SGR for this problem. Effects of the SGR parameters on the accuracy of the design derivative calculations are investigated in more detail later. Figure 3.4 shows strong agreement between the continuum and finite difference results. The absolute relative error, $\epsilon_{\dot{w}}$ between the continuum and finite difference solutions of the total design derivative of vertical displacement, $\dot{w}_{_{LC}}$ and $\dot{w}_{_{FD}}$, respectively, is calculated at the time of peak response ($t_p = 3.67s$) and at two locations, i.e. $x = L$ and $x = x_{_J}$.

$$\epsilon_{\dot{w}}(L, t_p) = \left| \frac{\dot{w}_{_{LC}}(L, t_p) - \dot{w}_{_{FD}}(L, t_p)}{\dot{w}_{_{FD}}(L, t_p)} \right| = 0.0268 \tag{3.17}$$

$$\epsilon_{\dot{w}}(x_{_J}, t_p) = \left| \frac{\dot{w}_{_{LC}}(x_{_J}, t_p) - \dot{w}_{_{FD}}(x_{_J}, t_p)}{\dot{w}_{_{FD}}(x_{_J}, t_p)} \right| = 0.0255 \tag{3.18}$$

Calculating the design derivative of the root bending moment time history is also of interest. The root bending moment is an internal force, and therefore, a means to calculate the design derivatives of interal forces is needed. Internal forces for a specific element can be calculated by using

$$\left\{F^{(e)}\right\} = \left[K^{(e)}\right]\left\{u^{(e)}\right\} \tag{3.19}$$

where, $\{F^{(e)}\}$ is a vector of elemental forces for the eth element, $\left[K^{(e)}\right]$ is the elemental stiffness matrix for the eth element, and $\{u^{(e)}\}$ is a vector of displacements for the eth

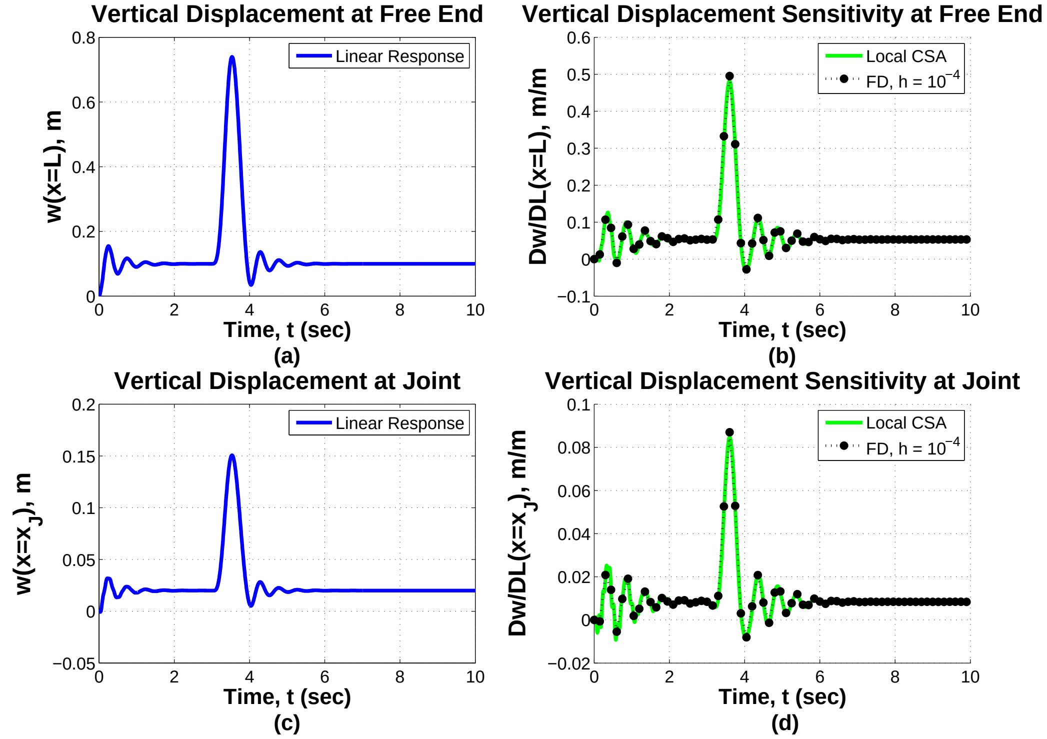


Figure 3.4: Joined Beam Aeroelastic Gust Response. (a) Vertical Displacement at Free End (b) Design derivative of Vertical Displacement at Free End w.r.t. L (c) Vertical Displacement at Joint (d) Design derivative of Vertical Displacement at Joint w.r.t. L

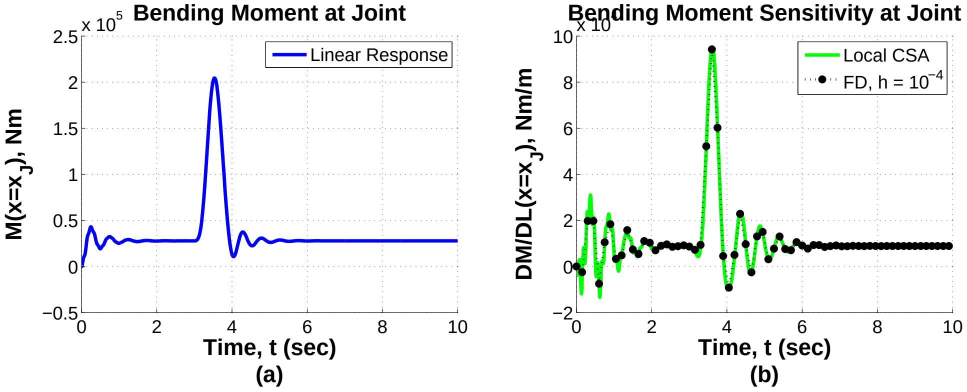


Figure 3.5: Joint Bending Moment Response (a) and design derivative w.r.t to L (b)

element. The total sensitivities of the internal forces can be calculated using

$$\{\dot{F^{(e)}}\} = \left\{F^{(e)}\right\}' + \nabla\left\{F^{(e)}\right\} \cdot \left\{\mathcal{V}^{(e)}\right\} \tag{3.20}$$

$$\left\{F^{(e)}\right\}' = \left[K^{(e)}\right]\left\{u^{(e)}\right\}' \tag{3.21}$$

In Eq. (3.21) the local design derivatives of displacement, $\left\{u^{(e)}\right\}'$ are used to calculate the local design derivatives of elemental forces, $\left\{F^{(e)}\right\}'$. Eq. (3.20) can then be used to calculate the total design derivatives of elemental forces, $\{\dot{F^{(e)}}\}$. Figure 3.5 shows the time histories of the bending moment and the design derivative of the bending moment with respect to the length of the beam, L, at the joint. The continuum results agree well with the finite difference results. The absolute relative error, $\epsilon_{\dot{M}}$ between the continuum and finite difference solutions of the total design derivative of internal bending moment, $\dot{M}_{LC}$ and $\dot{M}_{FD}$, respectively, is calculated at the time of peak response ($t_p = 3.67s$) and the joint location ($x = x_J$).

$$\epsilon_{\dot{M}}(x_J, t_p) = \left|\frac{\dot{M}_{LC}(x_J, t_p) - \dot{M}_{FD}(x_J, t_p)}{\dot{M}_{FD}(x_J, t_p)}\right| = 0.0069 \tag{3.22}$$

3.2 Application to Plate and Beam-Stiffened Plate Models

3.2.1 Linear Static Bending of a Simply-Supported Rectangular Plate

Comparisons of analytic and finite element solutions of the linear static bending of a rectangular plate are presented. Timoshenko, S. (1940) used Kirchhoff plate theory when deriving analytic solutions for linear bending of rectangular plates. The finite element analysis was conducted with Nastran using Kirchhoff CQUAD4 and CQUAD8 elements. Analytic shape design derivatives were derived from the analytic solutions, and local continuum shape design derivatives are calculated from the finite element solutions using local CSA with SGR. Comparisons of analytic and continuum results are presented. Choi, K. and Kim, N.H. (2005) derive continuum sensitivity solutions for plate problems using total CSA and what they refer to as the boundary method. The boundary method has similar properties to local CSA, but it still requires knowledge of the analysis formulation or source code. For instance, using either of those methods requires presumption of the plate theory being used. The formulation of the local CSA method with SGR presented here holds for both Kirchhoff or Mindlin-Reissner plate theory, and can be extended to nonlinear plate theory as well.

Model Information and Governing Equations

The governing equation according to Kirchhoff plate theory solution is a special case of Eq. (2.1)

$$A(w;\boldsymbol{b}) = \frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^2 w}{\partial x^2}\frac{\partial^2 w}{\partial y^2} + \frac{\partial^4 w}{\partial y^4} = f(x,y;\boldsymbol{b}) \tag{3.23}$$

where $w(x,y;\boldsymbol{b})$ is the transverse displacement. The plate lies in the x-y plane and has the distributed transverse load

$$f(x,y;\boldsymbol{b}) = \frac{q_0}{D}\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.24}$$

where a and b are the length dimensions of the plate in the x- and y- directions, respectively, q_0 is the load intensity at the center of the plate, and D is the flexural rigidity of an isotropic plate.

$$D = \frac{Et^3}{12\left(1-\nu^2\right)} \tag{3.25}$$

Since Kirchhoff plate thoery assumes small deflections and rotations, the plate rotations are related to the slopes by the expressions in Eq. (3.26).

$$\psi_x(w) = \frac{\partial w(x,y)}{\partial y} \quad , \quad \psi_y(w) = -\frac{\partial w(x,y)}{\partial x} \tag{3.26}$$

Furthermore, the moments are related to w by the expressions in Eq. (3.27).

$$\begin{aligned} M_x(w) &= D\left(\frac{\partial^2 w(x,y)}{\partial x^2} + \nu \frac{\partial^2 w(x,y)}{\partial y^2}\right) \\ M_y(w) &= D\left(\frac{\partial^2 w(x,y)}{\partial y^2} + \nu \frac{\partial^2 w(x,y)}{\partial x^2}\right) \end{aligned} \tag{3.27}$$

The simply-supported boundary conditions at $x = 0$ and $x = a$ are a special case of Eq. (2.2).

$$\boldsymbol{B}_{\Gamma_y}(w;\boldsymbol{b}) = \boldsymbol{g}_{\Gamma_y}(x,y;\boldsymbol{b}) = \left.\{w, \psi_x(w), M_x(w)\}^T\right|_{x=0,a} = \{0,0,0\}^T \tag{3.28}$$

The simply-supported boundary conditions at $y = 0$ and $y = b$ are also a special case of Eq. (2.2).

$$\boldsymbol{B}_{\Gamma_x}(w;\boldsymbol{b}) = \boldsymbol{g}_{\Gamma_x}(x,y;\boldsymbol{b}) = \left.\{w, M_y(w), \psi_y(w)\}^T\right|_{y=0,b} = \{0,0,0\}^T \tag{3.29}$$

The properties of the plate for the solution shown here are $a = 1m$, $b = 1.5m$, $q_0 = 100000N/m$, $E = 7e10N/m^2$, $t = 0.01m$, and $\nu = 0.25$. A linear static finite element analysis is conducted in Nastran with a 30-by-30 mesh of quadrilateral (CQUAD4) plate elements. In Figures (3.6) through (3.12) the surface represents the analytic solution, and the black dots represent the computational solution as approximated by the finite element analysis, SGR, and CSA. The only shape sensitivity solution presented here is the sensitivity of the plate response to a change in the plate dimension a. Therefore, it can be stated that $\boldsymbol{b} = a$.

Displacements

The analytic solutions of transverse displacement, rotation about the x-axis, and rotation about the y-axis are provided below. Figure 3.6 shows the finite element solution to be accurate, compared to the analytic derivative.

$$w(x, y) = \frac{q_0}{D\pi^4} \frac{a^4 b^4}{\left(a^2 + b^2\right)^2} \sin \frac{\pi x}{a} \sin \frac{\pi y}{b} \tag{3.30}$$

$$\psi_x(x, y) = \frac{q_0}{D\pi^3} \frac{a^4 b^3}{\left(a^2 + b^2\right)^2} \sin \frac{\pi x}{a} \cos \frac{\pi y}{b} \tag{3.31}$$

$$\psi_y(x, y) = -\frac{q_0}{D\pi^3} \frac{a^3 b^4}{\left(a^2 + b^2\right)^2} \cos \frac{\pi x}{a} \sin \frac{\pi y}{b} \tag{3.32}$$

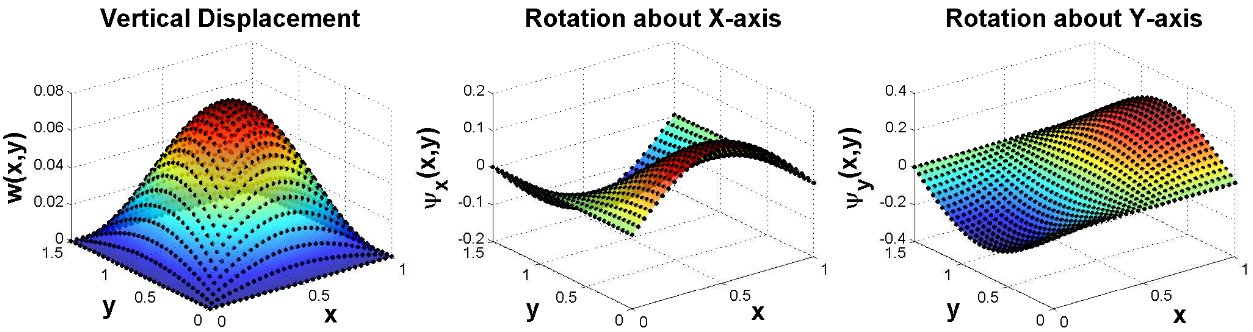


Figure 3.6: Plate displacements (surface: analytic solution, dots: finite element solution)

Displacement Gradients

When the shape design variable a is perturbed, the plate geometry changes only in the x-direction. Therefore, the sensitivity analysis only requires spatial derivatives of response variables in the x-direction. The analytic solutions of the spatial derivatives in the x-direction of transverse displacement, rotation about the x-axis, and rotation about the y-axis are shown in Eqs. (3.33) through (3.35). Figure 3.7 shows that the SGR results computed from the finite element solution are accurate.

$$w_{,x}(x,y) = \frac{q_0}{D\pi^3}\frac{a^3b^4}{\left(a^2+b^2\right)^2}\cos\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.33}$$

$$\psi_{x,x}(x,y) = \frac{q_0}{D\pi^2}\frac{a^3b^3}{\left(a^2+b^2\right)^2}\cos\frac{\pi x}{a}\cos\frac{\pi y}{b} \tag{3.34}$$

$$\psi_{y,x}(x,y) = \frac{q_0}{D\pi^2}\frac{a^2b^4}{\left(a^2+b^2\right)^2}\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.35}$$

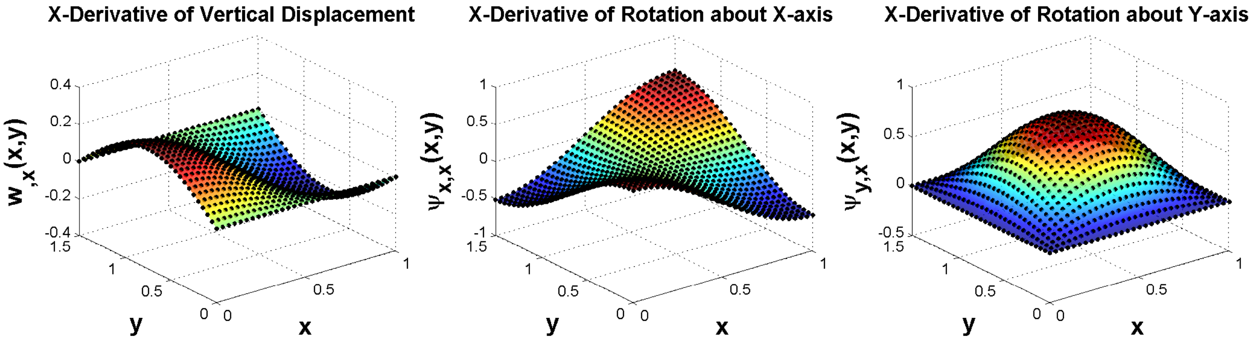


Figure 3.7: Spatial derivatives of plate displacement (surface: analytic solution, dots: finite element solution)

Internal Forces and Internal Force Gradients

The x-derivative of moments about the x- and y- axes are the only remaining derivatives to appear in the local CSE boundary conditions. The analytic solutions of the moment about the y-axis and its spatial derivative in the x-direction are given in Eqs. (3.36) and (3.37) and shown along with finite element and SGR approximations in Figure 3.8.

$$M_x(x,y) = D\left(\frac{\partial^2 w}{\partial x^2}+\nu\frac{\partial^2 w}{\partial y^2}\right) = \frac{q_0}{\pi^2}\frac{\left(a^2b^4+\nu a^4b^2\right)}{\left(a^2+b^2\right)^2}\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.36}$$

$$M_{x,x}(x,y) = D\frac{\partial}{\partial x}\left(\frac{\partial^2 w}{\partial x^2}+\nu\frac{\partial^2 w}{\partial y^2}\right) = \frac{q_0}{\pi}\frac{\left(ab^4+\nu a^3b^2\right)}{\left(a^2+b^2\right)^2}\cos\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.37}$$

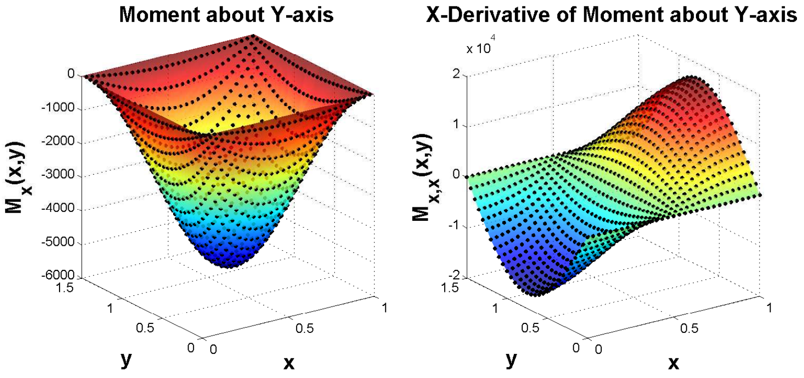


Figure 3.8: Moment about y-axis, M_x (left) and spatial derivative of moment about y-axis, $M_{x,x}$ (right) (surface: analytic solution, dots: finite element and SGR solutions)

The analytic solution of the moment about the x-axis and its spatial derivative in the x-direction are given in Eqs. (3.38) and (3.39) and shown along with finite element and SGR approximations in Figure 3.9.

$$M_y(x,y) = D\left(\frac{\partial^2 w}{\partial y^2} + \nu\frac{\partial^2 w}{\partial x^2}\right) = \frac{q_0}{\pi^2}\frac{(a^4b^2 + \nu a^2b^4)}{(a^2+b^2)^2}\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.38}$$

$$M_{y,x}(x,y) = D\frac{\partial}{\partial x}\left(\frac{\partial^2 w}{\partial y^2} + \nu\frac{\partial^2 w}{\partial x^2}\right) = \frac{q_0}{\pi}\frac{(a^3b^2 + \nu ab^4)}{(a^2+b^2)^2}\cos\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.39}$$

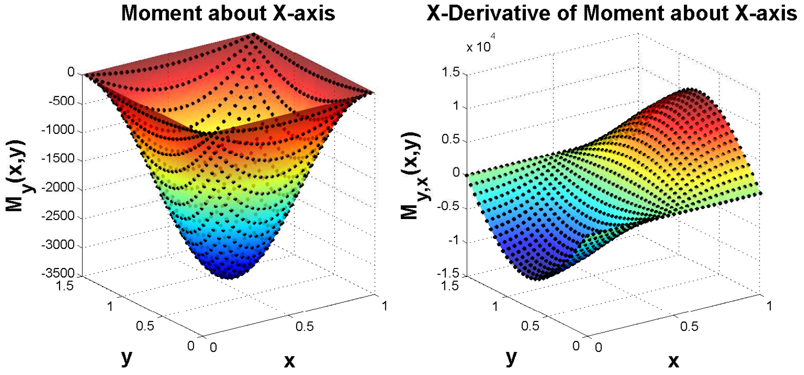


Figure 3.9: Moment about x-axis, M_y (left) and spatial derivative of moment about x-axis, $M_{y,x}$ (right) (surface: analytic solution, dots: finite element and SGR solutions)

Following the implementation used by Duvigneau, R. and Pelletier, D. (2006), the moment gradients, $M_{x,x}$ and $M_{y,x}$, could be reconstructed from the higher-order derivatives of w, which appear in Eqs. (3.37) and (3.39). This requires specific knowledge of the boundary operator formulation, and therefore would require separate sensitivity formulations for different plate theories and element types. The SGR method used here reconstructs the moment gradients, $M_{x,x}$ and $M_{y,x}$, directly from the finite element approximation of the moments, M_x and M_y. This depends only on the analysis output, which means a single sensitivity algorithm can be used for multiple plate theories and element types.

Local Continuum Sensitivity Equations and Boundary Conditions

The governing equation of the local CSE system with respect to the plate dimension a is

$$A(w';\boldsymbol{b}) = f'(x,y;\boldsymbol{b}) \tag{3.40}$$

where, $f'(x,y;\boldsymbol{b})$ is the local design derivative of the distributed load. Recall from Eq.(2.4) that $(\cdot)'$ denotes a partial derivative with respect to the design variable. In this case, the design variable is the plate dimension a.

$$f'(x,y;\boldsymbol{b}) = -\frac{q_0}{D}\frac{\pi}{a^2}x\cos\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.41}$$

Derivation of the local CSE boundary conditions requires formulation of the design velocity, which parameterizes the design derivative of the domain with respect to the design variable a.

$$\boldsymbol{\mathcal{V}}(x,y,a) = \mathcal{V}_x\hat{x} + \mathcal{V}_y\hat{y} = \frac{x}{a}\hat{x} \tag{3.42}$$

Where, $\hat{x}$ and $\hat{y}$ are unit normals in Cartesian space. The local CSE boundary conditions at $x = 0$ are

$$w'(0, y) = \dot{w}(0, y) - w_{,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{3.43}$$

$$\psi'_x(0, y) = \dot{\psi}_x(0, y) - \psi_{x,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{3.44}$$

$$M'_x(0, y) = \dot{M}_x(0, y) - M_{x,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{3.45}$$

The local CSE boundary conditions at $x = a$ are

$$w'(a, y) = \dot{w}(a, y) - w_{,x}(a, y)\mathcal{V}_x(a, y) = -w_{,x}(a, y) \tag{3.46}$$

$$\psi'_x(a, y) = \dot{\psi}_x(a, y) - \psi_{x,x}(a, y)\mathcal{V}_x(a, y) = -\psi_{x,x}(a, y) \tag{3.47}$$

$$M'_x(a, y) = \dot{M}_x(a, y) - M_{x,x}(a, y)\mathcal{V}_x(a, y) = -M_{x,x}(a, y) \tag{3.48}$$

The local CSE boundary conditions at $y = 0$ are

$$w'(x, 0) = \dot{w}(x, 0) - w_{,x}(x, 0)\mathcal{V}_x(x, 0) = -w_{,x}(x, 0) \cdot \frac{x}{a} \tag{3.49}$$

$$M'_y(x, 0) = \dot{M}_y(x, 0) - M_{y,x}(x, 0)\mathcal{V}_x(x, 0) = -M_{y,x}(x, 0) \cdot \frac{x}{a} \tag{3.50}$$

$$\psi'_y(x, 0) = \dot{\psi}_y(x, 0) - \psi_{y,x}(x, 0)\mathcal{V}_x(x, 0) = -\psi_{y,x}(x, 0) \cdot \frac{x}{a} \tag{3.51}$$

The local CSE boundary conditions at $y = b$ are

$$w'(x, b) = \dot{w}(x, b) - w_{,x}(x, b)\mathcal{V}_x(x, b) = -w_{,x}(x, b) \cdot \frac{x}{a} \tag{3.52}$$

$$M'_y(x, b) = \dot{M}_y(x, b) - M_{y,x}(x, b)\mathcal{V}_x(x, b) = -M_{y,x}(x, b) \cdot \frac{x}{a} \tag{3.53}$$

$$\psi'_y(x,b) = \dot{\psi}_y(x,b) - \psi_{y,x}(x,b)\mathcal{V}_x(x,b) = -\psi_{y,x}(x,b) \cdot \frac{x}{a} \tag{3.54}$$

Traditional implementation of the local CSA method would have the right hand sides of Eqs. (4.28) through (4.45) be recovered from the finite element solution and shape functions. Instead, SGR is used to recover the necessary derivatives, eliminating the need to know the shape functions, enabling an element agnostic method. Figure 3.10 shows how the SGR approximation of the boundary conditions compares to the analytic boundary conditions. The SGR accurately approximates the local CSE boundary conditions at $x = 0$ and $x = a$ as shown in the upper plots of Figure 3.10. The lower plots in Figure 3.10 clearly illustrate that SGR introduces some numerical error. In this particular case, the numerical error in the boundary conditions had a negligible effect on the sensitivity solution. In the event that it has a significant effect, several things can be done to reduce the error. They are to refine the mesh, add more layers per patch during SGR, increase the Taylor series order during SGR, and/or impose physical boundary conditions where they are known instead of using SGR to reconstruct them from the output of the discretized analysis.

Local Design Derivatives

The analytic local design derivative solutions of transverse displacement, rotation about the x-axis, and rotation about the y-axis are found by taking the partial derivative of Eqs. (3.30-3.32) with respect to the plate length, a.

$$\begin{aligned} w'(x,y) = &-\frac{q_0}{D\pi^3}\frac{a^2b^4}{\left(a^2+b^2\right)^2} x \cos\frac{\pi x}{a} \sin\frac{\pi y}{b} + ... \\ &... + \frac{q_0}{D\pi^4}\frac{4}{\left(a^2+b^2\right)^2}\left(-\frac{a^5b^4}{(a^2+b^2)} + a^3b^4\right)\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \end{aligned} \tag{3.55}$$

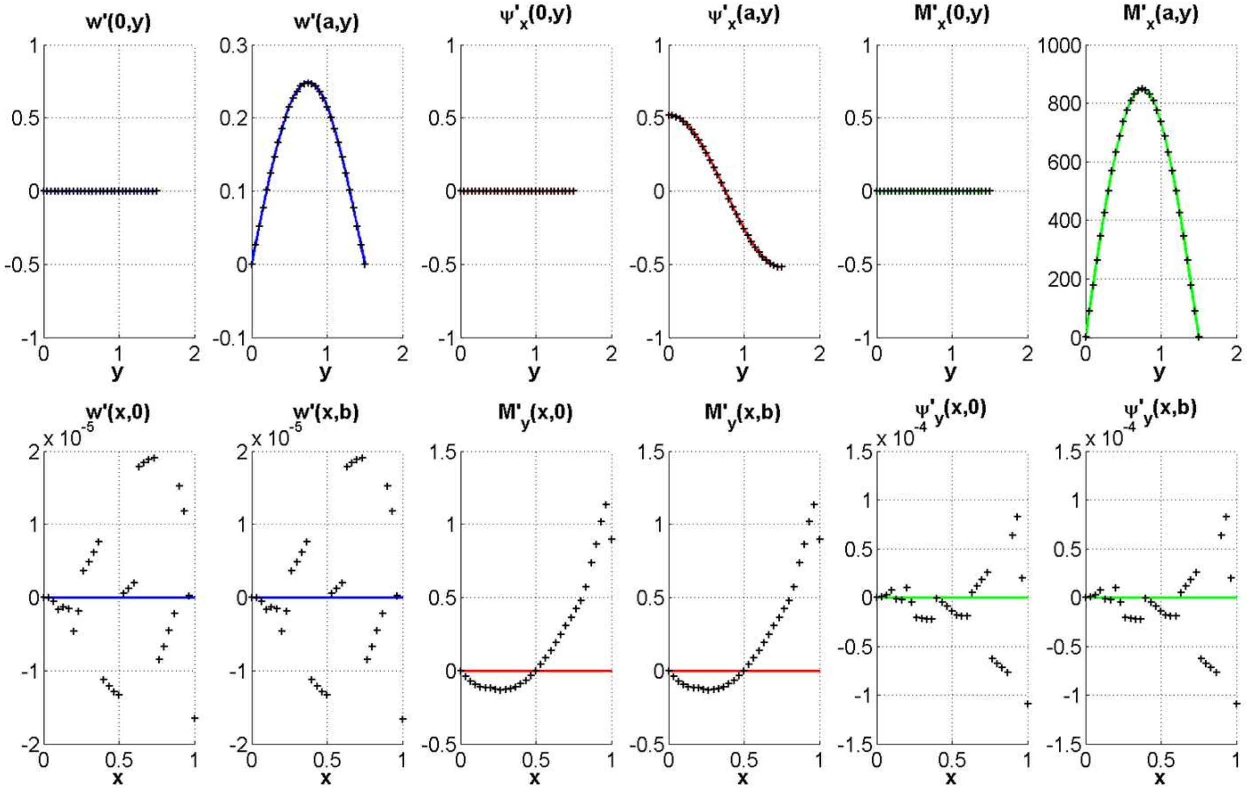


Figure 3.10: Local CSE boundary conditions corresponding to Eqs. (4.28-4.35) in the upper plots and Eqs. (4.38-4.45) in the lower plots. Solid lines represent analytic boundary condition values, and the black markers represent the boundary conditions as approximated by the SGR approach.

$$\begin{aligned}\psi'_x(x,y) &= -\frac{q_0}{D\pi^2}\frac{a^2b^3}{\left(a^2+b^2\right)^2}x\cos\frac{\pi x}{a}\cos\frac{\pi y}{b}+...\\ &...+\frac{q_0}{D\pi^3}\frac{4}{\left(a^2+b^2\right)^2}\left(-\frac{a^5b^3}{\left(a^2+b^2\right)}+a^3b^3\right)\sin\frac{\pi x}{a}\cos\frac{\pi y}{b}\end{aligned} \tag{3.56}$$

$$\begin{aligned}\psi'_y(x,y) &= -\frac{q_0}{D\pi^2}\frac{ab^4}{\left(a^2+b^2\right)^2}x\sin\frac{\pi x}{a}\sin\frac{\pi y}{b}+...\\ &...+\frac{q_0}{D\pi^3}\frac{1}{\left(a^2+b^2\right)^2}\left(\frac{4a^4b^4}{\left(a^2-b^2\right)}+3a^2b^4\right)\cos\frac{\pi x}{a}\sin\frac{\pi y}{b}\end{aligned} \tag{3.57}$$

Figure 3.11 shows a comparison of the local CSA with SGR and analytic local design derivative solutions. The accuracy of the method is quantified in a later section, but the numerical error in the local CSE boundary conditions appears graphically to have minimal impact.

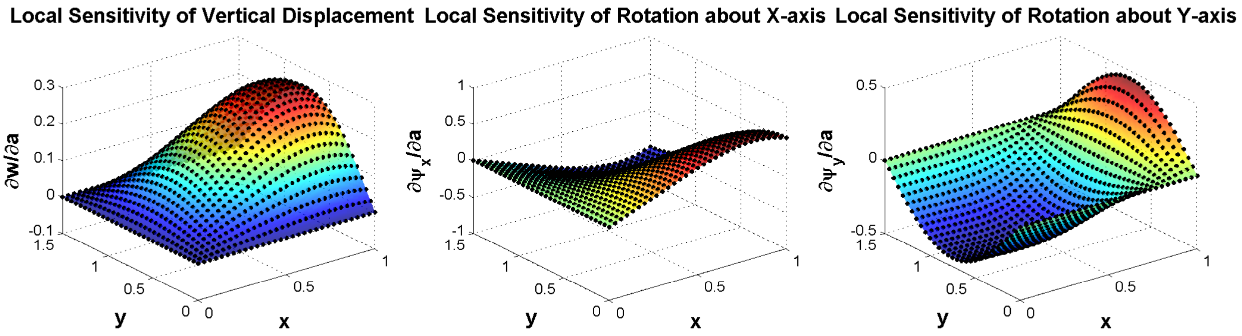

Local Sensitivity of Vertical Displacement Local Sensitivity of Rotation about X-axis Local Sensitivity of Rotation about Y-axis

Figure 3.11: Local design derivatives of displacement w.r.t. plate length, a (surface: analytic, dots: local CSA with SGR)

Total Design Derivatives

The analytic total sensitivity solutions of transverse displacement, rotation about the x-axis, and rotation about the y-axis are are found by taking the material derivative of Eqs. (3.30-3.32) with respect to the plate length, a.

$$\dot{w}(x,y) = \frac{q_0}{D\pi^4}\frac{4}{\left(a^2+b^2\right)^2}\left(-\frac{a^5b^4}{\left(a^2+b^2\right)}+a^3b^4\right)\sin\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.58}$$

$$\dot{\psi}_x(x,y) = \frac{q_0}{D\pi^3}\frac{4}{(a^2+b^2)^2}\left(-\frac{a^5b^3}{(a^2+b^2)}+a^3b^3\right)\sin\frac{\pi x}{a}\cos\frac{\pi y}{b} \tag{3.59}$$

$$\dot{\psi}_y(x,y) = \frac{q_0}{D\pi^3}\frac{1}{(a^2+b^2)^2}\left(\frac{4a^4b^4}{(a^2+b^2)}-3a^2b^4\right)\cos\frac{\pi x}{a}\sin\frac{\pi y}{b} \tag{3.60}$$

Figure 3.12 compares the total design derivatve solution computed from local CSA with SGR and the total design derivative solution derived analytically. The total design derivatives were computed by adding the convective term to the local CSA with SGR solution. Eqs. (3.55) through (3.60) satisfy the relationship between the local and total derivatives as formulated in Eq. (2.3). Figure 3.12 provides a strong indication that the method is accurate. The following section quantifies the accuracy and discusses it in greater detail.

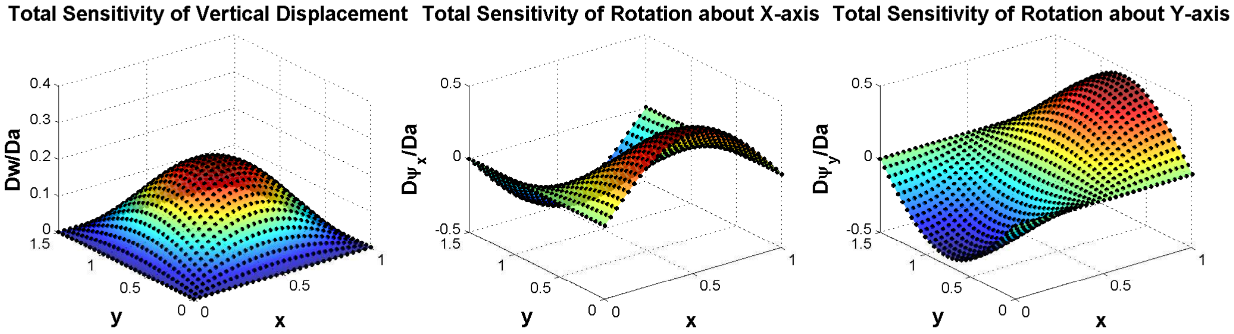


Figure 3.12: Total design derivatives of displacement w.r.t. plate length, a (surface: analytic, dots: local CSA with SGR)

Mesh Convergence Study

There are, primarily, three potential sources of error in the local CSA solution. They are discretization error in the original finite element analysis, error in the SGR computation, and discretization error in the local CSE finite element analysis. The simply-supported rectangular plate model was analyzed for a variety of mesh sizes to ensure the finite element analysis and the local CSA analysis are converged. The mesh convergence study was conducted using bilinear CQUAD4 elements as well as eight-noded serendipity quadrilateral elements (CQUAD8). In addition, both sinusoidal and uniform distributed loads were applied. The analytic solution for the uniform load is also provided in Timoshenko, S. (1940). Table 3.2.1 contains the error norms for the transverse displacement, w, its spatial derivative in the x-direction, $w_{,x}$, and its total design derivative with respect to the plate dimension a, $\dot{w}$. The first two columns in the table indicate the number of elements per side of the plate and the total number of finite element nodes in the system. Nastran's PLOAD4 data card was used to specify the uniform loading. The PLOAD4 data card cannot be used to accurately represent the sinusoidal loading. Therefore, in order to accurately apply the sinusoidal loading, consistent nodal loads were calculated and specified at each grid point using Nastran's FORCE data card. The error norms were calculated by

$$\epsilon_\phi = \frac{\|\phi_{approx} - \phi_{true}\|_\infty}{\|\phi_{true}\|_\infty} \times 100 \tag{3.61}$$

where, ϕ represents a general response variable.

Table 3.1: True Absolute Percent Relative Errors for Displacement Degree of Freedom

CQUAD4		Sinusoidal			Uniform		
# Elems	# Nodes	ϵ_w	$\epsilon_{w,x}$	$\epsilon_{\dot{w}}$	ϵ_w	$\epsilon_{w,x}$	$\epsilon_{\dot{w}}$
5	36	1.136	12.991	12.995	1.772	11.495	14.970
10	121	0.756	3.575	4.517	0.896	4.086	4.437
20	441	0.264	0.481	0.356	0.284	0.483	0.521
30	961	0.127	0.170	0.066	0.134	0.152	0.195

CQUAD8		Sinusoidal			Uniform		
# Elems	# Nodes	ϵ_w	$\epsilon_{w,x}$	$\epsilon_{\dot{w}}$	ϵ_w	$\epsilon_{w,x}$	$\epsilon_{\dot{w}}$
5	96	0.153	4.900	3.770	0.098	6.090	4.916
10	341	0.012	0.484	0.779	0.009	0.699	0.733
20	1281	0.006	0.038	0.086	0.002	0.053	0.213
30	2821	0.004	0.007	0.079	0.001	0.016	0.080

The results indicate strong convergence overall. As expected, when comparing the transverse displacement errors, it is clear that the CQUAD8s outperform the CQUAD4s because they are biquadratic instead of bilinear. SGR with seven-layer patches and fourth-order Taylor series expansions was used to approximate the spatial derivatives. As expected the SGR calculations converge as more nodes are added to the system. There is a large jump in the number of nodes going from ten elements per side of the plate to twenty elements per side. Correlating to this jump is a dramatic reduction in the errors of the SGR calculations. Even though the number of patches and the Taylor series order are unchanged, the node density of the patches increases. To make the calculations even more accurate, the number of layers and the Taylor series order should be increased. Lastly, as the errors of the analysis and SGR solutions are reduced, so too are the errors of the CSA solutions. Tables 3.2.1 and 3.3 show the error norms associated with the rotational degrees of freedom. These results have convergence trends similar to the displacement degree of freedom.

Table 3.2: True Absolute Percent Relative Errors for x-Rotational Degree of Freedom

CQUAD4		Sinusoidal			Uniform		
No. Elem	No. Nodes	ϵ_{ψ_x}	$\epsilon_{\psi_{x,x}}$	$\epsilon_{\dot{\psi}_x}$	ϵ_{ψ_x}	$\epsilon_{\psi_{x,x}}$	$\epsilon_{\dot{\psi}_x}$
5	36	1.753	9.439	8.734	2.967	9.708	6.356
10	121	0.025	2.431	3.765	0.816	3.657	3.303
20	441	0.097	0.331	0.356	0.297	0.682	1.025
30	961	0.057	0.111	0.087	0.142	0.343	0.549

CQUAD8		Sinusoidal			Uniform		
No. Elem	No. Nodes	ϵ_{ψ_x}	$\epsilon_{\psi_{x,x}}$	$\epsilon_{\dot{\psi}_x}$	ϵ_{ψ_x}	$\epsilon_{\psi_{x,x}}$	$\epsilon_{\dot{\psi}_x}$
5	96	0.179	3.764	24.899	0.509	5.000	18.770
10	341	0.015	0.386	3.387	0.062	1.317	4.883
20	1281	0.006	0.033	0.216	0.008	0.291	1.494
30	2821	0.003	0.007	0.083	0.002	0.126	0.660

Table 3.3: True Absolute Percent Relative Errors for y-Rotational Degree of Freedom

CQUAD4		Sinusoidal			Uniform		
No. Elem	No. Nodes	ϵ_{ψ_y}	$\epsilon_{\psi_{y,x}}$	$\epsilon_{\dot{\psi}_y}$	ϵ_{ψ_y}	$\epsilon_{\psi_{y,x}}$	$\epsilon_{\dot{\psi}_y}$
5	36	1.394	11.818	12.339	1.966	16.841	31.695
10	121	0.535	2.263	4.274	0.608	4.910	5.199
20	441	0.182	0.147	0.316	0.188	0.975	0.908
30	961	0.088	0.072	0.173	0.088	0.427	0.433

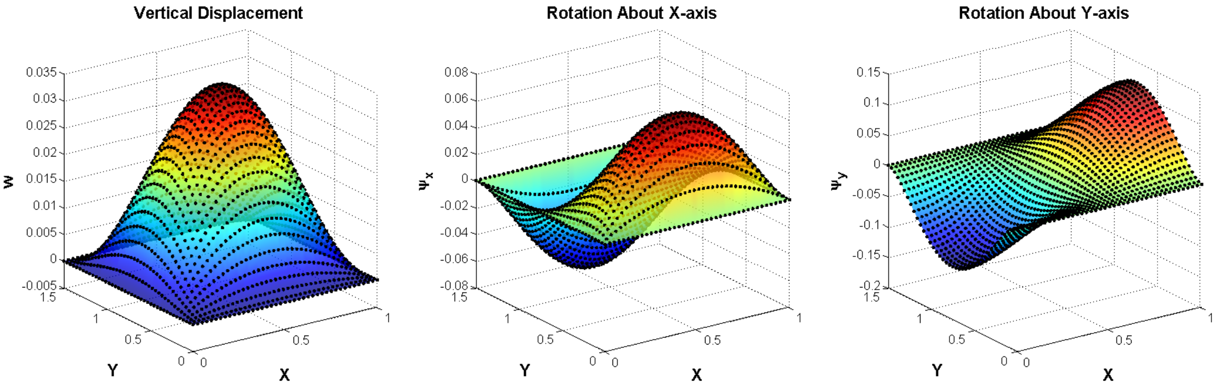
CQUAD8		Sinusoidal			Uniform		
No. Elem	No. Nodes	ϵ_{ψ_y}	$\epsilon_{\psi_{y,x}}$	$\epsilon_{\dot{\psi}_y}$	ϵ_{ψ_y}	$\epsilon_{\psi_{y,x}}$	$\epsilon_{\dot{\psi}_y}$
5	96	0.239	5.631	25.084	0.295	10.051	19.080
10	341	0.015	0.350	4.267	0.038	1.760	5.382
20	1281	0.005	0.025	0.656	0.005	0.425	2.064
30	2821	0.004	0.009	0.338	0.001	0.188	1.000

Furthermore, this study demonstrates that the local CSA method with SGR is element agnostic; it is independent of the element formulation and operates solely on analysis output. In fact, while Kirchhoff plate theory was used here in order to verify CSA computations with analytical results in the literature, Reissner-Mindlin finite elements could have been used without affecting the CSA code or computation at all. Therefore, the sensitivity algorithm does not need any adjustments when running the different cases shown in this convergence study. This would not be possible when implementing other analytic sensitivity methods. It is for this reason that the approach presented here is considered a general formulation.

3.2.2 Linear Static Bending of a Rectangular Plate with Mixed Boundary Conditions

The same plate model from Section 3.2.1 was subjected to a uniform distributed load. The only change to the model was that the plate had clamped edges along $y = 0$ and $y = b$ and simply-supported edges along $x = 0$ and $x = a$. Figure 3.13 shows that the finite element analysis in Nastran agrees well with the analytic solution from Timoshenko, S. (1940).

For the clamped edges all degrees of freedom are fixed on the boundary, and therefore, formulation of the local sensitivity boundary conditions follows Eq. (2.29). The sensitivity boundary conditions for the simply-supported edges follow the same formulation in Eqs. (4.33) through (4.35) and (4.43) through (4.45). Figures 3.14 and 3.15 show the local and total design derivatives of the displacement response with respect to the plate dimension a. The true percent relative errors of the dispalcement and design derivative solutions are shown in Table 3.4. The error norms are calculated in a manner consistent with Eq. (3.61). The results indicate a strong agreement between the approximate and analytic solutions.


Vertical Displacement

Rotation About X-axis

Rotation About Y-axis

Figure 3.13: Displacements of a rectangular plate with two opposite edges clamped and the other two opposite edges simply-supported (surface: analytic, dots: finite element)

Table 3.4: True Absolute Percent Relative Errors for Plate with Mixed Boundary Conditions

	w	w'	$\dot{w}$	ψ_x	ψ'_x	$\dot{\psi}_x$	ψ_y	ψ'_y	$\dot{\psi}_y$
ϵ	0.210	0.374	0.524	0.156	0.273	0.381	0.172	0.305	0.434

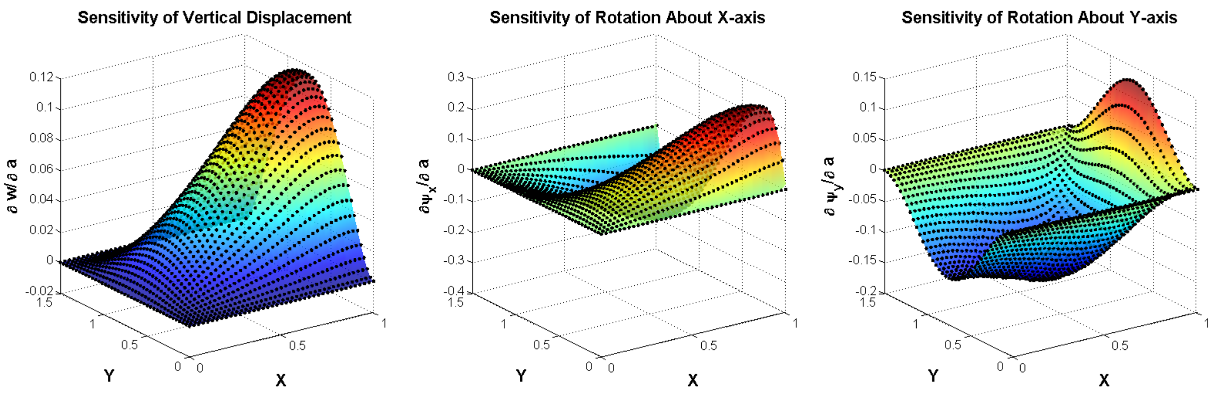


Figure 3.14: Local design derivatives of displacement w.r.t. plate length, a for plate with mixed boundary conditions (surface: analytic, dots: local CSA with SGR)

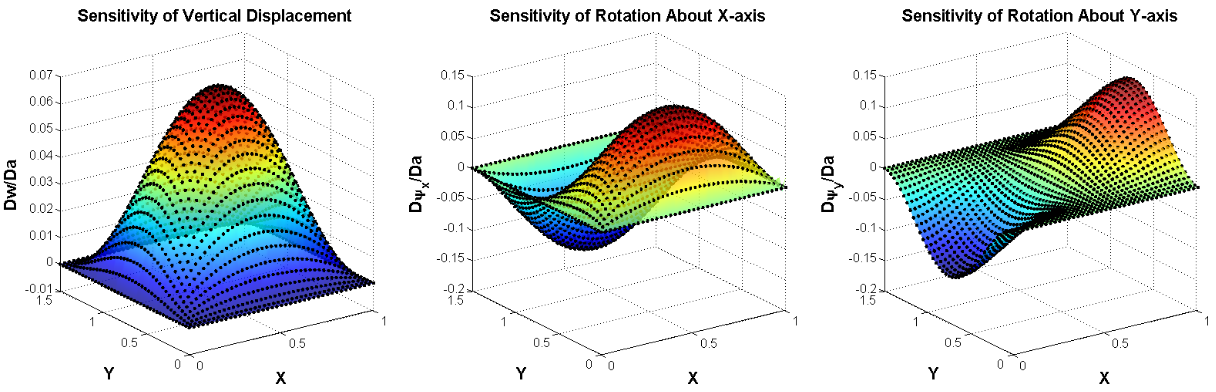


Figure 3.15: Total design derivatives of displacement w.r.t. plate length, a for plate with mixed boundary conditions (surface: analytic, dots: local CSA with SGR)

3.2.3 Linear Static Bending of a Beam-Stiffened Rectangular Plate

The following example is a comparison of local CSA with SGR and finite difference design derivative solutions for a beam-stiffened rectangular plate subjected to a uniform distributed transverse loading. This is another example of how to implement local CSA with SGR when structural interfaces are present. Similar examples are presented by Choi, K. and Kim, N.H. (2005) for total CSA, but this is the first implementation of local CSA for a built-up structure with mixed element types.

Model Information

The same rectangular plate model from 3.2.1 is used again here. The plate, however, is modified by a stiffener that is placed along the $x = a/2 = 0.5m$ axis. The stiffener is modeled as a beam having $E = 7e10N/m^2$, $G = 26e9N/m^2$, $A = 0.0025m^2$, and $I = 1e-6m^4$. The structural nodes of the beam are coincident with the structural nodes of the plate that lie on the $x = 0.5m$ axis. The neutral axis of the beam is not offset from the nodes, but an offset would not affect the sensitivity formulation. All edges of the plate are simply-supported, and the finite element analysis is conducted on a 60-by-40 mesh of CQUAD4 elements. The total design derivatives of displacement with respect to the plate dimension a are calculated.

Displacements of Beam-Stiffened Model

The displacement solution of the stiffened plate is shown in Figure 3.16. The beam introduces a discontinuity in the through-thickness shear strain, which corresponds to the second-order derivatives of ψ_x and ψ_y. The local CSE boundary conditions only depend on first-order derivatives shown in Figure 3.17, and do not include second-order derivatives. Therefore, the shear strain discontinuity does not affect the local CSE boundary conditions. However, the following subsection shows that force discontinuities do play a role in the local CSE boundary conditions. Figure 3.17 shows that the derivative functions are at least C^0 continuous.

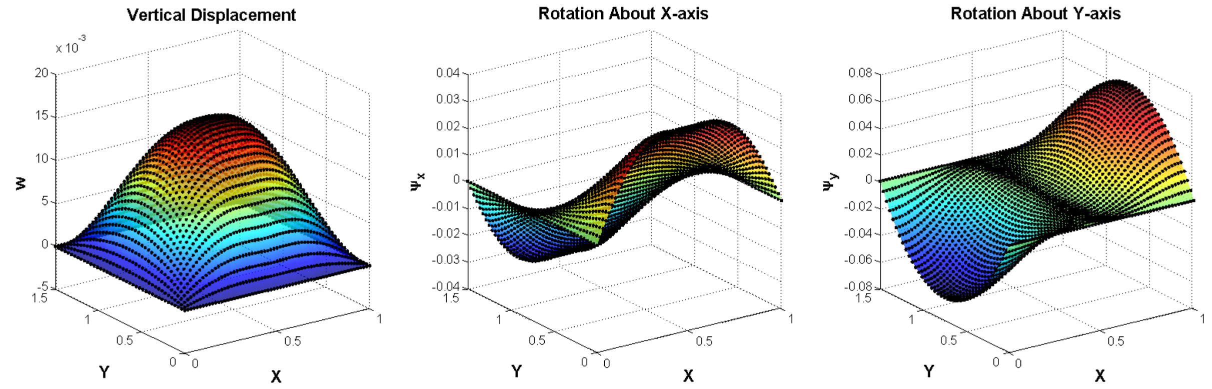


Figure 3.16: Displacements of the beam-stiffened plate (surface: interpolation of finite element solution, dots: finite element nodal solution)

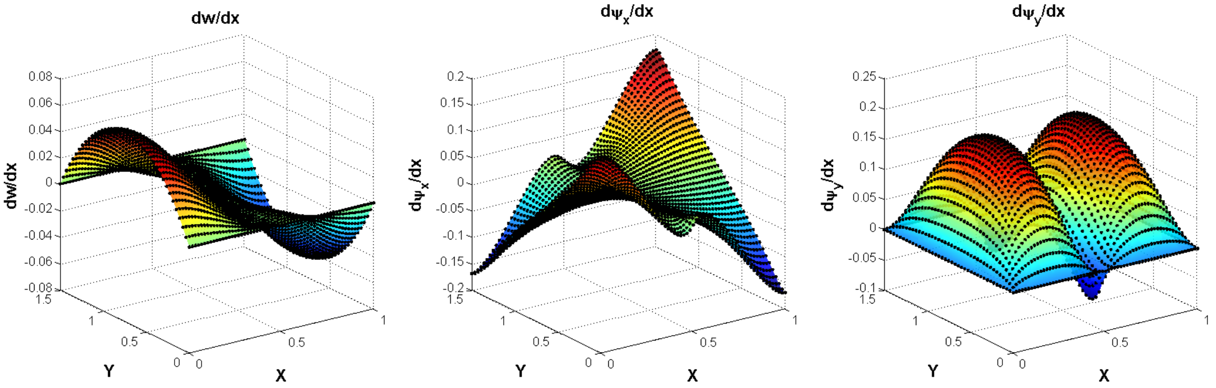


Figure 3.17: Spatial derivatives of displacement of the beam-stiffened plate (surface: interpolation of SGR solution, dots: SGR nodal solution)

Structural Interface Boundary Conditions

The presence of the stiffener creates discontinuities in the spatial derivatives of the internal forces. The additional local CSE boundary conditions shown in Eqs. (3.62) through (3.64) must be imposed.

$$V'_y(\frac{a}{2}^{+}, y) = \left(V_{y,x}(\frac{a}{2}^{+}, y) - V_{y,x}(\frac{a}{2}^{-}, y)\right) \mathcal{V}_x(\frac{a}{2}, y) \tag{3.62}$$

$$M'_x(\frac{a^+}{2}, y) = \Big(M_{x,x}(\frac{a^+}{2}, y) - M_{x,x}(\frac{a^-}{2}, y)\Big)\, \mathcal{V}_x(\frac{a}{2}, y) \tag{3.63}$$

$$M'_y(\frac{a^+}{2}, y) = \Big(M_{y,x}(\frac{a^+}{2}, y) - M_{y,x}(\frac{a^-}{2}, y)\Big)\, \mathcal{V}_x(\frac{a}{2}, y) \tag{3.64}$$

The derivatives in the x-direction of the internal forces are approximated with SGR and shown in Figure 3.18. Once again, it is critical that any patch used for SGR of any response variable does not cross over the structural interface.

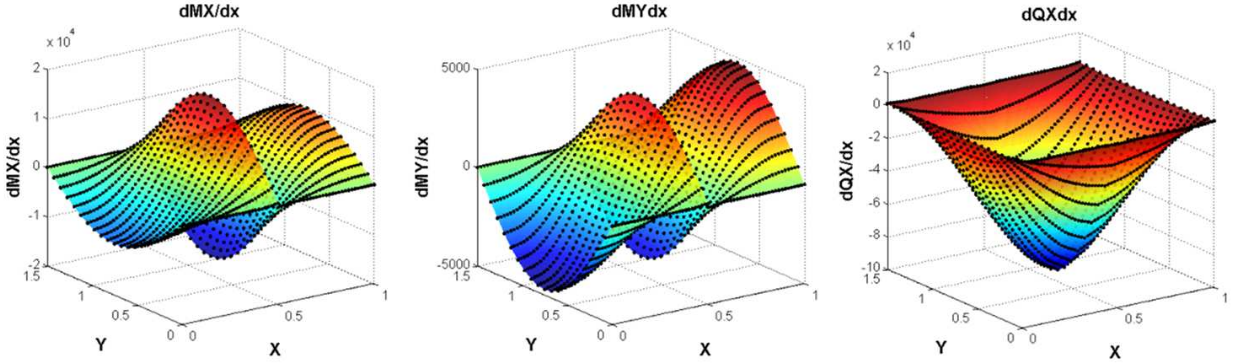


Figure 3.18: Spatial derivatives of forces of the beam-stiffened plate (surface: interpolation of SGR solution, dots: SGR nodal solution)

Design Derivative Results

After solving the local CSE system, the resulting local design derivatives are converted to total design derivatives using the convective term in accordance with Eq. (2.3). Figures 3.19 and 3.20 show the local and total design derivative results using both the local CSA method with SGR and the finite difference method. The percent relative error norm between the continuum and finite difference results are shown in Table 3.5. Eq. (3.65) is used to calculate the error norms.

$$\epsilon_\phi = \frac{\|\phi_{LC} - \phi_{FD}\|_\infty}{\|\phi_{FD}\|_\infty} \times 100 \tag{3.65}$$

Here, ϕ is a general design derivative variable, and the subscripts LC and FD indicate coming from the local continuum and finite difference solution, respesctively. Local CSA with SGR, as evidenced here, can yield accurate design derivatives for built-up structures. However, the additional boundary conditions must be formulated and applied at every structural interface where discontinuities exist, which is not required for total CSA. One possibility is to assemble the additional boundary condition at every structural interface using SGR. If a discontinuity is present, then it is captured appropriately, and if one is not present, then the terms in Eqs. (3.62-3.64) cancel, yielding zero applied load. The latter opens the door to more error contributions, but it eliminates the requirement to know where discontinuities do and do not exist.

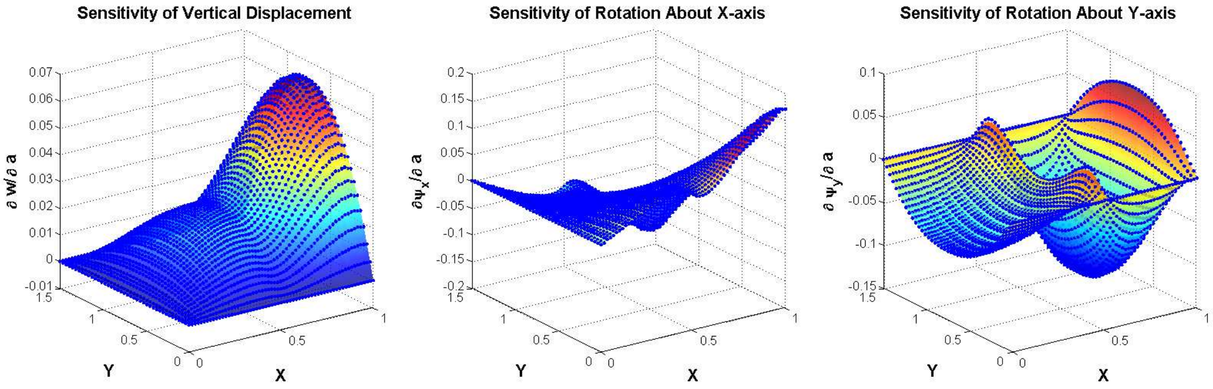


Figure 3.19: Local design derivatives of displacement w.r.t. plate length, a (surface: local CSA with SGR, blue dots: finite difference)

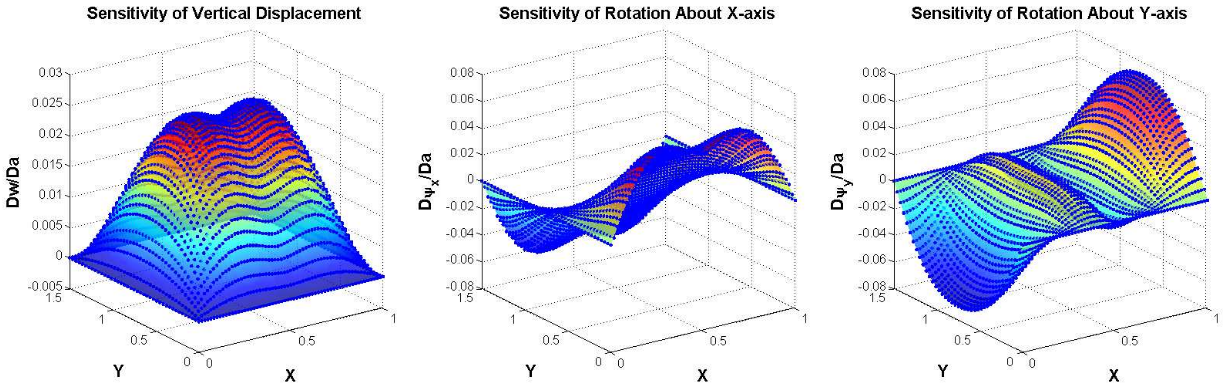


Figure 3.20: Total design derivatives of displacement w.r.t. plate length, a (surface: local CSA with SGR, blue dots: finite difference)

Table 3.5: Absolute Percent Relative Errors for Beam-Stiffened Plate

	w'	$\dot{w}$	ψ'_x	$\dot{\psi}_x$	ψ'_y	$\dot{\psi}_y$
ϵ	0.664	1.659	1.257	3.507	1.308	2.487

3.2.4 Design Derivatives of Stress for Rectangular Plate Bending

In structural shape optimization the objective and constraint functions often depend on stress values. Therefore, accurate computation of stress design derivatives is a necessary capability. This section formulates the design derivatives for plane stress. Design derivatives of stress are calculated using local CSA with SGR for the rectangular plate bending problems presented in 3.2.1 and 3.2.3. The design derivatives are compared to finite difference results.

Shape Design Derivative Formulation for Stresses

The total design derivative of stress is defined as

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{\sigma}' + \nabla\boldsymbol{\sigma} \cdot \boldsymbol{\mathcal{V}} \tag{3.66}$$

Typically the analysis tool, in this case, Nastran, provides stress output. Therefore, $\nabla\boldsymbol{\sigma}$ can be obtained by using the stress output to employ SGR. The design velocity, $\boldsymbol{\mathcal{V}}$, is known from the geometry paramterization. This leaves the local sensitivity, $\boldsymbol{\sigma}'$. The linear stress definition is

$$\boldsymbol{\sigma} = [E]\,[\partial]\,\boldsymbol{u} \tag{3.67}$$

where $\boldsymbol{\sigma}$ is a vector of stress components at a point in the domain, and $\boldsymbol{u}$ is a vector of displacement components at the same point in the domain. The stresses are linearly related to the displacements through a coefficient matrix, $[E]$, and an operator matrix, $[\partial]$. Partial differentiation of Eq. (3.67) with respect to a design variable yields

$$\boldsymbol{\sigma}' = [E]\,[\partial]\,\boldsymbol{u}' \tag{3.68}$$

Eq. (3.68) shows that the local design derivatives of stress, $\boldsymbol{\sigma}'$, are linearly related to the local displacement, $\boldsymbol{u}'$, through the same coefficient and operator matrices. Therefore the solution for the local design derivatives of displacement, $\boldsymbol{u}'$, is used to calculate the total design derivatives of stress, $\dot{\boldsymbol{\sigma}}$. The operator matrix contains partial derivatives with respect to spatial coordinates, which means that $\boldsymbol{\sigma}'$ depends on spatial derivatives of $\boldsymbol{u}'$. These derivatives, like $\nabla\boldsymbol{\sigma}$, can be approximated using SGR. The last obvious detail is determining the definition of the matrices, $[E]$ and $[\partial]$.

If it is unknown how the analysis tool defines these matrices, then an assumed definition can be validated by using Eq. (3.67) to compare calculated stresses with analysis output. If the assumed definition is correct, then it can be used to compute local and total design derivatives of stress as previously described. However, in the event that the stress output cannot be matched, the user has the ability to use his/her own stress-displacement relationship. If this is the case, then it is important that the stress gradient, $\nabla\boldsymbol{\sigma}$, be obtained by using the user's own stress calculation with SGR, instead of using the analysis' stress output.

Plane Stress Formulation

The linear static plate bending analysis in Nastran utilizes plane stress theory for stress evaluations. The stress-displacement relationship for plane stress theory is (Timoshenko, S., 1940)

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = -z \begin{Bmatrix} \frac{E}{1-\nu^2}\left(\frac{\partial\psi_y}{\partial x} - \nu\frac{\partial\psi_x}{\partial y}\right) \\ \frac{E}{1-\nu^2}\left(-\frac{\partial\psi_x}{\partial y} + \nu\frac{\partial\psi_y}{\partial x}\right) \\ \frac{E}{2(1+\nu)}\left(-\frac{\partial\psi_x}{\partial x} + \frac{\partial\psi_y}{\partial y}\right) \end{Bmatrix} \tag{3.69}$$

For the unstiffened plate in 3.2.1, Nastran's displacement solution was used in conjuction with SGR to calculate stresses according to Eq. (3.69). The SGR used five-layer patches and fourth-order Taylor series expansions. The calculated stresses matched Nastran's stress output to within one percent.

Stress Sensitivity Results

First, total design derivatives of stress were calculated for the unstiffened rectangular plate with respect to the plate dimension, a. The local design derivatives of stress were calculated using Eq. (3.68). The local CSA with SGR solution for $\boldsymbol{u}'$ utilized five-layer patches and fourth-order Taylor series expansions. The derivatives of stress in the convective term in Eq. (3.66) were approximated using SGR. Nastran's stress output, three-layer patches, and second-order Taylor series expansions were utilized. Figure 3.21 shows that the total design derivatives (surface) agree well with the finite difference results (blue dots). However, there is some noise present on the boundaries. Figure 3.22 shows that a finer, 40-by-40 finite element mesh helps to eliminate the noise. This is quantified by the error norms of the total design derivatives of stress for the two meshes shown in Table 3.6. These error norms were calculated using Eq. (3.65).

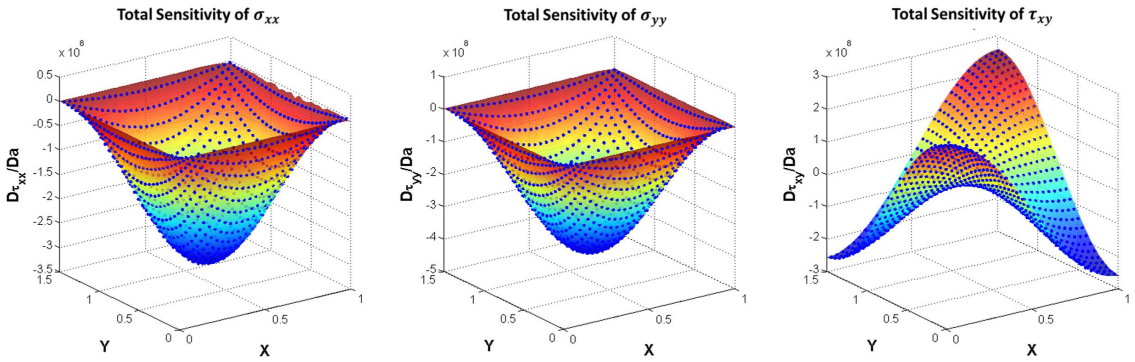


Figure 3.21: Total design derivatives of stress for a 30-by-30 mesh (surface: local CSA with SGR, dots: finite difference)

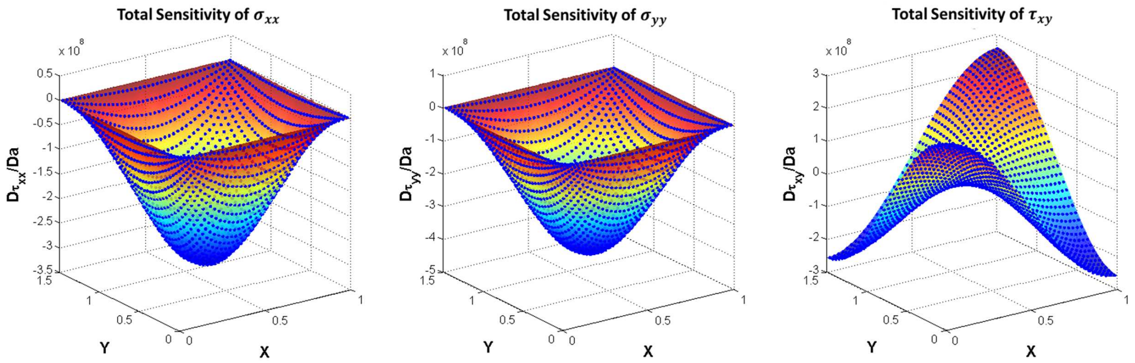


Figure 3.22: Total design derivatives of stress for a 40-by-40 mesh (surface: local CSA with SGR, dots: finite difference)

Table 3.6: Absolute Percent Relative Errors for Design Derivatives of Plate Stresses

	$\dot{\sigma}_{xx}$	$\dot{\sigma}_{yy}$	$\dot{\tau}_{xy}$
$\epsilon_{30\times30}$	4.29	1.62	1.57
$\epsilon_{40\times40}$	2.38	0.88	0.89

Next, total design derivatives of stress were calculated for the beam-stiffened rectangular plate with respect to the plate dimension, a. The local design derivatives of stress were calculated using Eq. (3.68). The local CSA with SGR solution for $\boldsymbol{u}'$ utlized five-layer patches and fourth-order Taylor series expansions. The derivatives of stress in the convective term in Eq. (3.66) were approximated using SGR. Nastran's stress output, three-layer patches, and second-order Taylor series expansions were utilized. Figure 3.23 shows that the total design derivatives (surface) possess moderate to strong agreement with the finite difference results (blue dots). A significant amount of noise is present on the boundaries and at the corners of the structural interface. It should be expected that the design derivatives of stress are less accurate than the design derivatives of displacement, because more approximations are being made with SGR. Figure 3.24 shows the total design derivatives of stress for a finer, 70-by-70 finite element mesh. Here, ten-layer patches and eigth-order Taylor series expansions were used for all executions of SGR. The results improved dramatically, as the numerical noise near the boundaries was diminsihed. This example illustrates quite clearly that that the majority of numerical error is present on the boundaries, thus, adaptive meshing may be advantageous for future implementations. Table 3.7 presents the error norms, calculated with Eq. (3.65), for the two different solutions. The results confirm a significant reduction

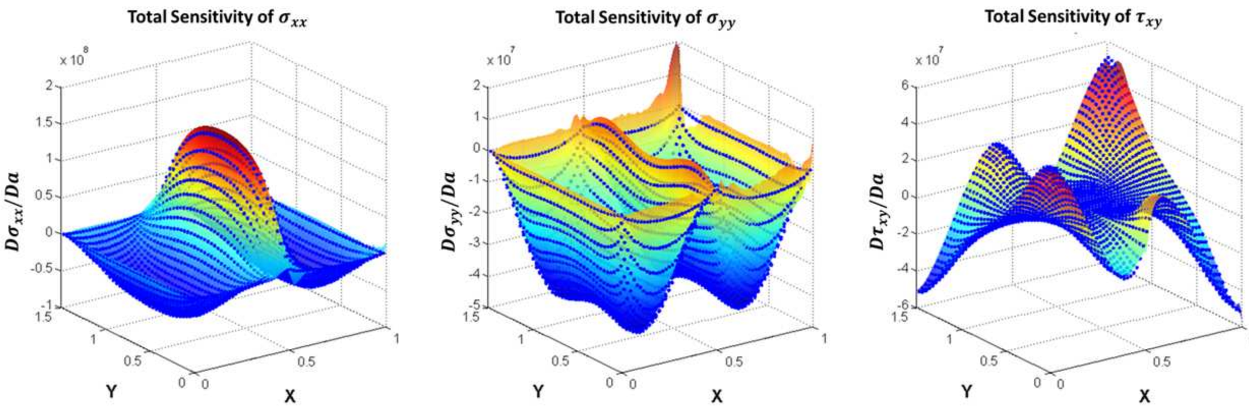
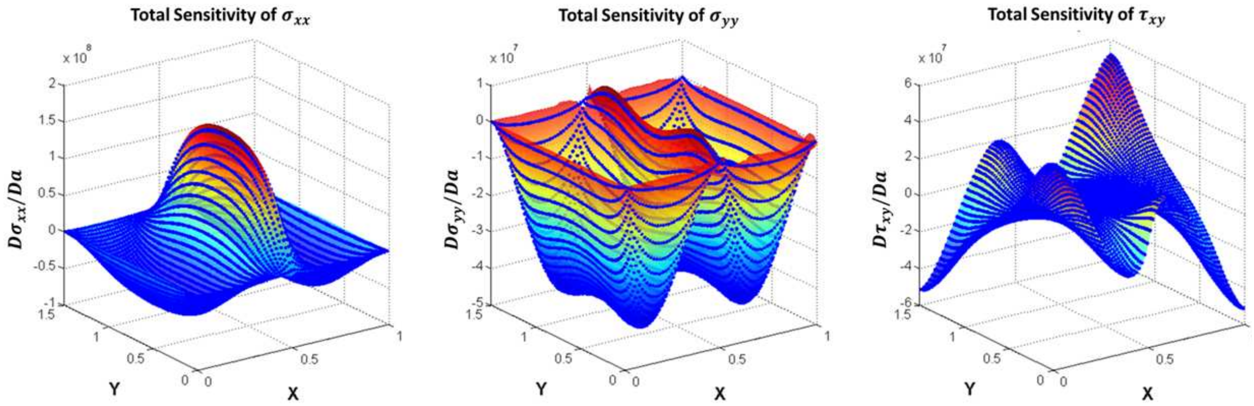


Figure 3.23: Total design derivatives of stress for a 60-by-40 mesh of the beam-stiffened plate (surface: local CSA with SGR, dots: finite difference)


Total Sensitivity of σ_{xx}

Total Sensitivity of σ_{yy}

Total Sensitivity of τ_{xy}

Figure 3.24: Total design derivatives of stress for a 70-by-70 mesh of the beam-stiffened plate with Adjusted SGR (surface: local CSA with SGR, dots: finite difference)

Table 3.7: Absolute Percent Relative Errors for Design Derivatives of Stiffened Plate Stresses

	$\dot{\sigma}_{xx}$	$\dot{\sigma}_{yy}$	$\dot{\tau}_{xy}$
$\epsilon_{60\times40}$	6.69	40.64	4.94
$\epsilon_{70\times70}$	2.89	6.43	1.71

3.3 Discussion

3.3.1 Development of a General Local Continuum Shape Sensitivity Method

A general local continuum shape sensitivity method was developed. Spatial gradient reconstruction (SGR) was used to approximate certain derivative terms that appear in the local CSE boundary conditions. The SGR only depends on analysis output, which enables a general formulation of all local CSE boundary conditions. In effect, the method can be implemented for a wide variety of analysis types and element formulations without any changes being made to the algorithm. This allows the method to treat the analysis largely as a “black box,” much as finite difference methods do. For example, a single algorithm can handle Euler-Bernoulli beam theory, Timoshenko beam theory, Kirchhoff plate theory, Reissner-Mindlin plate theory, bilinear or biquadratic elements, or even mixed element types. The only requirement is that the analysis tool be able to provide displacement and element force data. If the local CSE system is being solved externally, as it was for the plate problems in this paper, then the analysis tool must also provide the system matrices (e.g., stiffness matrix). Other analytic sensitivity methods cannot be implemented in such a general fashion. They require separate derivations for different element theories. The semi-analytic method to date is the only other approach that provides this generality.

The semi-analytic method and the local CSA method with SGR are comparable, in that they achieve, through approximation, general formulations of the discrete analytic (total continuum) and the local continuum sensitivity equations, respectively. However, the semi-analytic method can be highly innacurate for computing structural shape design derivatives (Haftka, R.T. and Adelman, H.M., 1989; Haftka, R. and Gurdal, Z., 1992). The user's control over the accuracy of the semi-analytic method is by choosing the mesh size, the finite difference step size, and the order of the finite difference approximation. Often, there is not a suitable finite difference step size and the accuracy worsens with mesh refinement. Higher-order finite difference approximations become computationally expensive, because they call for further analyses for additional finite difference perturbations. In contrast, the examples presented here illustrate that the local CSA method with SGR computes structural shape design derivatives with a high degree of accuracy and no additional finite element analyses. Furthermore, the user has several parameters by which to control the tradeoff between accuracy and efficiency, namely, mesh size, number of patch layers, and order of the Taylor series expansions.

3.3.2 Observations Regarding Spatial Gradient Reconstruction (SGR)

SGR was shown to be more robust at approximating derivative terms than the finite difference method. With the finite difference method, the user is at the mercy of the step size. With a step size too large, truncation errors begin to dominate, and with a step size to small, round off errors begin to dominate. However, the accuracy of SGR converges monotonically with increasing node density of the patches and increasing order of the Taylor series expansions. It is the recommendation of the author that when employing SGR, the number of data points included within a patch should be between one and three times the number of unknown coefficients in the Taylor series expansion. Also, it is critical that the patch definitions never cross over structural interfaces. Each structural interface has to act as a boundary, and the SGR must be executed within all boundaries.

3.3.3 Present Capabilities and Future Work

The local CSA method with SGR was first demonstrated for linear static bending of a cantilever beam. Comparing the results to those of other common sensitivity methods clearly demonstrated that this method is accurate and effective. The method was also implemented for a linear aeroelastic transient gust response of a 2-D structure. The design derivative results showed strong agreement with finite difference results. Nastran's linear transient analysis was used, and the OpenFSI toolbox allowed for typical section aerodynamics to be defined for the gust problem. The method was also implemented for linear static bending of a simply-supported rectangular plate, a rectangular plate with mixed boundary conditions, and a beam-stiffened rectangular plate using Nastran. The design derivatives of the plate responses showed strong agreement with analytic and finite difference solutions. The 2-D beam model and the beam-stiffened plate model demonstrated that the method is viable for models with structural interfaces between joined structural components. Lastly, a general approach for computing design derivatives of stress from design derivatives of displacement was presented.

Future work should implement the local CSA method with SGR for higher fidelity models. This could include non-rectangular plates, plates with holes, composite plates, or multiple structural interfaces. Furthermore, higher fidelity aerodynamics, such as 2D potential flow or Euler's equations, should be used for the aeroelastic gust response problem. Each of these would help take steps towards developing a general purpose design sensitivity code that can be nonintrusively implemented with black box tools. The results in the next chapter help take another critical step, which is the application of the method to nonlinear analysis.

Chapter 4

Local Continuum Sensitivity Analysis with Spatial Gradient Reconstruction: Nonlinear Applications

This chapter extends local CSA with SGR to nonlinear applications. Section 4.1 presents a slightly different formulation that is required for nonlinear systems. The remaining sections of this chapter present the implementation of local CSA with SGR to compute design derivatives of nonlinear systems. First, nonlinear static analyses of a cantilevered beam are conducted using Euler-Bernoulli and Timoshenko beam theories. Second, the transient gust analysis of the 2-D beam model presented in Section 3.1.2 is revisited with the inclusion of geometric nonlinearity. Lastly, the rectangular plate model with mixed boundary conditions from Section 3.2.2 and the beam-stiffened plate model from Section 3.2.3 are also revisited with the inclusion of geometric nonlinearity.

4.1 Local CSA Formulation for Nonlinear Analysis

The formulation for linear problems presented in Chapter 2 is also documented in Cross, D. and Canfield, R.A. (2013). A convenience of linear problems is that the local CSEs maintain the same left-hand side differential operators as the governing equations. This is not the case for nonlinear problems. By factoring out a $\boldsymbol{u}$ from Eqs. (2.1) and (2.2), the governing equations of the nonlinear boundary value problem can be rewritten as

$$\boldsymbol{A}(\boldsymbol{u},t;\boldsymbol{b}) = \boldsymbol{A}_{NL}(\boldsymbol{u},t)\boldsymbol{u} = \boldsymbol{f}(\boldsymbol{x},t;\boldsymbol{b}) \qquad \text{on} \qquad \Omega \tag{4.1}$$

$$\boldsymbol{B}(\boldsymbol{u},t;\boldsymbol{b}) = \boldsymbol{B}_{NL}(\boldsymbol{u},t)\boldsymbol{u} = \boldsymbol{g}(\boldsymbol{x},t;\boldsymbol{b}) \qquad \text{on} \qquad \Gamma \tag{4.2}$$

where $\boldsymbol{A}_{NL}$ and $\boldsymbol{B}_{NL}$ are nonlinear differential operators that are functions of $\boldsymbol{u}$. The static version of the linearized system can be discretized using the Galerkin finite element method to give

$$[K\left(\{u\}^{(r-1)}\right)]\{u\}^{(r)} = \{F\} \tag{4.3}$$

where u^r denotes the solution at the r-th iteration of the iterative nonlinear solver. Following the derivation in Cross, D. and Canfield, R.A. (2013), partial differentiation of Eq. (4.1) and total differentiation of Eq. (4.2) yields

$$\left(\boldsymbol{A}_{NL}(\boldsymbol{u},t) + \frac{\partial \boldsymbol{A}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u},t)\right)\boldsymbol{u}' = \boldsymbol{f}'(\boldsymbol{x},t;\boldsymbol{b}) - \boldsymbol{A}'_{NL}(\boldsymbol{u},t)\boldsymbol{u} \qquad \text{on} \qquad \Omega \tag{4.4}$$

$$\left(\dot{\overline{\boldsymbol{B}_{NL}(\boldsymbol{u},t)}}\right)\boldsymbol{u} + \left(\boldsymbol{B}_{NL}(\boldsymbol{u},t)\right)\dot{\boldsymbol{u}} = \dot{\boldsymbol{g}}(\boldsymbol{x},t;\boldsymbol{b}) \qquad \text{on} \qquad \Gamma \tag{4.5}$$

Eq. (4.5) can be expressed as the local CSE boundary condition by expanding the total design derivatives on the left hand side according to Eq. (2.4) and rearranging terms.

$$\left(\boldsymbol{B}_{NL}(\boldsymbol{u},t)+\frac{\partial \boldsymbol{B}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u},t)\right)\boldsymbol{u}'=\dot{\boldsymbol{g}}(\boldsymbol{x},t;\boldsymbol{b})-\boldsymbol{B}'_{NL}(\boldsymbol{u},t)\boldsymbol{u}-\nabla_{\boldsymbol{x}}\left(\boldsymbol{B}_{NL}(\boldsymbol{u},t)\boldsymbol{u}\right)\cdot\boldsymbol{\mathcal{V}} \qquad \text{on} \qquad \Gamma \tag{4.6}$$

Eqs. (4.4) and (4.6) are the local CSEs and their sensitivity boundary conditions, respectively. If non-differentiable concentrated loads are present, then they must be defined as domain interfaces with appropriate interface boundary conditions. The governing equations over each domain and all associated boundary conditions can then be differentiated according to Eqs. (4.4) and (4.6).

Borggaard, J. and Burns, J. (1997) and Liu, S. and Canfield, R.A. (2013b) showed that, if the same discretization used for the analysis is used to discretize the local CSEs, then the static version of the discretized local CSEs is

$$\left[T\left(\{u\}\right)\right]\{u'\}=\{F_{Local}\} \tag{4.7}$$

where $\left[T\left(\{u\}\right)\right]$ is the tangent stiffness matrix of a Newton-Raphson solver and $\{F_{Local}\}$ is the reduced vector of local CSE boundary conditions. The tangent stiffness matrix is defined in relation to the nonlinear stiffness matrix as

$$T_{ij}(u)=K_{ij}(u)+\sum_{m=1}^{n}\frac{\partial K_{im}}{\partial u_j}u_m \tag{4.8}$$

Notice the similarity between Eq. (4.8) and the left-hand sides of Eqs. (4.4) and (4.6). The only differences between the local CSEs for linear problems (Chapter 2 and Cross, D. and Canfield, R.A. (2013)) and the local CSEs presented here for nonlinear problems are the definitions of the differential operators. The next subsection discusses how this effects the implementation of the local continuum method.

4.1.1 Implementation and Solution of the Local Continuum Sensitivity Equations

For linear problems, the analysis tool used to solve the original system can also be used to solve the local CSEs, because the corresponding local CSEs are governed by the same differential operators that appear in the governing equations. After conducting the original analysis, the output can be used to formulate the boundary conditions of the local CSEs. These can then be applied to a second analysis which yields the local design derivative solution. Another option is to solve the local CSEs externally, which requires that the analysis tool provide the system matrices used to solve the original analysis.

It is not as straight-forward for nonlinear problems, because the differential operators of the local CSEs are not equivalent to those of the governing equations. Therefore, the primary option is to solve the local CSEs for nonlinear problems externally by having the analysis tool provide the converged tangent matrices, such as the one in Eq. (4.7).

For the essential and non-essential boundary conditions it can be stated that

$$\boldsymbol{B}^{(e)}_{_{NL}}\boldsymbol{u}_{_{NL}} = \dot{\boldsymbol{g}}_{\Gamma_e}(\boldsymbol{x}, t; \boldsymbol{b}) \qquad \text{on} \qquad \Gamma_e, \tag{4.9}$$

and

$$\boldsymbol{B}^{(n)}_{_{NL}}\boldsymbol{u}_{_{NL}} = \dot{\boldsymbol{g}}_{\Gamma_n}(\boldsymbol{x}, t; \boldsymbol{b}) \qquad \text{on} \qquad \Gamma_n \tag{4.10}$$

where $\boldsymbol{B}^{(e)}_{_{NL}}$ and $\boldsymbol{B}^{(n)}_{_{NL}}$ are the essential and non-essential nonlinear boundary condition operators, respectively, and $\boldsymbol{u}_{_{NL}}$ and $\boldsymbol{Q}_{_{NL}}$ are the primary variable (e.g. displacement) and secondary variable (e.g. element force) responses of the nonlinear analysis, respectively. Assuming that $\boldsymbol{B}'_{_{NL}}$ vanishes from Eq. (4.6) (discussed in Cross, D. and Canfield, R.A. (2013)), it follows that the local CSE boundary conditions for nonlinear problems, can be rewritten as

$$\left(\boldsymbol{B}^{(e)}_{NL}(\boldsymbol{u}_{_{NL}}) + \frac{\partial \boldsymbol{B}^{(e)}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u}_{_{NL}})\right)\boldsymbol{u}'_{_{NL}} = \boldsymbol{u}'_{_{NL}} = \dot{\boldsymbol{g}}_{\Gamma_e}(\boldsymbol{x}, t; \boldsymbol{b}) - \nabla_{\boldsymbol{x}}\boldsymbol{u}_{_{NL}} \cdot \boldsymbol{\mathcal{V}} \qquad \text{on} \qquad \Gamma_e \tag{4.11}$$

$$\left(\boldsymbol{B}^{(n)}_{NL}(\boldsymbol{u}_{_{NL}}) + \frac{\partial \boldsymbol{B}^{(n)}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u}_{_{NL}})\right)\boldsymbol{u}'_{_{NL}} = \boldsymbol{Q}'_{_{NL}} = \dot{\boldsymbol{g}}_{\Gamma_n}(\boldsymbol{x}, t; \boldsymbol{b}) - \nabla_{\boldsymbol{x}}\boldsymbol{Q}_{_{NL}} \cdot \boldsymbol{\mathcal{V}} \qquad \text{on} \qquad \Gamma_n \tag{4.12}$$

In order to solve the local CSEs, the system matrices of the original analysis are used as the sensitivity coefficient matrices of the local CSEs. The $\dot{\boldsymbol{g}}$ terms in boundary condition equations (4.11) and (4.12) can be formulated directly from the enforced displacements and applied loads. The $\boldsymbol{\mathcal{V}}$ term in boundary condition equations (4.11) and (4.12) is the design velocity, which can be formulated directly from the geometric parameterization. Lastly, the spatial derivatives that appear in boundary condition equations (4.11) and (4.12) can be approximated using spatial gradient reconstruction (SGR). The details of implementing SGR, which originated from work previously done by Duvigneau, R. and Pelletier, D. (2006) and Zienkiewicz, O.C. and Zhu, J.Z. (1992), are provided in Cross, D. and Canfield, R.A. (2013). In short, Duvigneau, R. and Pelletier, D. (2006) used this approach to recover high-order derivatives of the primary variables, but here, Eqs. (4.11) and (4.12) only require first-order derivatives of both primary and secondary variables. The first-order derivatives of a state or response variable at a particular finite element node are approximated through a least-squares match of the response data and a Taylor series approximation of the response data expanded around the node of interest. The response data that is included in the least-squares match is determined by the definition of a patch. A patch is a local region comprised of nearby finite elements. Qualitatively speaking, the accuracy of SGR is improved by increasing the nodal density of the patches and by increasing the order of the Taylor series expansions. For subsequent examples, illustrations of patches and the corresponding least-squares match are shown.

As discussed in Cross, D. and Canfield, R.A. (2013) conventional analytic sensitivity methods are intrusive, meaning that formulation of the boundary conditions requires "intimate knowledge" of the source code and numerical formulation (Haftka, R.T. and Adelman, H.M., 1989). However, this particular method can be implemented non-intrusively, if the following data can be provided by the analysis tool:

1. linear system matrices and converged tangent system matrices
2. primary variable response data (e.g. displacements)
3. secondary variable response data (e.g. element forces)

If these data can be provided by the analysis tool as output, then the boundary conditions can be formulated non-intrusively. Then, the local CSEs can be solved, and the local design derivative solution can be transformed into a total design derivative solution via Eq. (2.4). Conventional implementation of local CSA (Liu, S. et al., 2010) uses only primary variable response data, which results in an intrusive method, because the definition of the boundary condition operator, $\boldsymbol{B}_{NL}$, is required. Furthermore, it results in local CSE boundary conditions that depend on high-order derivatives of the primary variables. Approximation of these high-order derivatives can introduce numerical error into the design derivative solution. As previously mentioned, this is one reason that local CSA has largely been ignored by the structures community. Using the secondary variable response data provides two unique advantages. First, a non-intrusive implementation is achieved, and second, only first-order derivatives need to be approximated. Many analysis tools use techniques to improve the accuracy of the secondary variable response data (MSC Software Corporation, 2010b), which could result in a more accurate boundary condition formulation.

Furthermore, because conventional methods are intrusive, separate derivations of the local CSEs are required for different models. In contrast, the local CSA with SGR method presented here is non-intrusive and element agnostic. Therefore, if the local CSA with SGR algorithm takes the system matrices, primary variable response data, and secondary variable response data as input, then information redarding the analysis source of the input is not required by the algorithm. For example, the same algorithm can be used for DSA of linear and nonlinear problems; problems that use Euler-Bernoulli or Timoshenko beam theories; or problems solved using different orders of nodal based shape functions. This advantageous characteristic is demonstrated in the remaining sections with several examples.

4.2 Euler-Bernoulli vs. Timoshenko Beam Theory

This section presents design derivative results for a short, thick cantilevered beam that is modeled in Nastran (SOL 400) using both Euler-Bernoulli and Timoshenko beam theories (MSC Software Corporation, 2010b). The beam has Young's modulus, $E = 70GPa$, a shear modulus, $G = 26GPa$, a cross-sectional area, $A = 0.0015m^2$, a area moment of inertia, $I = 4.3 \times 10^{-5}m^4$, and a shear correction factor, $k_s = 5/6$. The beam, which has a length, $L = 0.5m$, is subjected to an end load, $P = 1MN$. Geometric nonlinearity is included in the analyses. The design derivative of the displacement response is calculated with respect to the length of the beam, L. The design velocity vector, with unit normal coordinate directions

$\hat{x}$, $\hat{y}$, and $\hat{z}$, is defined as

$$\boldsymbol{\mathcal{V}} = \frac{x}{L}\hat{x} + 0\hat{y} + 0\hat{z} \tag{4.13}$$

The sensitivity system is solved in the global reference frame. The global displacement degrees of freedom are $\boldsymbol{u} = \{u, w, \psi\}$, where u is horizontal displacement, w is vertical displacement, and ψ is rotation. The forces are $\boldsymbol{Q} = \{N, V, M\}$, where N is the horizontal force, V is the vertical force, and M is the bending moment. Therefore, the local CSEs simplify to

$$\left(\boldsymbol{A}_{NL}(\boldsymbol{u}) + \frac{\partial \boldsymbol{A}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u})\right)\boldsymbol{u}' = 0 \qquad \text{on} \qquad \Omega \tag{4.14}$$

$$\boldsymbol{u}'(0) = -\nabla_x \boldsymbol{u}(0) \cdot \boldsymbol{\mathcal{V}}(0) = \boldsymbol{0} \tag{4.15}$$

$$\boldsymbol{Q}'(L) = -\nabla_x \boldsymbol{Q}(L) \cdot \boldsymbol{\mathcal{V}}(L) = -\{N_{,x}(L), V_{,x}(L), M_{,x}(L)\}^T \tag{4.16}$$

SGR was used to approximate the spatial derivatives that appear in Eq. (4.16) and that appear in the convective term used to transform the local design derivative solution into a total design derivative solution. Five-layer patches and fourth-order Taylor series expansions were used to conduct SGR. Figure 4.1 illustrates how a five-layer patch and fourth-order Taylor series expansion are used to approximate the first-order spatial derivative of the axial force at the end of the beam.

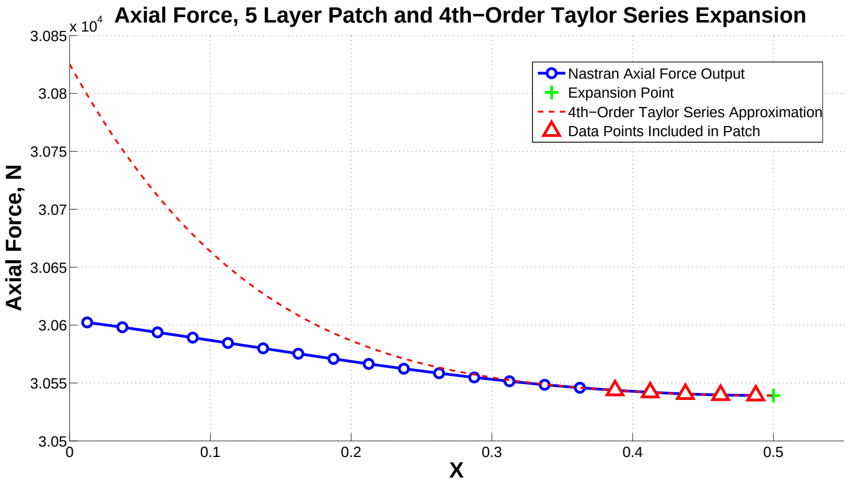


Figure 4.1: SGR of axial force employed with a five-layer patch and fourth-order Taylor series expansion

Both Euler-Bernoulli and Timoshenko beam theory results are shown in Figure 4.2, with vertical displacement on the left, and the total design derivative of vertical displacement on the right. Again, the total design derivatives were calculated with respect to the length of the beam, L, and are compared to finite difference results with a relative step size of 10^{-3}. The displacement results indicate that the Euler-Bernoulli beam behaves more stiffly, because shear effects are neglected. As a result its design derivative solution is less sensitive to the length of the beam. The absolute percent relative difference between the local continuum results and finite difference results were calculated using Eq. (4.17) and are tabulated in Table 4.1.

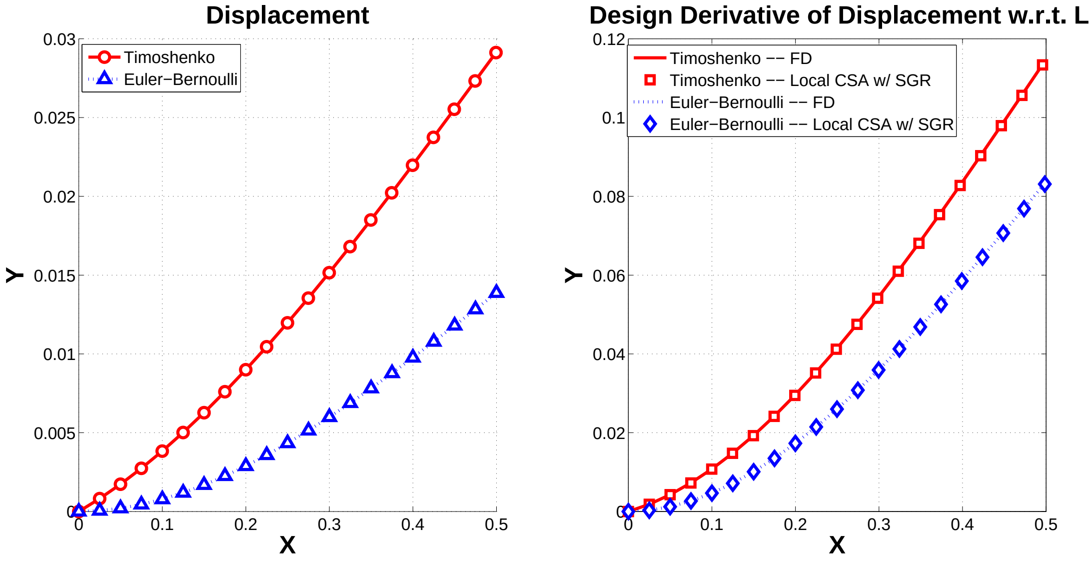


Figure 4.2: Vertical displacement (left) and design derivative of vertical displacement w.r.t. L (right) for a cantilevered beam

$$\epsilon_\phi = \frac{\|\phi_{LC} - \phi_{FD}\|_\infty}{\|\phi_{FD}\|_\infty} \times 100 \tag{4.17}$$

Here, ϕ is a general design derivative variable, and the subscripts LC and FD indicate coming from the local continuum and finite difference solution, respesctively.

Table 4.1: Absolute Percent Relative Difference of Design Derivative Results

ϵ	$\dot{u}$	$\dot{w}$	$\dot{\psi}$
Timoshenko	0.142	0.081	0.030
Euler-Bernoulli	0.234	0.109	0.029

The same algorithm was used for the Euler-Bernoulli and Timoshenko design derivative calculations. Although, different nonlinear operators are used for the analyses, the same type of data feeds into the local CSA with SGR algorithm. Conventional analytic sensitivity methods would require different calculations, based on the specific definitions of the nonlinear operators. For this reason, the non-intrusive nature of this method is advantageous.

Furthermore, so long as accurate derivative approximations can be made using SGR, then one can expect similarly accurate design derivative solutions. (Cross, D. and Canfield, R.A., 2013)

4.3 Nonlinear Transient Gust Response of 2-D Beam Model

4.3.1 Model Information

The joined-beam model presented in Section 3.1.2 and Cross, D. and Canfield, R.A. (2013) is again presented here, but is now modeled with the inclusion of geometric nonlinearity. The objective is to compute the design derivatives of the transient gust response with respect to the length of the cantilevered beam, L. In this section the design derivative solutions for the linear and nonlinear aeroelastic gust responses are compared, and the effects of geomtric nonlinearity are discussed.

Previously, the linear aeroelastic gust response was calculated using Nastran's OpenFSI interface. This allows Nastran's SOL 400 transient solver to communicate with an externally defined C++ code, where the quasi-steady aerodynamics load is defined (MSC Software Corporation, 2010b). The local CSEs were then solved outside of Nastran using a linear transient solver. This required Nastran to provide the mass matrix, the linear stiffness matrix (global and elemental), and the time histories of displacements and element forces. Unfortunately, for the nonlinear transient gust problem (SOL 400, LGDISP=1), Nastran does not provide element forces as output (MSC Software Corporation, 2010b). Therefore, Nastran does not meet the previously defined requirements to conduct local CSA with SGR.

Therefore, a nonlinear transient aeroelastic analysis tool, provided by Liu, S. et al. (2010), was used to compute the nonlinear gust response and design derivative results shown here. This tool was treated entirely as a black box, but was chosen because it could provide the necessary output data required to perform local CSA with SGR. The model parameters for both the structure and typical section aerodynamics that were used for the linear aeroelastic gust response were used again here.

4.3.2 Design Derivative Results

The design velocity takes the same definition as the linear example, Eq. (3.13), and the system has the same degrees of freedom, $\boldsymbol{u} = \{u, w, \psi\}$ and $\boldsymbol{Q} = \{N, V, M\}$. The local CSEs also take a similar form, but with the nonlinear operators.

$$\left(\boldsymbol{A}_{NL}(\boldsymbol{u}, t) + \frac{\partial \boldsymbol{A}_{NL}}{\partial \boldsymbol{u}}(\boldsymbol{u}, t)\right)\boldsymbol{u}' = 0 \qquad \text{on} \qquad \Omega \tag{4.18}$$

$$\boldsymbol{u}'(0, t) = -\nabla_{\boldsymbol{x}}\boldsymbol{u}(0, t) \cdot \boldsymbol{\mathcal{V}}(0) = \boldsymbol{0} \tag{4.19}$$

$$\begin{aligned}\boldsymbol{Q}'(L, t) &= \dot{\boldsymbol{g}}(L, t) - \nabla_x \boldsymbol{Q}(L, t) \cdot \boldsymbol{\mathcal{V}}(L) \\ &= \{N_{,x}(L, t), \dot{F}_f(t) - V_{,x}(L, t), M_{,x}(L, t)\}^T\end{aligned} \tag{4.20}$$

where the total design derivative of the aeroelastic load, $\dot{F}_f$ is defined as

$$\begin{aligned}\dot{F}_f\,(t) &= q_\infty s C_{l_\alpha}\left(\dot{\theta}_{tip} + \frac{1}{U_\infty}\dot{h}_{,t}\right) \\ &= q_\infty s C_{l_\alpha}\left(\psi' + \psi_{,x} + \frac{1}{U_\infty}\left(-w_{,t}' - w_{,tx}\right)\right)\Bigg|_{x=L}\end{aligned} \tag{4.21}$$

Furthermore, strain discontinuities at the joint of the main beam and the strut require the following boundary conditions to be enforced. Details concerning these boundary conditions can be found in Liu, S. et al. (2010).

$$\boldsymbol{u}'(x_j^+) = \boldsymbol{u}'(x_j^-) - \left(\boldsymbol{u}_{,x}(x_j^+) - \boldsymbol{u}_{,x}(x_j^-)\right)\boldsymbol{\mathcal{V}}(x_j) \tag{4.22}$$

$$\boldsymbol{Q}'(x_j) = -\sum \boldsymbol{Q}_{,x}(x_j)\boldsymbol{\mathcal{V}}(x_j) \tag{4.23}$$

where x_j is the joint location and is equal to ϵL and $\epsilon = 1/2$. A $()^+$ indicates the evlaution is from the right, and a $()^-$ indicates the evaluation is from the left. These boundary conditions require that SGR be used to reconstruct spatial derivatives of $\boldsymbol{u}$ from both the left and right sides of the joint and $\boldsymbol{Q}$ from each of the three structural elements adjacent to the joint. When conducting SGR to approximate these terms, it is essential that the patches are adjacent to the joint, but do not cross over it. This is necessary to capture any discontinuities that may be present.

A more detailed derivation of the local CSEs is included in Cross, D. and Canfield, R.A. (2013) and also appear in Section 3.1.2. The results are shown in Figure 4.3 with vertical displacement on the left and the design derivative of vertical displacement on the right. Again, design derivatives were calculated with respect to the length of the beam, L, and are validated by comparison to finite difference results (10^{-3} was the converged step size). In addition to the results for the nonlinear gust response, the results for the linear gust response presented in Cross, D. and Canfield, R.A. (2013) are also shown. Four-layer patches and third-order Taylor series expansions were used to conduct SGR. Furthermore, the same linear transient solver was used to solve the local CSEs for both the linear and nonlinear gust problems.

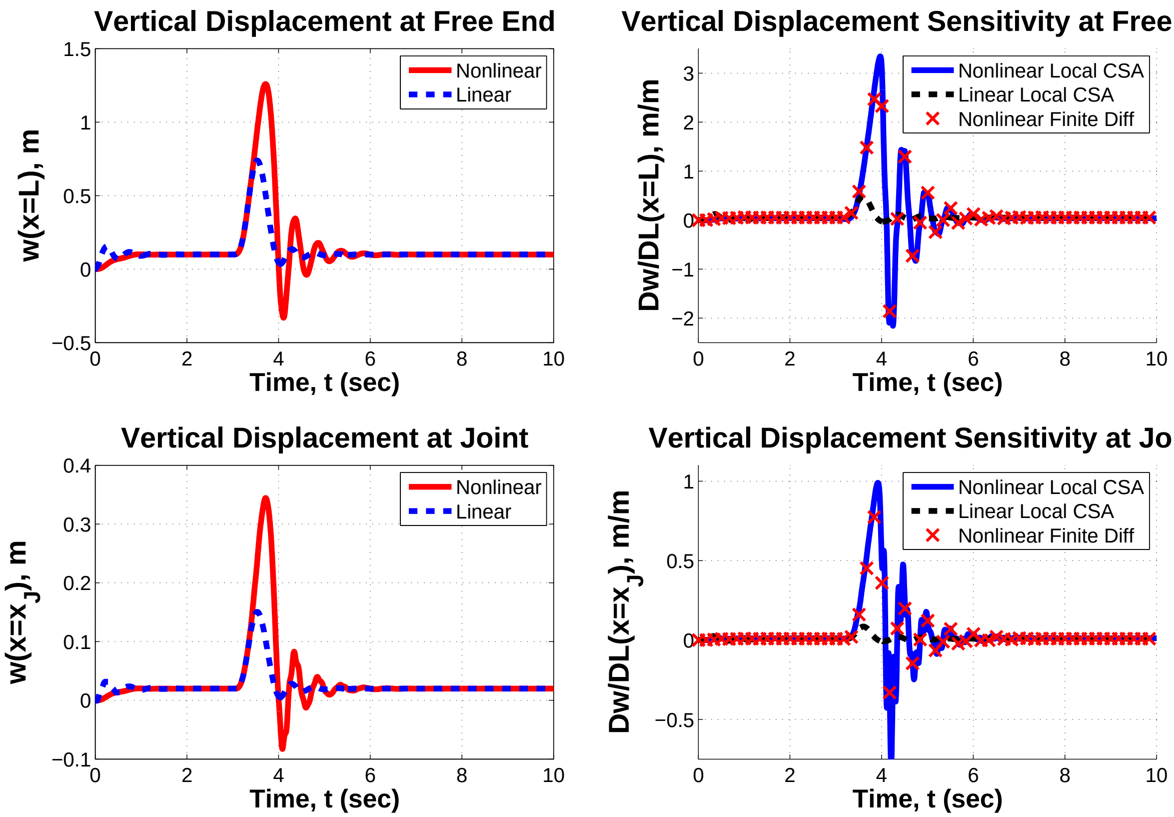


Figure 4.3: Joined-beam aeroelastic gust response. (a) Vertical displacement at free end (b) Sensitivity of vertical displacement at free end w.r.t. length, L (c) Vertical displacement at joint (d) Sensitivity of vertical displacement at joint w.r.t. length, L

The relative difference between the local CSA and finite difference solutions is calculated for the time integrated design derivative of nonlinear displacement at both the end of the beam

$$\frac{\int_0^t \dot{w}_{LC}(L,t)dt - \int_0^t \dot{w}_{FD}(L,t)dt}{\int_0^t \dot{w}_{FD}(L,t)dt} = 0.0198 \tag{4.24}$$

and the joint location.

$$\frac{\int_0^t \dot{w}_{LC}(x_J,t)dt - \int_0^t \dot{w}_{FD}(x_J,t)dt}{\int_0^t \dot{w}_{FD}(x_J,t)dt} = 0.0578 \tag{4.25}$$

The finite element model consisted of 40 beam elements, and the time integration consisted of 1500 time steps. When investigated this 40 element mesh under static loading for linear analysis, it was determined that four-layer patches and third-order Taylor series produced the most accurate results. Any more layers pollutes the SGR calculations with truncation error, becuase the data points are relatively far from the expansion point. Therefore, four-layer patches and third-order Taylor series were assumed to be the best SGR parameters for local CSA of the linear and nonlinear transient gust responses. A finite element mesh with more degrees of freedom would certainly improve the accuracy but at an additional cost. Furthermore, the linear and nonlinear analyses are sufficiently converged using 1500 time steps. However, a finer temporal mesh of 2000 time steps reduced the relative differences reported in Eqs. (4.24) and (4.25) to 0.0132 and 0.036 respectively.

The linear and nonlinear responses are critically different. During the gust response, the bracing member (strut) exhibits dramatic geometric nonlinearity, which minimizes its load carrying capability. This softening effect results in very large displacements and rotations, which are illustrated in Figure 4.4. The maximum vertical displacement that occurs during the nonlinear response is approximately 70 percent larger than that of the linear response. This is the same type of behavior that was determined to be a critical design condition for HALE aircraft such as Helios (NASA, 2004) and Sensorcraft (Johnson, 2001). Furthermore, the design derivative results indicate that the nonlinear repsonse is significantly more sensitive to the geometry of the joined beam configuration than the linear response. The maximum total design derivative of vertical displacement that occurs during the nonlinear response is approximately 560 percent larger than that of the linear response. This is further evidence that an accurate and efficient design derivative capability is necessary for such nonlinear models.

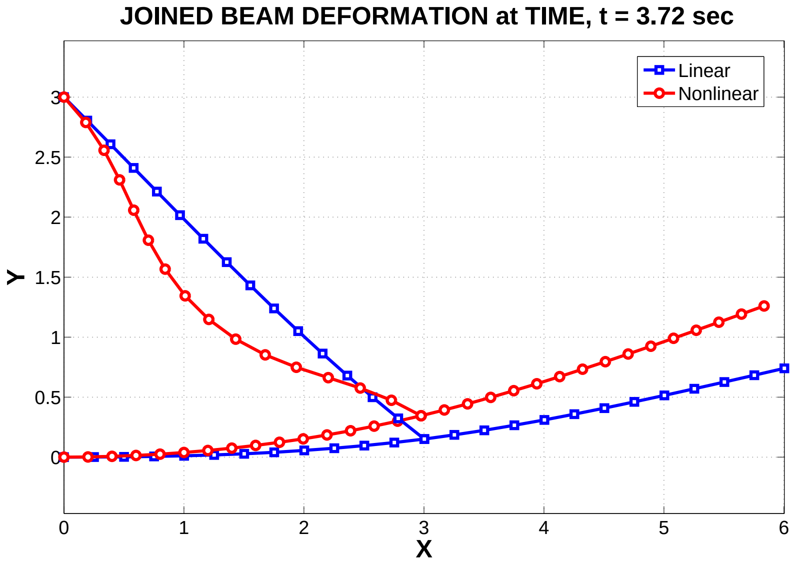


Figure 4.4: Joined-beam deflection during the gust response at t = 3.72 seconds (blue: linear response, red: nonlinear response)

4.4 Nonlinear Static Plate Bending

4.4.1 Rectangular Plate with Mixed Boundary Conditions

The plate model from Section 3.2.2, which has two clamped edges and two simply-supported edges, is modeled in Nastran with geometric nonlinearity included. The in-plane degrees of freedom are coupled with the out-of-plane degrees of freedom for the nonlinear problem. The plate lies in the x-y plane with x-dimension, $a = 1m$ and y-dimension, $b = 1.5m$. The magnitude of the uniformly distributed out-of-plane load is $q_0 = 100000 N/m^2$, and the plate has Young's modulus, $E = 7e10 N/m^2$; thickness, $t = 0.01m$; and Poisson's ratio, $\nu = 0.25$. The edges along $x = 0$ and $x = a$ are simply-supported, while the edges along $y = 0$ and $y = b$ are clamped. Design derivatives of the displacement response are calculated with respect to the plate dimension a. Therefore, the design velocity takes the form

$$\boldsymbol{\mathcal{V}}(x, y; a) = \boldsymbol{\mathcal{V}}_x \hat{x} + \boldsymbol{\mathcal{V}}_y \hat{y} = \frac{x}{a} \hat{x} \tag{4.26}$$

The displacement degrees of freedom are $\boldsymbol{u}(x, y) = \{u, v, w, \psi_x, \psi_y\}^T$, where u and v are in-plane displacements, w is the out-of-plane displacement, and ψ_i is the rotation about the i-th axis. For this example the rotation about the z-axis (drilling degree of freedom) is held fixed. The forces are $\boldsymbol{Q}(x, y) = \{N_x, N_y, V, M_x, M_y\}^T$, where N_i is axial force in the i-th direction, V is through-thickness shear force, and M_i is the bending moment acting on a face whose outward normal is in the i-th direction. The local CSEs are

$$\left(\boldsymbol{A}_{NL}(\boldsymbol{u}) + \frac{\partial \boldsymbol{A}_{NL}}{\partial \boldsymbol{u}} (\boldsymbol{u}) \right) \boldsymbol{u}' = 0 \qquad \text{on} \qquad \Omega \tag{4.27}$$

The local CSE boundary conditions along $x = 0$ are

$$u'(0, y) = \dot{u}(0, y) - u_{,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{4.28}$$

$$v'(0, y) = \dot{v}(0, y) - v_{,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{4.29}$$

$$w'(0, y) = \dot{w}(0, y) - w_{,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{4.30}$$

$$\psi_x'(0, y) = \dot{\psi}_x(0, y) - \psi_{x,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{4.31}$$

$$M_x'(0, y) = \dot{M}_x(0, y) - M_{x,x}(0, y)\mathcal{V}_x(0, y) = 0 \tag{4.32}$$

The local CSE boundary conditions along $x = a$ are

$$u'(a, y) = \dot{u}(a, y) - u_{,x}(a, y)\mathcal{V}_x(a, y) = -u_{,x}(a, y) \tag{4.33}$$

$$v'(a, y) = \dot{v}(a, y) - v_{,x}(a, y)\mathcal{V}_x(a, y) = -v_{,x}(a, y) \tag{4.34}$$

$$w'(a,y) = \dot{w}(a,y) - w_{,x}(a,y)\mathcal{V}_x(a,y) = -w_{,x}(a,y) \tag{4.35}$$

$$\psi_x'(a,y) = \dot{\psi}_x(a,y) - \psi_{x,x}(a,y)\mathcal{V}_x(a,y) = -\psi_{x,x}(a,y) \tag{4.36}$$

$$M_x'(a,y) = \dot{M}_x(a,y) - M_{x,x}(a,y)\mathcal{V}_x(a,y) = -M_{x,x}(a,y) \tag{4.37}$$

The local CSE boundary conditions along $y = 0$ are

$$u'(x,0) = \dot{u}(x,0) - u_{,x}(x,0)\mathcal{V}_x(x,0) = -u_{,x}(x,0)\frac{x}{a} \tag{4.38}$$

$$v'(x,0) = \dot{v}(x,0) - v_{,x}(x,0)\mathcal{V}_x(x,0) = -v_{,x}(x,0)\frac{x}{a} \tag{4.39}$$

$$w'(x,0) = \dot{w}(x,0) - w_{,x}(x,0)\mathcal{V}_x(x,0) = -w_{,x}(x,0)\frac{x}{a} \tag{4.40}$$

$$M_y'(x,0) = \dot{M}_y(x,0) - M_{y,x}(x,0)\mathcal{V}_x(x,0) = -M_{y,x}(x,0)\frac{x}{a} \tag{4.41}$$

$$\psi_y'(x,0) = \dot{\psi}_y(x,0) - \psi_{y,x}(x,0)\mathcal{V}_x(x,0) = -\psi_{y,x}(x,0)\frac{x}{a} \tag{4.42}$$

Lastly, The local CSE boundary conditions along $y = b$ are

$$u'(x,b) = \dot{u}(x,b) - u_{,x}(x,b)\mathcal{V}_x(x,b) = -u_{,x}(x,b)\frac{x}{a} \tag{4.43}$$

$$v'(x,b) = \dot{v}(x,b) - v_{,x}(x,b)\mathcal{V}_x(x,b) = -v_{,x}(x,b)\frac{x}{a} \tag{4.44}$$

$$w'(x,b) = \dot{w}(x,b) - w_{,x}(x,b)\mathcal{V}_x(x,b) = -w_{,x}(x,b)\frac{x}{a} \tag{4.45}$$

$$M_y'(x,b) = \dot{M}_y(x,b) - M_{y,x}(x,b)\mathcal{V}_x(x,b) = -M_{y,x}(x,b)\frac{x}{a} \tag{4.46}$$

$$\psi_y'(x,b) = \dot{\psi}_y(x,b) - \psi_{y,x}(x,b)\mathcal{V}_x(x,b) = -\psi_{y,x}(x,b)\frac{x}{a} \tag{4.47}$$

Nastran SOL 400 was used to conduct the nonlinear static analysis on a 40-by-40 mesh of CQUAD4 bilinear elements. SGR was used to approximate the spatial derivatives that appear in the local CSE boundary conditions. Five-layer patches and fourth-order Taylor series expansions were used to conduct the SGR. Figure 4.5 illustrates a five-layer patch and a fourth-order Taylor series reconstruction of transverse displacement used to approximate the first-order spatial derivatives of transverse displacement at a particular node.

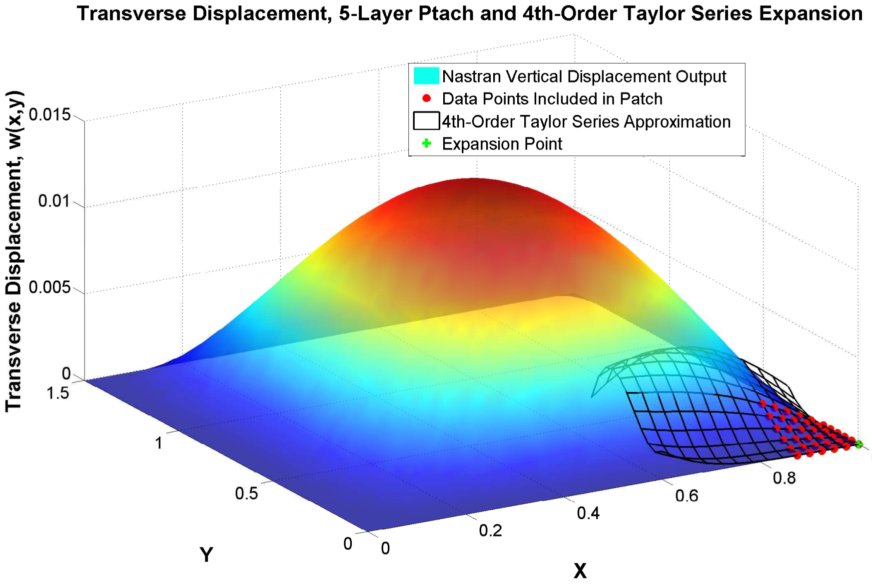


Figure 4.5: SGR of vertical displacement employed with a five-layer patch and fourth-order Taylor series expansion

Figure 4.6 shows the displacement response. The dots represent the result at the finite element nodes, the surface is an interpolation of the finite element solution, and the grey, transparent surface is the linear static solution previously presented in Cross, D. and Canfield, R.A. (2013).

The coupling between the in-plane and out-of-plane degrees of freedom result in a hardening nonlinearity, making the deformations for each of the primary out-of-plane degrees of freedom roughly 55 percent smaller than the linear result (Cross, D. and Canfield, R.A., 2013). For nonlinear static analysis, Nastran does support element force output. Therefore,

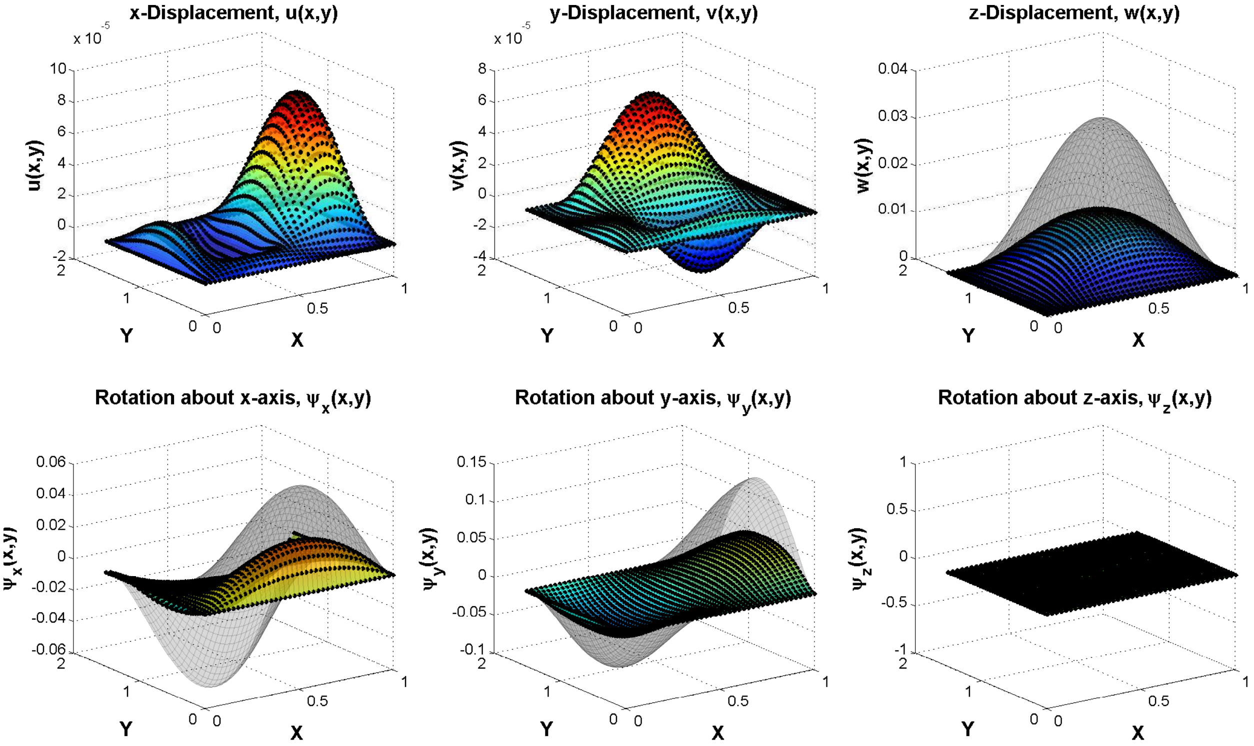


Figure 4.6: Displacements of a rectangular plate with two adjacent edges clamped and the other two adjacent edges simply-supported (grey surface: linear analytic, color surface: finite element interpolation, dots: finite element)

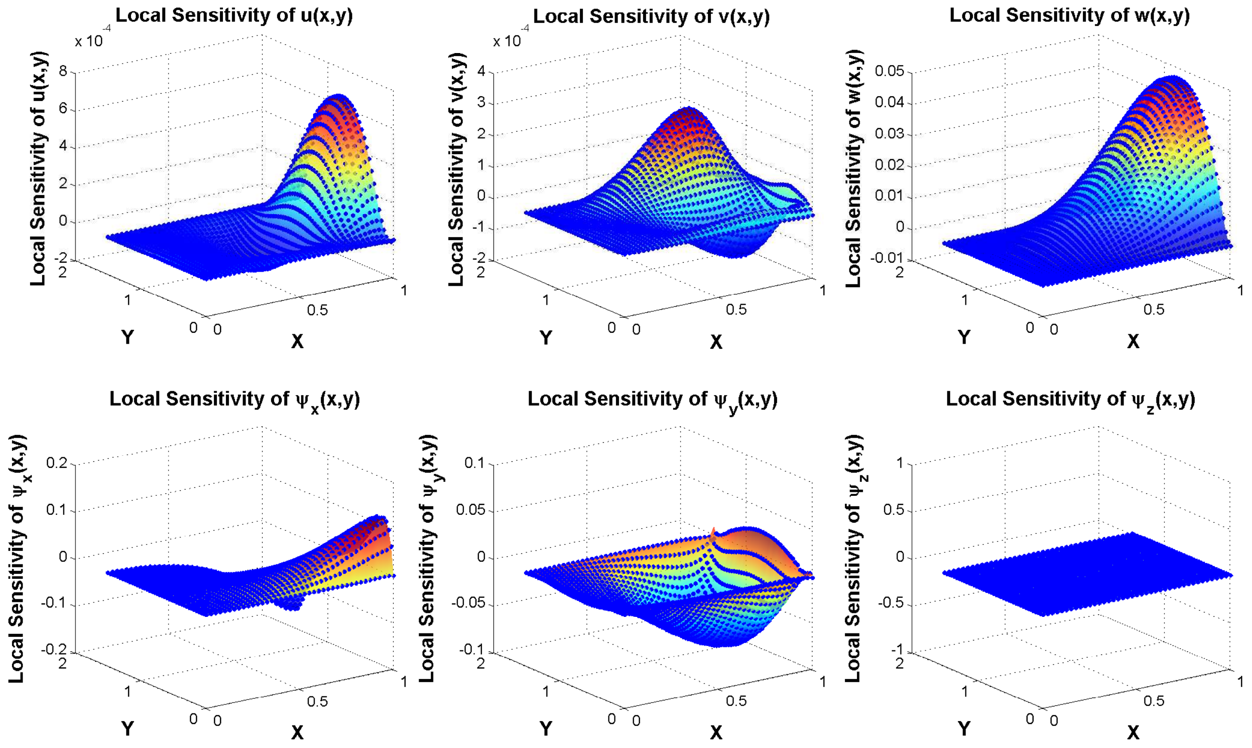


Figure 4.7: Local design derivatives of displacements with respect to plate length, a (color surface: local CSA with SGR, blue dots: finite difference)

all of the output required to conduct local CSA with SGR can be obtained from the Nastran analysis. Figures 4.7 and 4.8 show the local and total design derivative solutions with respect to the plate dimension in the x-direction, a. The color surfaces represent the local CSA solutions with SGR, and the blue dots represent the finite difference solutions using a 10^{-4} relative step size. A relative step size of 10^{-4} was determined to be optimal for linear static plate bending using Nastran analyses by a convergence study of finite difference calculations. Smaller step sizes in Nastran introduced numerical round-off error or went unrecognized due to single precision calculations. This step size was determined to be converged for nonlinear problems as well. The nonlinear plate response is less sensitive to the design parameter than the linear response. This is the expected result, given that the nonlinear response of the plate is stiffer than the linear response of the plate. The total design derivatives of

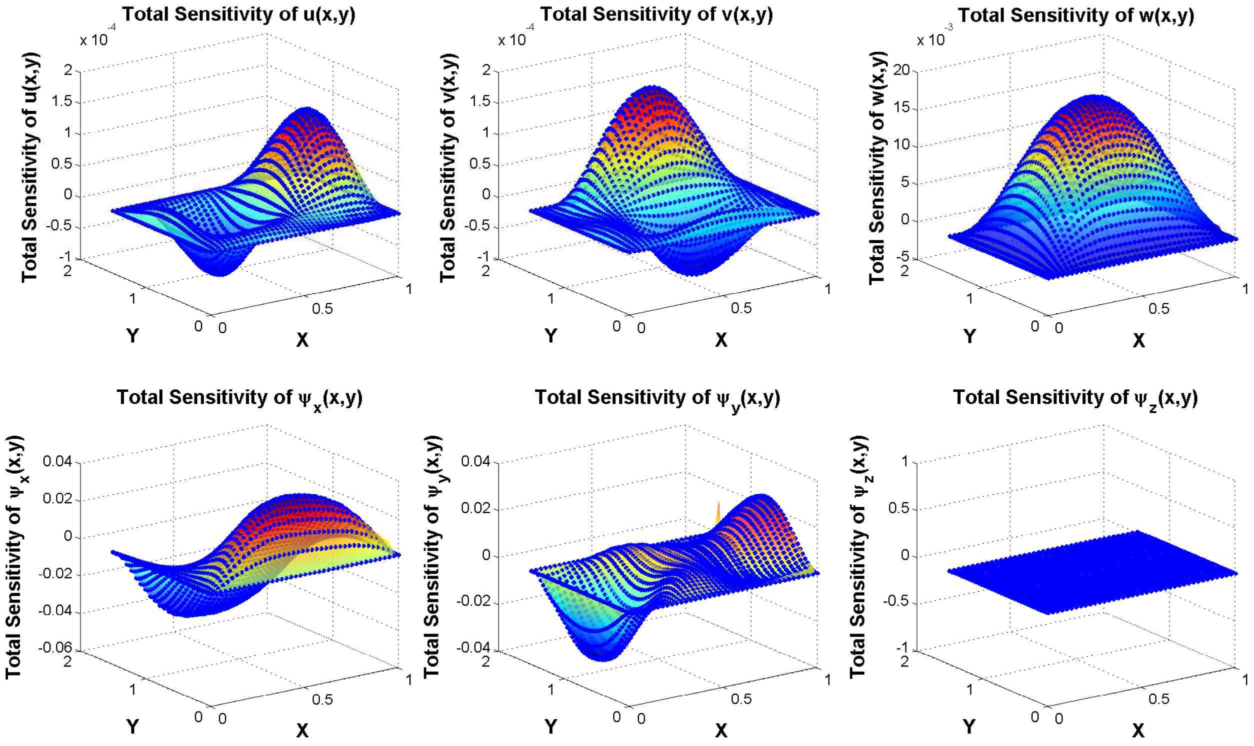


Figure 4.8: Total design derivatives of displacements with respect to plate length, a (color surface: local CSA with SGR, blue dots: finite difference)

the nonlinear response are roughly 80 percent smaller than the total design derivatives of the linear response (Cross, D. and Canfield, R.A., 2013). Quantifying the accuracy is difficult, because an analytic solution is not available for the nonlinear response. However, the absolute percent relative differences between local CSA and finite difference design derivatives were calculated using Eq. (4.17). The results of this calculation, shown in Table 4.2, demonstrate a strong agreement between the different calculations. It also suggests that local CSA with SGR can be used to non-intrusively calculate design derivatives of nonlinear plate displacements.

Table 4.2: Maximum Absolute Percent Relative Difference of Design Derivative Results

	u'	$\dot{u}$	v'	$\dot{v}$	w'	$\dot{w}$	ψ'_x	$\dot{\psi}_x$	ψ'_y	$\dot{\psi}_y$
ϵ	0.059	2.639	1.629	0.200	0.347	0.462	0.010	0.449	0.222	0.846

4.4.2 Combined Loading of a Beam-Stiffened Plate

The beam-stiffened rectangular plate presented in Section 3.2.3 is modeled again here with the inclusion of geometric nonlinearity. All of the model properties remain the same except, in addition to the uniformly distributed transverse load, $q_0 = 5 \times 10^4 N/m^2$, an uniformly distributed in-plane load is applied in the positive y-direction, $p_0 = 2 \times 10^6 N/m^2$. For the linear model only the out-of-plane degrees of freedom were considered. The combined loading for the nonlinear analysis creates significant coupling between the in-plane and out-of-plane degrees of freedom. Therefore, the displacement degrees of freedom are $\boldsymbol{u}(x,y) = \{u, v, w, \psi_x, \psi_y\}^T$, where u and v are in-plane displacements, w is the out-of-plane displacement, and ψ_i is the rotation about the i-th axis, and the forces are $\boldsymbol{Q}(x,y) = \{N_x, N_y, V, M_x, M_y\}^T$, where N_i is axial force in the i-th direction, V is through-thickness shear force, and M_i is the bending moment acting on a face whose outward normal is in the i-th direction. Figure 4.9 shows the displacement results for the beam-stiffened rectangular plate under combined loading. For the out-of-plane degrees of freedom, the nonlinear response is roughly 15 percent stiffer (smaller) than the linear response. For the in-plane degrees of freedom, the nonlinear response is roughly 15 percent softer (larger) than the linear response. Nastran was used to conduct the analysis on a 60-by-40 mesh of CQUAD4 bilinear elements.

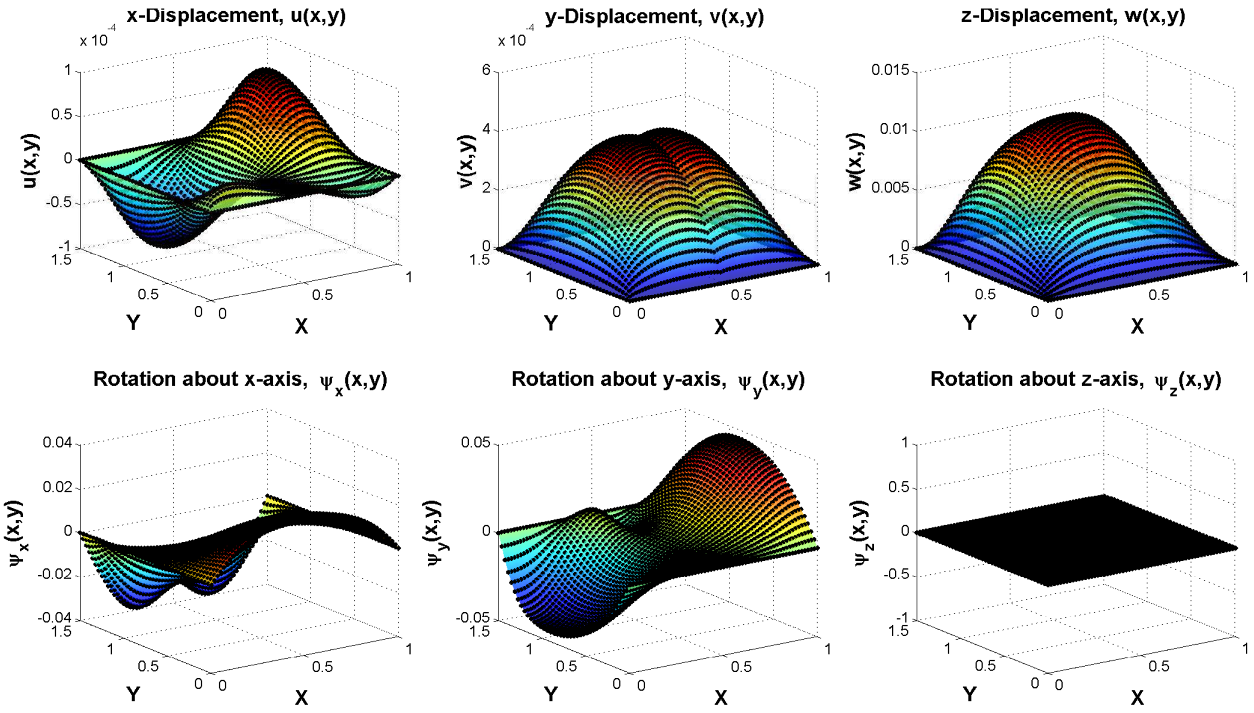


Figure 4.9: Displacements of a rectangular plate with two adjacent edges clamped and the other two adjacent edges simply-supported; combined out-of-plane and in-plane loading (surface: finite element interpolation, dots: finite element)

Local CSA with SGR was implemented to calculate the design derivatives of the plate displacements with respect to the plate dimension, a. The local CSE boundary conditions were derived in a similar fashion to those presented in Eqs. (4.28) through (4.47). The design velocity that was used is

$$\boldsymbol{\mathcal{V}}(x, y; a) = \boldsymbol{\mathcal{V}}_x \hat{x} + \boldsymbol{\mathcal{V}}_y \hat{y} = (x/a - 1/2)\, \hat{x} \tag{4.48}$$

The strain discontinuities that occur at the stiffener are automatically accounted for in the local CSE system, because the design velocity at this point is equal to zero. Otherwise, non-homogeneous interface boundary conditions would be applied (Cross, D. and Canfield, R.A., 2013). Five-layer patches and fourth-order Taylor series expansions were used to conduct SGR. Figures 4.10 and 4.11 show a comparison of the local and total design derivative solutions using local CSA with SGR and finite differences with a 10^{-4} relative step size. The finite difference calculations are converged for a relative step size equal to 10^{-4}.

Due to the coupling effects and the presence of the in-plane load, the response is most sensitive to the design variable in the $0.5 < y < b$ part of the domain. Table 4.3 contains the maximum absolute percent relative difference between local CSA with SGR and finite difference design derivative calculations. Figures 4.10 and 4.11 and the comparison below demonstrate that local CSA with SGR can be used to non-intrusively calculate design derivatives of a nonlinear plate response using a black box tool.

Table 4.3: Maximum Absolute Percent Relative Difference of Design Derivative Results

	u'	$\dot{u}$	v'	$\dot{v}$	w'	$\dot{w}$	ψ_x'	$\dot{\psi}_x$	ψ_y'	$\dot{\psi}_y$
ϵ	3.9	7.0	1.7	3.6	1.5	2.6	2.7	3.7	8.0	8.2

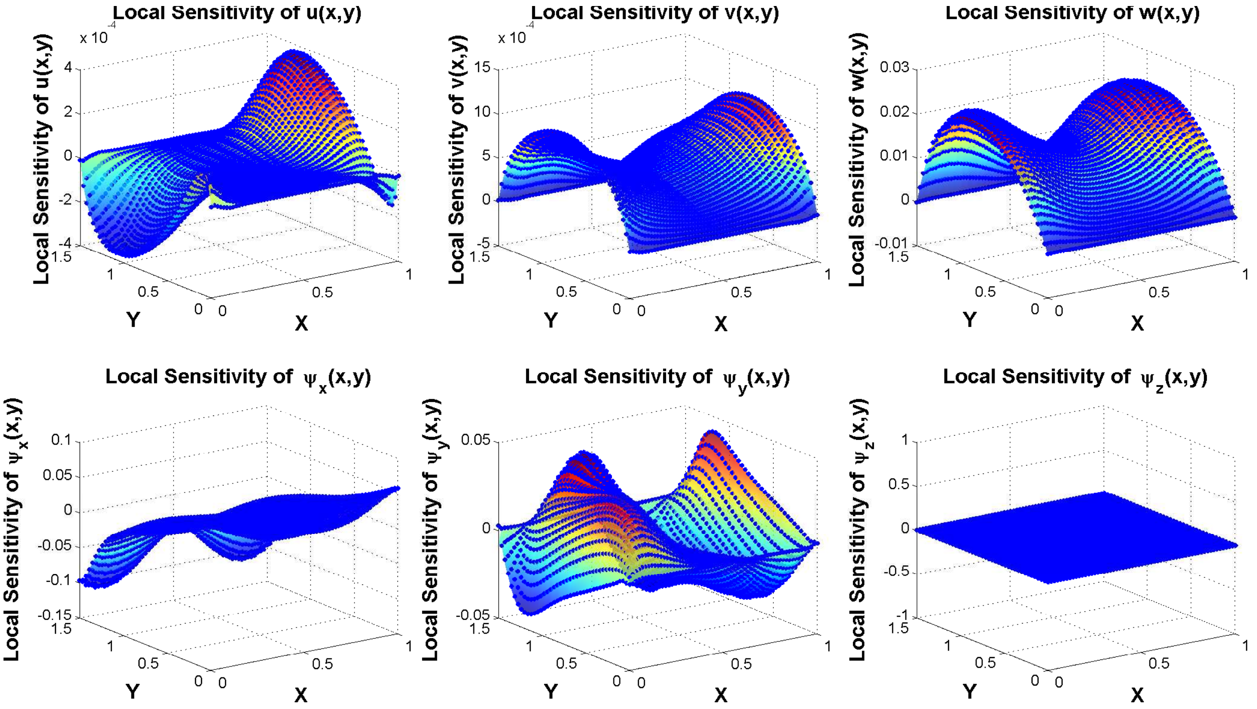

Local Sensitivity of u(x,y)
Local Sensitivity of v(x,y)
Local Sensitivity of w(x,y)
Local Sensitivity of $\psi_x(x,y)$
Local Sensitivity of $\psi_y(x,y)$
Local Sensitivity of $\psi_z(x,y)$

Figure 4.10: Local design derivatives of displacements for plate with combined loading (color surface: local CSA with SGR, blue dots: finite difference)

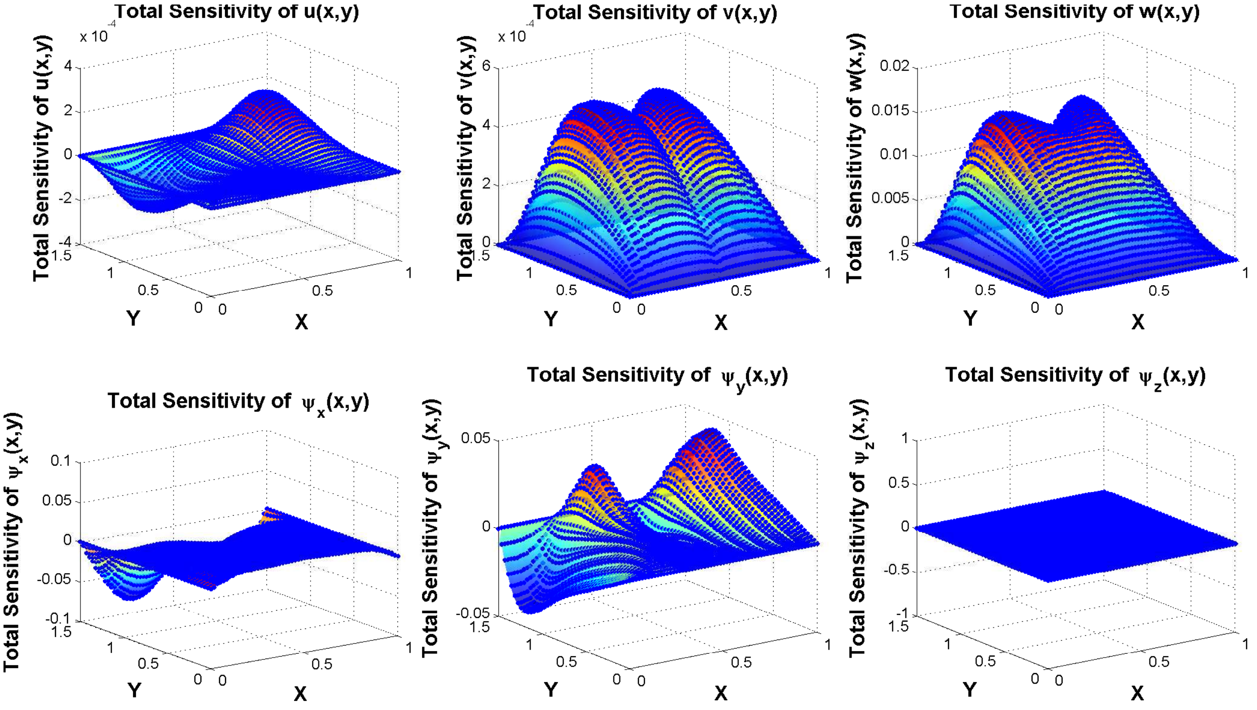

Total Sensitivity of u(x,y)
Total Sensitivity of v(x,y)
Total Sensitivity of w(x,y)
Total Sensitivity of $\psi_x(x,y)$
Total Sensitivity of $\psi_y(x,y)$
Total Sensitivity of $\psi_z(x,y)$

Figure 4.11: Total design derivatives of displacements for plate with combined loading (color surface: local CSA with SGR, blue dots: finite difference)

4.5 Discussion of Nonlinear Applications

The non-intrusive local continuum formulation with SGR was extended to nonlinear problems. The structures community most commonly uses total CSA; one reason being that approximations of high-order derivatives of primary variables in the local CSA boundary conditions introduce error. Duvigneau, R. and Pelletier, D. (2006) used local CSA to calculate fluid flow design derivatives, and achieved accurate derivative approximations by using least-squares matching of Taylor series expansions to the response data. This approach has been adapted to calculate first-order derivatives of primary and secondary response variables, needed for local CSA of structural problems. This particular formulation in conjuction with SGR allows shape design derivatives to be calculated accurately, efficiently, and non-intrusively. The method was employed without modification for nonlinear static bending of a beam modeled with both Euler-Bernoulli and Timoshenko beam theories. This example demonstrates that the approach is element agnostic. The method was also used to calculate shape design derivatives of a nonlinear transient gust response of a 2-D beam model. Local CSA has not been commonly used for structural applications because of the strain discontinuities that can occur at structural interfaces. The local CSE interface boundary conditions, which account for the discontinuities, presented in Liu, S. et al. (2010), require additional spatial derivatives of the primary and secondary response variables. This example demonstrates that as long as accurate approximations of these derivatives can be obtained from the analysis output, then accurate design derivatives can be calculated using the local CSA method. Lastly, local CSA with SGR was used to calculate shape design derivatives for nonlinear static bending of a rectangular plate with mixed boundary conditions and for a beam-stiffened rectangular plate with combined loading. This example demonstrates that local CSA with SGR can be used to non-intrusively calculate design derivatives for nonlinear plate analysis in a black box tool. Furthermore, both the gust and plate examples demonstrated that nonlinear effects can have a significant effect on the design derivative solutions.

Conventional analytic sensitivity methods cannot treat the analysis tools as a black box, as numerical methods such as finite difference do, and for that reason cannot be implemented easily for nonlinear analysis conducted by a tool like Nastran. Now, the non-intrusive local CSA method with SGR can be used to more accurately calculate shape design derivatives of black box solvers. The method is limited by requiring that the analysis tool be able to provide system matrices, primary variable response data, and secondary variable response data. As evidenced by Nastran's inability to provide element force data for nonlinear transient analysis, said requirements are not always satisfied. Lastly, it was shown by Cross, D. and Canfield, R.A. (2013) that mesh refinement may be required to achieve derivative approximations that satisfy the desired error tolerance. However, Duvigneau, R. and Pelletier, D. (2006) show that if the analysis converges asymptotically with mesh refinement, then so will the derivative approximations by way of SGR. The next chapter investigates the numerical considerations surrounding this issue and quantifies the effects that patch parameters have on the accuracy of local CSA with SGR.

It is recommended that future work extend the formulation to high-fidelity aerodynamics, which will include nonlinearities and effect the formulation in a similar manner. This capability will be necessary when implementing local CSA with SGR for high-fidelity aeroelastic analysis.

Chapter 5

Grid Convergence and Two-Dimensional Potential Flow

This chapter investigates some of the numerical considerations surrounding the local CSA method with SGR. First, the benchmark axial bar problem from Section 2.2 is revisited to study the numerical behavior associated with SGR. Next, the effects of mesh refinement near design dependent boundaries are studied for a two-dimensional rectangular membrane modeled with unstructured finite element meshes. Lastly, the local CSA method with SGR is used to compute design derivatives of a two-dimensional potential flow around a Joukowsky airfoil.

5.1 Asymptotic Accuracy of Spatial Gradient Reconstruction

5.1.1 Benchmark Axial Bar Problem

Previous examples have shown that the spatial gradient reconstruction converges and becomes more accurate as the nodal density of the patch increases. However, the nodal density is implicitly defined by the number of nodes in the global finite element mesh and the number of layers chosen for the patch. Therefore, a closer investigation is required in order to properly quantify the numerical behavior of spatial gradient reconstruction.

Figure 5.1 shows the rate of convergence of the linear finite element analysis for the benchmark axial bar problem of Section 2.2. The plot on the left shows the analytic solution, u_{EX}, and the finite element solution using 10 elements, u_{FE}. The plot on the right shows the grid convergence for the finite element solution, where the error, ϵ was calculated by

$$\epsilon = \|u_{EX} - u_{FE}\|_2 \tag{5.1}$$

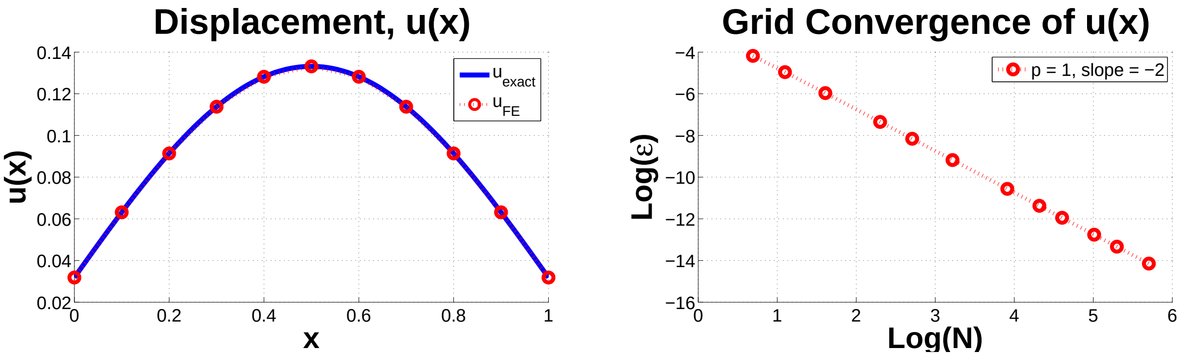


Figure 5.1: Rate of convergence for linear finite element analysis of benchmark axial bar problem

The slope of the grid convergence curve indicates that the linear finite element analysis is a second-order method. In order for the design derivative analysis to obtain the same rate of convergence, the approximations to the spatial derivatives that appear in the local CSE boundary conditions must be calculated with at least second-order accuracy.

Recall from Eq. (2.28) that the local CSE boundary conditions contain the first- and second-order spatial derivative of displacement at the end of the axial bar, i.e., $u_{,x}(1)$ and $u_{,xx}(1)$. Figure 5.2 shows the rates of convergence for approximations of these derivatives using spatial gradient reconstruction. In order to remove the numerical bias associated with the finite element analysis, the displacement data fitted by the least-squares match of the Taylor series expansion was sampled from the analytic solution. Furthermore, the patch was defined using a patch radius, r_P. The displacement data was sampled uniformly within the patch based on the number of patch layers, which varied from three to six. The order of the Taylor series expansion was chosen to be one less than the number of layers so that the number of data points was always greater than the number of unknowns. For these results, the error, ϵ is a true absolute integrated error of the first and second-order derivatives of axial displacement ($\mathcal{L}_2$ norm). The rate of convergence for the second-order derivative is always one less than that of the first-order derivative. The rate of convergence only jumps for the odd-order Taylor series, because the displacement is governed by a sine function. Therefore, going from a third-order to a fourth-order Taylor series improves the accuracy but not the rate of convergence. From these results, it is clear that in order to have at least a second-order accurate approximation of both derivatives, a four-layer patch and a third-order Taylor series expansion is required.

Figure 5.3 shows convergence of the local design derivative solutions, based on using

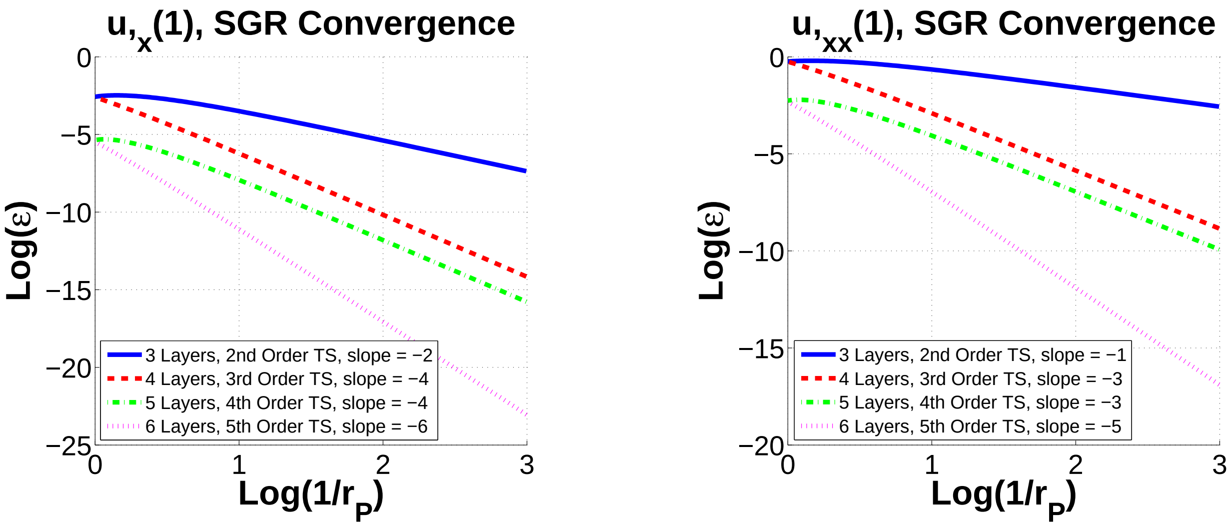


Figure 5.2: Rate of convergence for spatial gradient reconstruction of displacements

different methods to approximate the spatial derivatives that appear in the boundary conditions. The plot on the left shows the analytic local design derivative solution and five different local design derivative solutions that use 10 finite elements. The spatial derivatives in the boundary conditions were derived analytically for one solution, approximated using the shape functions for a second solution, and approximated using SGR for the other three solutions. The plot on the right shows the rate of convergence of each method as the number of finite elements is increased. The error, ϵ was again calculated using Eq. (5.1).

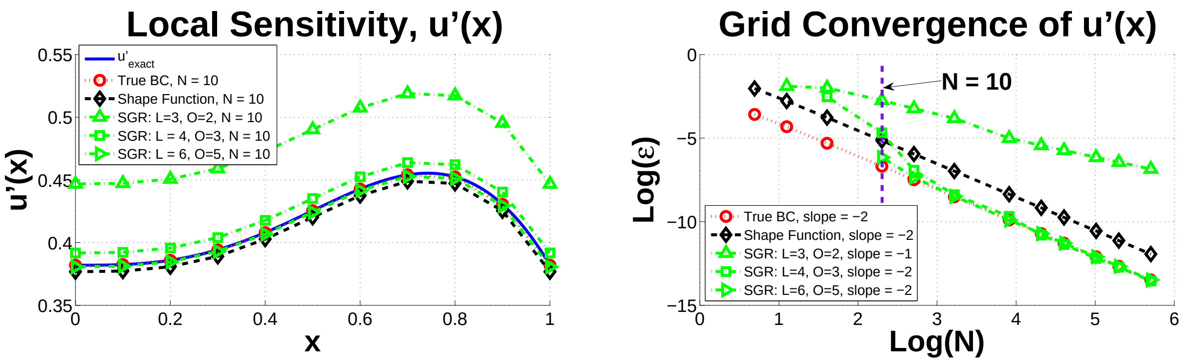


Figure 5.3: Rate of convergence for local design derivative of displacement for the benchmark axial bar problem

Approximating the derivatives with the shape functions yields a second-order rate of convergence, but does result in roughly one to two orders in magnitude more error compared to the local design derivative solution that utilizes analytically derived spatial derivatives. The linear shape functions are unable to recover the second-order spatial derivative that appears in the boundary condition. However, in this case the second-order derivative is analytically zero. Therefore, the shape functions luck out, and would typically result in a rate of converge that is at least one order lower than the analysis. When a three-layer patch and second-order Taylor series is used, the local design derivative solution is relatively inaccurate with a first-order rate of convergence. With a sufficient number of nodes in the global mesh the four-layer patch and third-order Taylor series does achieve a second-order rate of convergence. These results are consistent with the rates of convergence shown in Figure 5.2. However, a certain number of nodes in the global mesh, which populates more nodes near the boundary and makes the derivative approximations more accurate, is required to eliminate the numerical bias. As the number of nodes increases the numerical bias is reduced and the error converges asymptotically to the error curve associated with using the analytically derived boundary conditions. The six-layer patch and fifth-order Taylor series does manage to remove some of the numerical bias more quickly, but does not achieve a higher rate of convergence.

For some problems, achieving the desired accuracy may require mesh refinement near the boundary, which will reduce the amount of truncation error associated with the SGR. Figure 5.4 shows the grid convergence of the local design derivative solution when the finite element nodes are distributed quadratically (decreasing mesh size near critical boundary, $x = 1$) instead of uniformly. Now, when the four-layer patch and third-order Taylor series is used, the numerical bias is reduced more quickly.

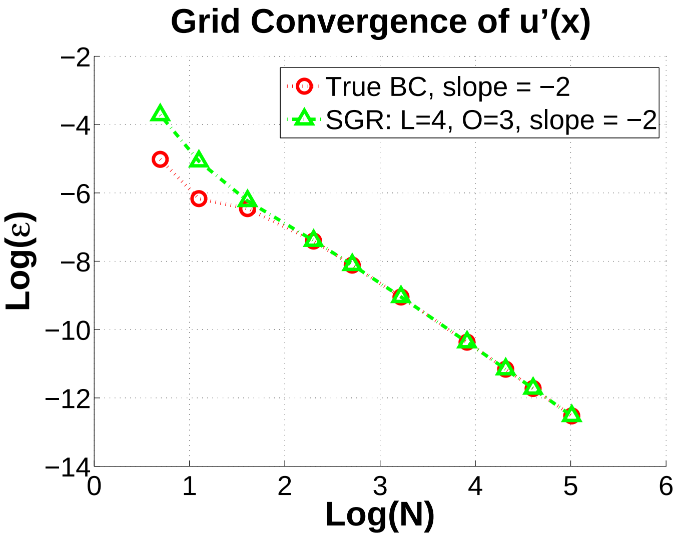


Figure 5.4: Rate of convergence for quadratically distributed finite element mesh

Another option is to use high-order shape functions instead of SGR to approximate the spatial derivatives that appear in the local CSE boundary conditions. Figure 5.5 shows the rates of convergence of the finite element analysis of the axial bar when high-order shape functions are used. The rate of convergence increases by one for each order the shape functions increases.

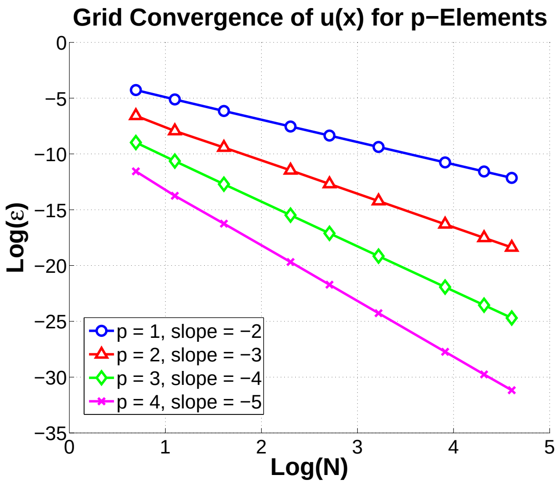


Figure 5.5: Rates of convergence for p-elements

Figure 5.6 shows the grid convergence for local design derivative solutions with second-order (p=2) shape functions. These results use the quadratically distributed finite element meshes, and the error was calculated using Eq. (5.1). Using the shape functions or SGR with a three-layer patch and second-order Taylor series fails to achieve the same rate of convergence as the analysis. Using SGR with a four-layer patch and third-order Taylor series does achieve the same rate of convergence, but with some numerical bias. The bias can be effectively removed by using a six-layer patch and fifth-order Taylor series expansion. Continuing to increase the order of the shape functions will continue to increase the order of the finite element analysis. Therefore, at some point, the order of the analysis might be larger than the order of accuracy for which one is willing to calculate the spatial derivatives. Ultimately, a trade-off between the accuracy and the computational cost associated with mesh refinement and SGR calculations must be balanced.

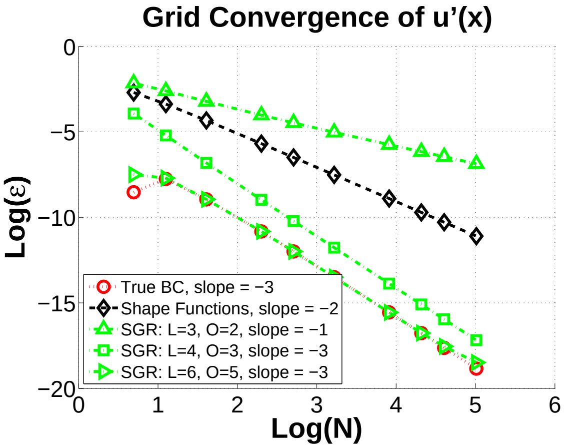


Figure 5.6: Rates of convergence for local design derivative solution using second-order (p=2) shape functions

5.2 Rectangular Membrane with Unstructured Meshes

The following example is designed to help study the effects that mesh refinement can have on the accuracy of the local CSA method with SGR. A rectangular membrane structure lies in the x-y plane and has edge lengths, s_x and s_y. The transverse displacement, $w(x, y)$ of the membrane is governed by Laplace's equation.

$$\nabla^2 w = 0 \tag{5.2}$$

A non-zero displacement is enforced along the $y = s_y$ edge, so that the boundary conditions of the system are

$$w(0, y) = w(s_x, y) = w(x, 0) = 0 \tag{5.3}$$

$$w(x, s_y) = \sin\left(\frac{\pi x}{s_x}\right) \tag{5.4}$$

The analytic solution for displacement (Brady, 2006) is

$$w(x, y) = \frac{1}{\sinh\left(\frac{\pi s_y}{s_x}\right)} \sin\left(\frac{\pi x}{s_x}\right) \sinh\left(\frac{\pi y}{s_x}\right) \tag{5.5}$$

The finite element analysis being used for this study is capable of modeling unstructured meshes using linear triangular elements. Figure 5.7 compares the finite element result using an unstructured mesh to the analytic solution.

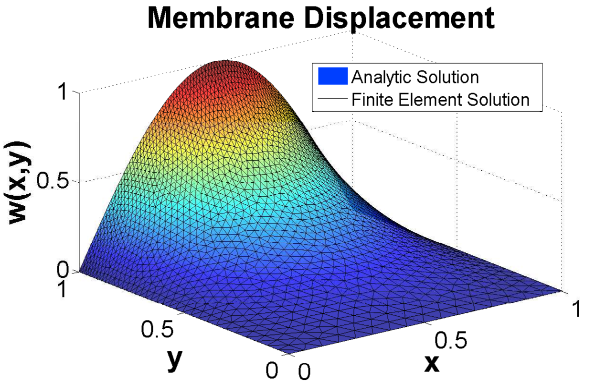


Figure 5.7: Transverse membrane displacement

The local design derivative of displacement with respect to the shape variable s_y (i.e., the length dimension of the membrane in the y-direction), consistent with the design velocity field

$$\mathcal{V} = \frac{y}{s_y} \tag{5.6}$$

is

$$w' = \frac{\partial w}{\partial s_y} = -\frac{\pi}{s_x} \coth\left(\frac{\pi s_y}{s_x}\right) \operatorname{csch}\left(\frac{\pi s_y}{s_x}\right) \sin\left(\frac{\pi x}{s_x}\right) \sinh\left(\frac{\pi y}{s_x}\right) \tag{5.7}$$

The local design derivative solution is calculated using the local CSA method with SGR. The local CSE is

$$\nabla^2 w' = 0 \tag{5.8}$$

with boundary conditions

$$w'(0, y) = -w_{,y}(0, y) \cdot \mathcal{V} \tag{5.9}$$

$$w'(s_x, y) = -w_{,y}(s_x, y) \cdot \mathcal{V} \tag{5.10}$$

$$w'(x, 0) = -w_{,y}(x, 0) \cdot \mathcal{V} \tag{5.11}$$

$$w(x, s_y) = w'(x, s_y) - w_{,y}(x, s_y) \cdot \mathcal{V} = -w_{,y}(x, s_y) \cdot \mathcal{V} \tag{5.12}$$

After solving the finite element analysis for the displacement, $w(x, y)$, SGR can be used to approximate the spatial derivatives that appear in the local CSE boundary conditions. The previous section established that, in some cases, mesh refinement is necessary to reduce the truncation error associated with SGR calculations. The design velocity has its largest magnitude along the $y = s_y$ edge. It follows that the spatial derivatives along this boundary will have the most signifcant contribution to the local design derivative solution. Therefore, mesh refinement along this edge is expected to lead to more accurate local design derivative solutions. Figure 5.8 shows three different mesh types that were used to study these effects. The left column of the figure shows examples of uniform meshes. The middle column shows non-uniform meshes with a gradually decreasing mesh size, and the right column shows non-uniform meshes with a more rapidly decreasing mesh size. A simple mesh generation MATLAB code called *DistMesh* was used (Persson, 2012). In addition, Figure 5.8 illustrates a three-layer patch used for SGR.

To quantify the errors, the $\mathcal{L}_2$ norm between the analytic solutions and the numerical finite element solutions were calculated using Eq. (5.1). Figure 5.9 shows the grid convergence of the finite element analysis using the three different mesh types. All three mesh types converge at the same rate, but the *Type* 2 non-uniform meshes sacrifice accuracy in the coarse regions of the mesh.

Figure 5.10 compares the grid convergence of local design derivative results using SGR to the grid convergence of local design derivative results that use the analytically derived local CSE boundary conditions. From Eq. (5.5), the analytic local CSE boundary conditions

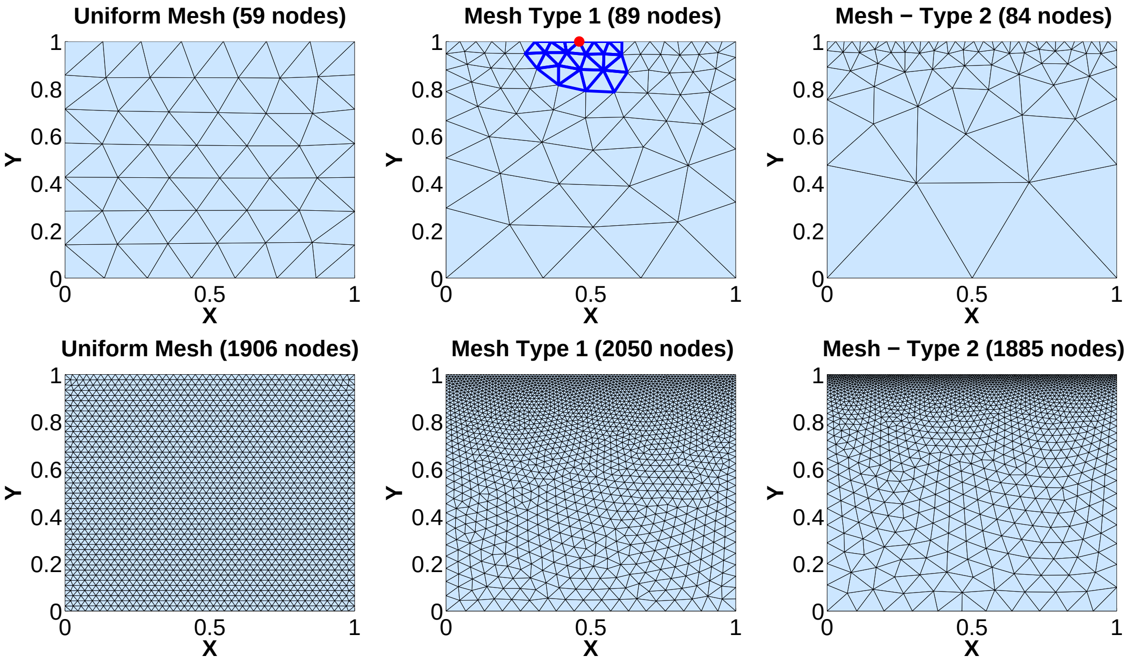


Figure 5.8: Mesh types used for membrane and local design derivative analysis

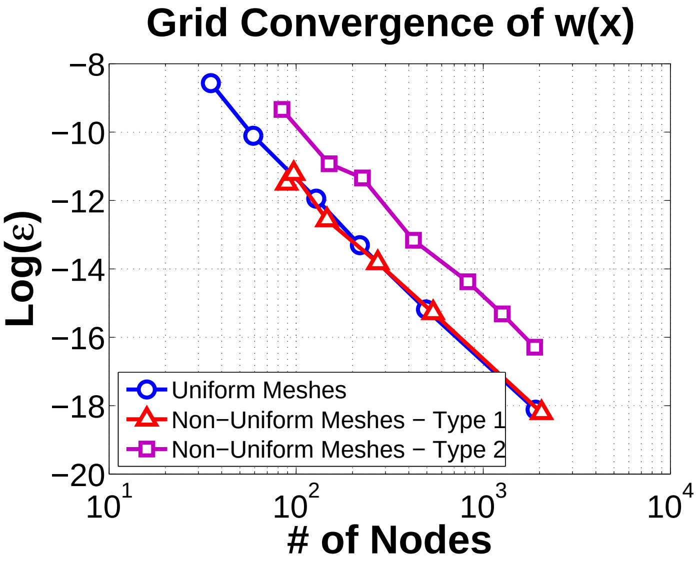


Figure 5.9: Grid convergence of membrane finite element analysis

in Eq. (5.9) through (5.12) are

$$w'(0,y) = w'(s_x,y) = w'(x,0) = 0 \tag{5.13}$$

$$w'(x,s_y) = -w_{,y}(x,s_y)\cdot\mathcal{V} = -\frac{\pi}{s_x}\coth\left(\frac{\pi s_y}{s_x}\right)\sin\left(\frac{\pi x}{s_x}\right) \tag{5.14}$$

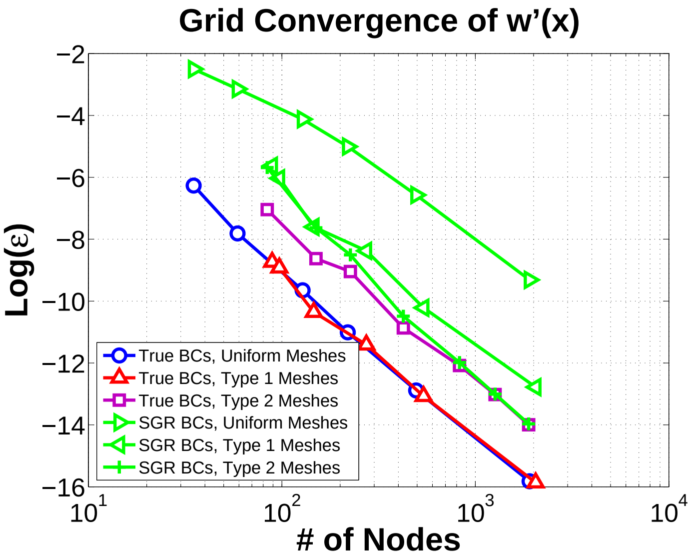


Figure 5.10: Grid convergence of local design derivative analysis

Three-layer patches and second-order Taylor series expansions were used for the SGR calculations, because the local CSE boundary conditions contain only first-order spatial derivatives. As expected, the SGR on the uniform meshes pollute the local design derivative solutions, because the nodal densities of the patches are too low. The non-unfirom meshes do much better in this regard, because the nodes are more tightly packed along the critical boundary, reducing the truncation error. However, the *Type* 1 non-uniform mesh still possesses numerical bias relative to the results that use analytic boundary conditions. Ultimately, it can be seen that the *Type 2* mesh is converging asymptotically to the error associated with using analytic boundary conditions. However, this came at the price of sacrificing accuracy of the response in other regions of the mesh. This example illustrates that mesh refinement near boundaries where the design velocity is high may significantly improve the accuracy of shape design derivatives. Even if the design velocity is high at a particular point, the critical nature of the local CSE boundary condition accuracy is also driven by the magnitude of the spatial derivative. Therefore, it is recommended that the convective term be used as a metric for results based mesh refinement. The design velocity can serve as the metric for non-results based mesh refinement or to initialize the mesh prior to the first analysis.

5.3 Potential Flow Around Joukowsky Airfoils

The mesh refinement that may be required to improve the accuracy of the SGR calculations can be limiting for structural applications, because structural meshes are often coarse, relatively speaking. However, for computational fluid dynamics the analysis itself typically requires mesh refinement near aerodynamic surfaces. Therefore, fluid meshes are typically well suited for the SGR calculations, and any mesh refinement that may be required is more easily afforded. The primary focus of this research has been structural applications. In this section, the local CSA method with SGR is applied to a two-dimensional potential flow analysis, as a step towards local CSA of high-fidelity fluid structure interaction. Wickert, D.P. and Canfield, R.A. (2008) and Liu, S. and Canfield, R.A. (2012) used local CSA to solve for design derivatives of the potential flow solution around a NACA airfoil. They used shape functions to approximate the spatial derivatives of the flow that appear in the local CSE boundary conditions. Here, it is proposed that SGR be used to approximate these derivatives. Furthermore, using a Joukowsky airfoil instead of a NACA airfoil will provide an analytic design derivative solution, which can be used for verification.

5.3.1 Local CSA Formulation for 2-D Potential Flow

The stream function of the flow, Ψ is governed by the Laplacian equation

$$\nabla^2\Psi = 0 \tag{5.15}$$

The x- and y-compenents of velocity, u and v, respectively, are related to the stream function by

$$u = \frac{\partial\Psi}{\partial y},\ \ v = -\frac{\partial\Psi}{\partial x} \tag{5.16}$$

The stream function at the far field boundary, Γ_f can be specified as

$$\Psi|_{\Gamma_f} = U_\infty y\cos\alpha - U_\infty x\sin\alpha \tag{5.17}$$

where U_∞ is the free stream velocity and α is the orientation of the flow (angle of attack). Due to the no-penetration boundary condition, the flow velocity on the airfoil surface is always tangent to the airfoil. Therefore, the stream function is a constant, resulting in an essential boundary condition.

$$\Psi|_{\Gamma_a} = \Psi_a \tag{5.18}$$

Here, Ψ_a is an unknown constant along the entire airfoil boundary. Lastly, the Kutta-condition at the trailing edge of the airfoil must also be specified.

$$\nabla\Psi\cdot\hat{n}_{te} = 0 \tag{5.19}$$

Here, $\hat{n}_{te}$ is a unit vector that bisects the upper and lower airfoil surfaces at the trailing edge. This boundary condition ensures that the flow at the trailing edge is tangential to this unit vector. From Liu, S. and Canfield, R.A. (2012) and the local CSA formulation presented in

Chapter 2, the local CSE for a shape design variable is

$$\nabla^2 \Psi' = 0 \tag{5.20}$$

The stream function at the far field boundaries is independent of any airfoil shape design parameters and the location of the far field boundary is also independent of any airfoil shape design parameters. Therefore, the far field boundary condition of the local CSE system is

$$\Psi'|_{\Gamma_f} = 0 \tag{5.21}$$

The essential boundary condition on the airfoil for the local CSE system is

$$\Psi'|_{\Gamma_a} = \dot{\Psi}_a - \left(\nabla\Psi \cdot \boldsymbol{\mathcal{V}}\right)|_{\Gamma_a} = \Psi'_a + \nabla\Psi_a \cdot \boldsymbol{\mathcal{V}}|_{\Gamma_a} - \left(\nabla\Psi \cdot \boldsymbol{\mathcal{V}}\right)|_{\Gamma_a} \tag{5.22}$$

Lastly, the sensitivity constraint derived from the Kutta-condition is

$$\nabla\Psi' \cdot \hat{n}_{te} = -\nabla\left(\nabla\Psi \cdot \hat{n}_{te}\right) \cdot \boldsymbol{\mathcal{V}}|_{te} - \nabla\Psi \cdot \hat{n}'_{te} \tag{5.23}$$

Upon solving the local CSE, Eq. (5.20), with the boundary conditions in Eqs. (5.21) through (5.23) the total design derivatives of the velocity components may be computed as

$$\dot{u} = u' + \nabla u \cdot \boldsymbol{\mathcal{V}} \tag{5.24}$$

$$\dot{v} = v' + \nabla v \cdot \boldsymbol{\mathcal{V}} \tag{5.25}$$

The coefficent of pressure

$$C_p = 1 - \frac{u^2 + v^2}{U_\infty^2} \tag{5.26}$$

has a total design derivative

$$\dot{C}_p = -\frac{2}{U_\infty^2}\left(u\dot{u} + v\dot{v}\right) \tag{5.27}$$

SGR is used to approximate the spatial derivatives of the stream function that appear in the local CSE boundary conditions, (5.22) and (5.23). Furthermore, SGR is used to approximate the spatial derivatives of velocity that appear in Eqs. (5.24) and (5.25). SGR can be used either to approximate second-order derivatives of the stream function or approximate first-order derivatives of velocity output.

5.3.2 Joukowsky Airfoil Transformation

The Joukowsky transformation is a conformal map that is popularly used for airfoil design studies. The transform is mathematically defined as

$$z = \xi + \frac{1}{\xi} \tag{5.28}$$

where $z = x + iy$ is a complex coordinate in the new space and $\xi = \chi + i\eta$ is a complex coordinate in the original space. The Joukowsky transform can be used to map circles in the ξ space into airfoil shapes in the z space. The specific shape of the airfoil depends on the centroid of the circle, $\mu = \mu_x + i\mu_y$. Therefore, in the context of design sensitivity analysis, μ_x and μ_y are the design variables, which parameterize the airfoil geometry. In addition, the potential flow solution around a circle, $\widetilde{W}$, in the ξ space can be transformed into the potential flow solution around the airfoil, W, in the z space.

$$W = \frac{\widetilde{W}}{\frac{dz}{d\xi}} = \frac{\widetilde{W}}{1 - \frac{1}{\xi^2}} \tag{5.29}$$

By the superposition of a uniform flow, a doublet, and a vortex, the analytic solution for flow around a circular cylinder (Kapania, N.R. et al., 2008) is

$$\widetilde{W} = U_\infty e^{-i\alpha} + \frac{i\Gamma}{2\pi\left(\xi - \mu\right)} - \frac{U_\infty R^2 e^{i\alpha}}{\left(\xi - \mu\right)^2} \tag{5.30}$$

where $\mu = \mu_x + i\mu_y$ is the coordinate center of the circle, $R = \sqrt{\left(1 - \mu_x\right)^2 + \mu_y^2}$ is the radius of the circle, and Γ is the circulation, which is defined as

$$\Gamma = 4\pi U_\infty R \sin\left(\alpha + \sin^{-1}\left(\frac{\mu_y}{R}\right)\right) \tag{5.31}$$

The velocity compenents are related to the complex flow solution by

$$W = u - iv \tag{5.32}$$

Figure 5.11 shows the analytic potential flow solutions around a cylinder and the coresponding Joukowsky airfoil. In this case, $\mu = -0.13 + 0.01i$, which roughly approximates a NACA 2215 airfoil (Kapania, N.R. et al., 2008), and $\alpha = 10$ degrees.

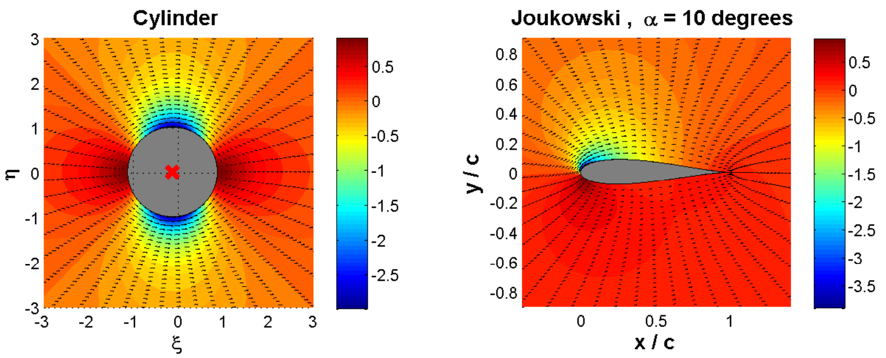


Figure 5.11: Flow solution around cylinder (left) and the transformed flow solution around the corresponding Joukowsky airfoil (right)

The analytic design derivatives can be derived by differentiating Eq. (5.29) with respect to the design variables. The design variables which control the airfoil shape are μ_x and μ_y. Figure 5.12 shows how each of the design variables affects the airfoil shape. In general, μ_x controls the thickness and μ_y controls the camber.

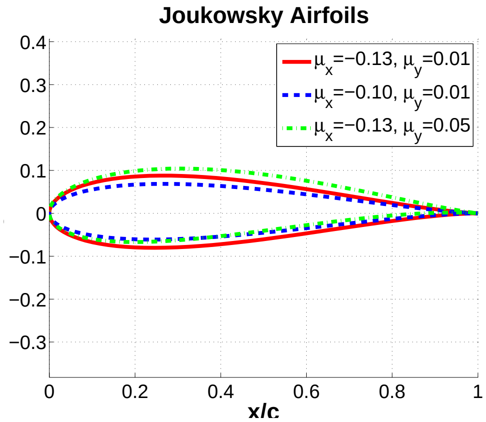


Figure 5.12: Shape parameterization of Joukowsky airfoils

5.3.3 Design Derivative Results of Potential Flow Analysis

DistMesh was used to generate an unstructured mesh (Persson, 2012) of linear triangular elements with 4452 nodes. Figures 5.13 and 5.14 show the Joukowsky airfoil, whose design parameters are $\mu_x = -0.13$ and $\mu_y = 0.01$, which roughly approximates a NACA 2215 airfoil section. Finite element analysis using linear triangular elements was used to approximate the stream function field around the airfoil. SGR was used to approximate the horizontal and vertical velocities around the airfoil. In addition, the velocities were used to calculate the coefficient of pressure around the airfoil. Four-layer patches and third-order Taylor series expansions were used for SGR. Figure 5.15 compares the calculated coefficient of pressure and velocity distributions to the analytic solutions. For clarity, the approximate solutions were plotted only every fifteenth node around the airfoil.

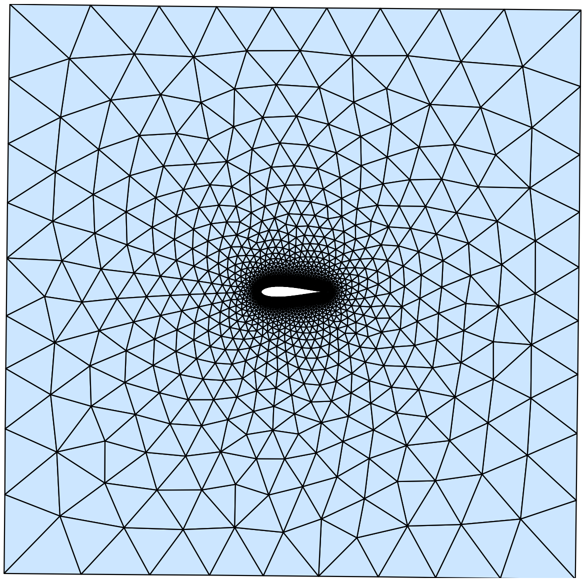


Figure 5.13: Finite element mesh for Joukowsky airfoil

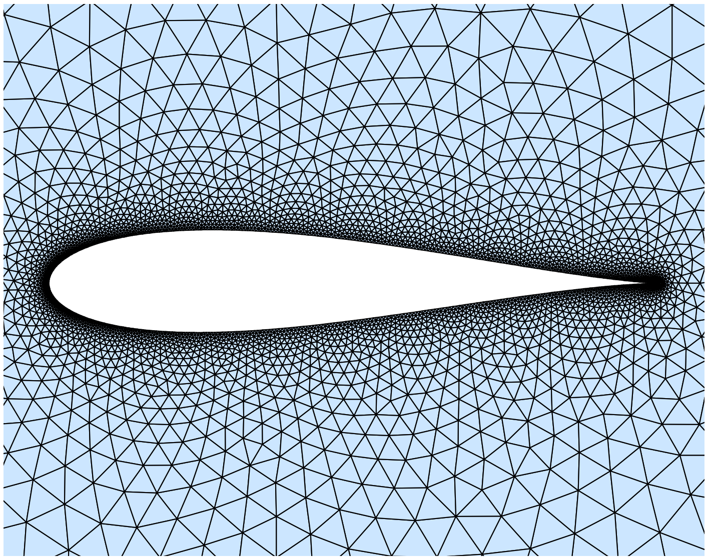


Figure 5.14: Close up view of finite element mesh around Joukowsky airfoil

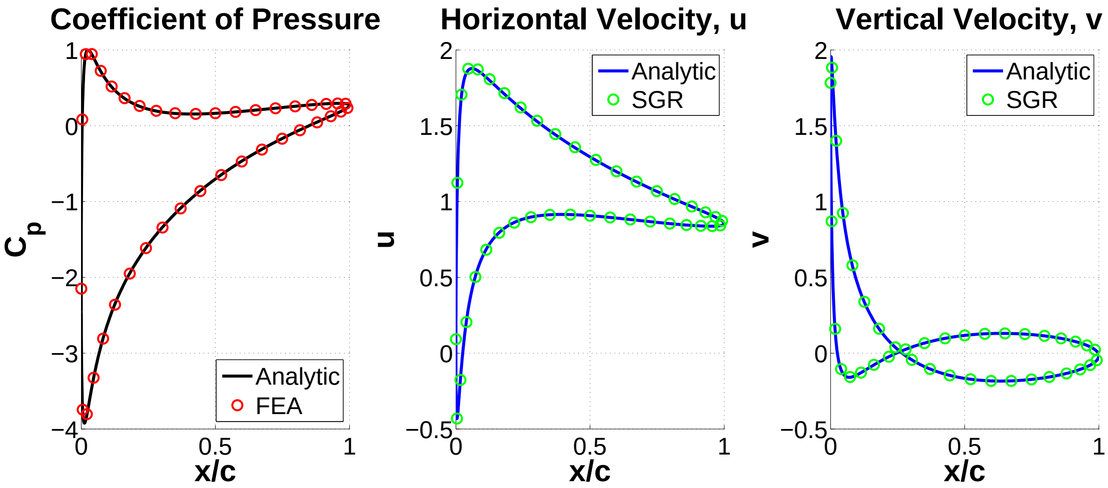


Figure 5.15: Finite element analysis and SGR results for Joukowsky airfoil

Next the local CSE boundary conditions are formulated from the velocities, the design velocities, and the sensitivity of the trailing edge vector. The local design derivative of the stream function is solved by applying these boundary conditions. From Eqs. (5.16), (5.24), and (5.25), the total design derivatives of velocity are

$$\dot{u} = \Psi'_{,y} + \Psi_{,xy}\,\mathcal{V}_x + \Psi_{,yy}\,\mathcal{V}_y \tag{5.33}$$

$$\dot{v} = \Psi'_{,y} - \Psi_{,xx}\,\mathcal{V}_x - \Psi_{,xy}\,\mathcal{V}_y \tag{5.34}$$

where SGR is used to calculate the first-order spatial derivatives of Ψ' and the second-order spatial derivatives of Ψ. At any given finite element node, the first-order spatial derivatives of Ψ' are approximated using a single Taylor series expansion of Ψ', and the second-order spatial derivatives of Ψ are approximated using a single Taylor series expansion of Ψ. The total design derivative of coefficient of pressure is calculated by Eq. (5.27). Figures 5.16 and 5.17 compare the calculated total design derivatives of the coefficient of pressure and velocity components to analytic solutions. Figure 5.16 shows total design derivative solutions of the flow with respect to the design variable μ_x. Figure 5.17 shows total design derivative solutions of the flow with respect to the design variable μ_y.

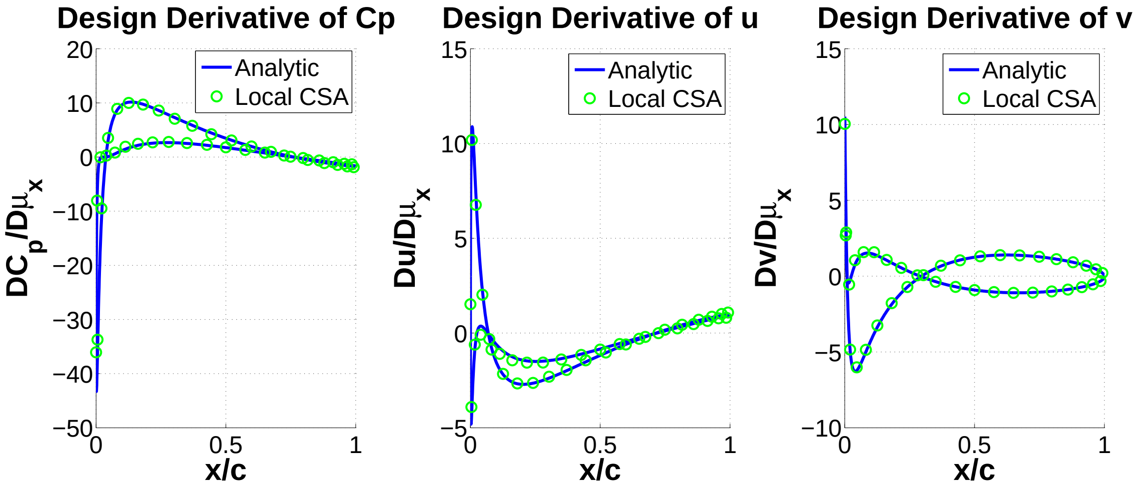


Figure 5.16: Total design derivatives of potential flow w.r.t. μ_x

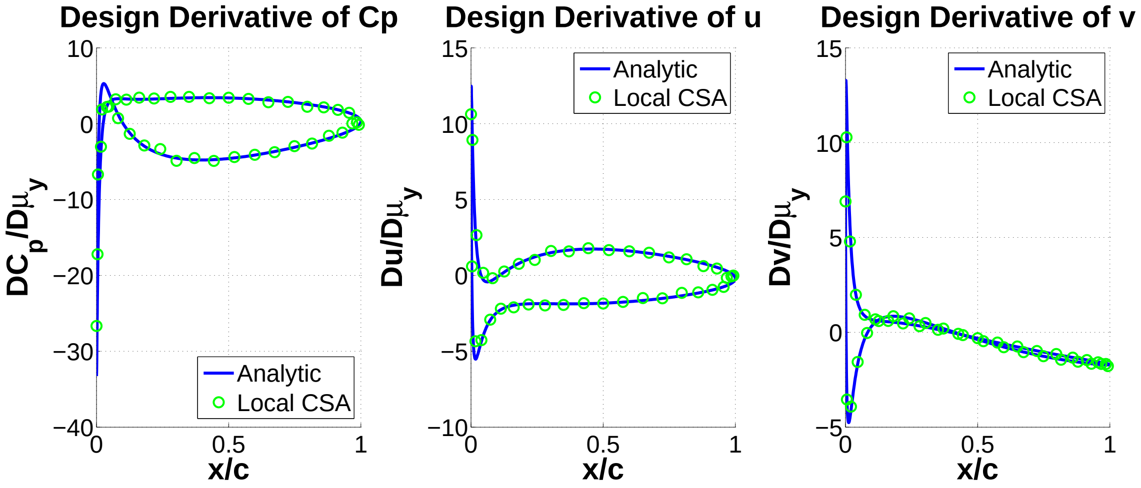


Figure 5.17: Total design derivatives of potential flow w.r.t. μ_y

The results presented here come from a finite element mesh with 4,453 nodes, of which 604 are on the airfoil surface. The true relative error of the coefficient of pressure is calculated by

$$\epsilon = \left| \frac{\oint C_p ds - \sum C_p \Delta s}{\oint C_p ds} \right| = 0.004 \tag{5.35}$$

Meshes containing 5,514 and 6,933 nodes were used to see if the finite element analysis was converged. These meshes had 666 and 745 nodes on the airfoil surface, respectively. Each of these results yielded a true relative error of the coefficient of pressure approximately equal to 0.004. It can be concluded from this that the finite element analysis is converged, and that the error is the product of the farfield perimeter of the mesh. For each of the meshes, the farfield perimeter had coordinate distances of 15. Table 5.1 shows that despite the analysis being converged, the design derivative calculations are not. The coefficient of pressure is only calculated from SGR approximations of first-order derivatives, but the design derivative of the coefficient of pressure is also dependent on second-order derivative approximations. The mesh refinement improves the accuracy of second-order derivative approximations necessary to calculate the design derivatives.

Table 5.1: Absolute Relative Errors for Design Derivatives of Coefficient of Pressure

# of Nodes	$\dot{C}_{p_{\mu_x}}$	$\dot{C}_{p_{\mu_y}}$
4453	0.0307	0.0473
5514	0.0292	0.0358
6933	0.0119	0.0232

5.4 Conclusions Regarding Grid Convergence

Grid convergence was studied for the benchmark axial bar problem and for a rectangular membrane structure. The results for the axial bar demonstrated that the design derivative solution will converge asymptotically with decreasing mesh size. It was also demonstrated that, depending on the convergence rate of the analysis and the order of the spatial derivatives in the local CSE boundary conditions, increasing the order of the Taylor series used for the SGR calculations may result in diminishing returns. The example also demonstrated that refining the mesh, especially selectively, near the design dependent boundaries reduces the numerical bias in accuracy that stems from the SGR calculations. Furthermore, this example demonstrates that SGR can have a higher convergence rate than relying on low-order shape functions to recover the spatial derivatives in the local CSE boundary conditions. It is recommended that the order of the Taylor series expansions used for SGR be at least one order higher than the order of the PDE.

The membrane example studied the effects of two-dimensional mesh refinement on accuracy. The uniform mesh was shown to be inefficient for SGR mesh refinement near the design dependent boundaries. The non-uniform meshes were much more efficient. The local sensitivity analysis benefits the most from a non-uniform mesh that has rapidly decreasing mesh size moving towards the design dependent boundaries (*Type* 2). However, the global accuracy of the original finite element analysis is worse for this type of mesh because of the errors associated with integrating the solution over the coarse regions in the mesh. It is recommended that the convective term associated with each of the local CSE boundary conditions be used as a metric for results based adaptive meshing. It is also important to consider the design velocities associated with each shape design variable.

Lastly, local CSA with SGR was implemented for two-dimensional potential flow around an airfoil. Using a Joukowsky airfoil provided the ability to compare the finite element solutions of the potential flow and its design derivatives to analytic solutions. This example demonstrated that SGR could successfully be employed on an unstructured mesh with curvilinear boundaries. Furthermore, the example illustrated that mesh refinement may be necessary to achieve accurate approximations of second or higher-order derivatives on the aerodynamic surface. However, the restrictions of mesh refinement will be less if only first-order derivatives need to be approximated. In this case, the primary variable was the stream function, which required first-order derivative approximations to compute velocities and the coefficient of pressure. Furthermore, it required second-order derivative approximations to compute the design derivatives. CFD tools will typically provide velocities as primary variables and fluxes as secondary variables. Implementing local CSA with SGR with CFD tools will only require first-order derivative approximations of velocities and fluxes. This is a recommended topic for future work.

Chapter 6

Conclusions and Future Work

6.1 Research Summary

This research is motivated by nontraditional, highly flexible aircraft configurations, which necessitate the use of high-fidelity models and design optimization at early stages of the design process, when critical design decisions are made. A significant hurdle in providing such a capability is having an accurate, efficient, and robust design sensitivity analysis (DSA) method. A brief literature review highlighted the limitations of traditional DSA methods. The numerical methods, despite having a general formulation, can be computationally expensive, inaccurate, or difficult to implement. The analytical methods have several clear advantages. They are typically highly accurate and computationally less expensive, but they can be difficult to formulate and often times, if access to the source code is unavailable, the implementation may be infeasible. The latter is especially true for nonlinear problems. Barthelemy, B. and Haftka, R.T. (1988) state that the semi-analytic method, desipite being amenable to general purpose codes, can be highly inaccurate for shape design derivative calculations. Early literature by Arora, J. and Haug, E. (1979), Haftka, R.T. and Adelman, H.M. (1989), and Haftka, R. and Gurdal, Z. (1992) suggested that continuum sensitivity analysis (CSA) may be able to overcome these limitations. However, a more in depth literature review of CSA revealed that conventional implementation does depend on the specifics of the finite element formulation. CSAheretofore required specific knowledge of the analysis formulation, shape functions, and other numerical techniques used to solve the system. When using general purpose codes, the user may not have all of this information available.

The literature review did garnish a deep understanding of the difference between local and total shape design derivatives and their role in CSA. Traditionally, total CSA has been used for structural applications, because gradient-based shape optimization of structures typically requires the total design derivatives. Also, total CSA is more amenable to built-up structures than local CSA, because continuity of the design derivative variables is maintained. In contrast, local CSA is popularly used for fluids applications or those formulated in an Eulerian reference frame. Chapter 2 discussed in detail the differences between the local and total CSA formulations. For instance, under certain conditions, total CSA is equivalent to the discrete analytic method (Akbari, J. et al., 2010; Liu, S. and Canfield, R.A., 2013a), which was shown to be not amenable to general purpose codes without access to source code. In contrast, formulation of the local CSA revealed the CSEs to be less intrusive than the other analytical methods. In fact, the only knowledge required of the finite element source code are the definitions of the boundary condition operators and the shape functions. However, conventional implementation of local CSA is less accurate than the other methods, because high-order spatial derivatives appear in the boundary conditions.

One significant contribution of this research is overcoming the accuracy issues that surround conventional implementation of local CSA. By using spatial gradient reconstruction

(SGR), an adaptation of the Taylor series l-patch method (Duvigneau, R. and Pelletier, D., 2006), accurate boundary conditions can be formulated for the local CSEs. Chapter 2 provided a detailed comparison of all DSA methods, and implemented each method for a single benchmark problem. The benchmark problem demonstrated that SGR can reliably produce more accurate local CSE boundary conditions than using shape functions. Furthermore, it demonstrated that local CSA with SGR can be as or more accurate than other DSA methods, including the discrete analytic method.

Another key contribution of this research is the ability to conduct local CSA with SGR nonintrusively for practical modeling applications. In Chapters 3 and 4, local CSA with SGR was implemented for a variety of linear and nonlinear systems using general purpose codes. The codes were treated entirely as black boxes, meaning that no regard whatsoever was given to the specific formulation used by the analysis tool (i.e., discretization, shape functions, beam/plate theory, other kinematic assumptions, etc.). This was one of the primary motivators for this research. The literature contains well estabilished theory for analytic design sensitivity analysis, but the methods have not been widely used, because they can typically only be implemented with access to source codes. This is especially true for nonlinear analysis. Local CSA with SGR can be used with general purpose codes, and in that regard sets it apart from the other analytic methods.

Design derivatives were calculated for linear and nonlinear static beam bending, linear and nonlinear transient aeroelastic gust analysis of a 2-D beam model, linear and nonlinear static bending of rectangular plates, linear and nonlinear static bending of a beam-stiffened rectangular plate, and two-dimensional potential flow around a Joukowsky airfoil. As previously mentioned, all implementations utilized general purpose codes, meaning the source code could not be or was not accessed. The element-agnostic nature of the method was demonstrated in Section 4.2, by computing design derivatives of a beam response using different beam theories, and in Section 3.2.1, by computing design derivatives of a plate response using different plate finite elements without a change to the CSA algorithm. This is a significant contribution, because conventional analytic DSA methods would require formulation of separate algorithms, whereas local CSA with SGR can handle the full variety of models with a single algorithm. The gust models help to illustrate how the method is to be implemented for FSI applications and the role that is played by flow sensitivities. The 2-D beam model and the beam-stiffened plate demonstrate how local CSA with SGR can be used when structural interfaces are present. In addition, Sections 3.1.2, 3.2.4, and 5.3 show how the method can be used to compute design derivatives of secondary variables (e.g., element forces, stress, velocity, and pressure). This research marks the first application of local CSA for nonlinear analysis of structural models using plate elements. Furthermore, it is the first application of local CSA to a built-up structure that uses mixed element types (plate and beam elements). Lastly, it is the first application of local CSA to nonlinear analysis of structures using a black box code.

Another key contribution of this research is the first use of SGR in the context of structural sensitivity. It also marks the first time that SGR has been used to approximate first-order derivatives of secondary variables that are needed to formulate the local CSE boundary conditions. This is a key ingredient in making the method element agnostic and amenable to black box codes. Furthermore, the examples demonstrate ways to control the accuracy and computational efficiency of the method.

6.2 Summary of the Method and Computational Considerations

Local CSA with SGR is a method for calculating shape design derivatives. This method has been primarily motivated by using the computed design derivatives of high-fidelity analyses to support gradient based shape optimization. However, there exist other potential uses of the design derivative information. For instance, the derivatives of objective, constraint, or other performance based functions with respect to shape parameters can be used to help develop and validate surrogate and reduced order models. Another example would be to treat control surface deflection as a shape variable. This could be used to provide the sensitivities of maneuver analysis, which could then be used to help validate and project test points. These are just a few applications, in addition to optimization, for which shape design derivatives are useful.

As illustrated by Figure 2.11, local CSA with SGR can be divided into six critical steps. The first step is to run the desired analysis. This can be done with the user's analysis tool of choice. Because this is a nonintrusive the method, the analysis tool can be a black box. This provides a unique alternative to other analytic DSA methods, which cannot be used in a straight forward manner with black box tools. It follows that even if the analysis tool is not a black box, the user can treat it as one if desired.

The second step is to parse the necessary analysis output. The system matrices, primary response variable output, and secondary response variable output are required to conduct local CSA with SGR. If the analysis tool cannot provide this information, then the local CSA method with SGR cannot be performed as outlined here. This is one limitation of the method. The examples used in this research utilized MSC Nastran to conduct the analyses. DMAP alters were used to obtain the system matrices. Stnadard output provided the primary (displacement and rotation) and secondary (force and moment) variable values. Nastran was able to provide the required data for each example except the nonlinear transient aeroelastic gust analysis, because Nastran does not support element force output for nonlinear transient analysis in SOL 400.

Once the necessary data is obtained, the third step is to conduct spatial gradient reconstruction (SGR). To be computationally efficient, the user should identify all finite element nodes that lie on design dependent boundaries. Based on the boundary conditions at each node the user should identify for which primary and secondary response variables the first-order spatial derivatives must be approximated. This involves defining a local patch and Taylor series expansion at each of these nodes. Then for each node the coefficients of the Taylor series expansion are computed by solving a least squares system. In addition, interface boundary conditions may be required if any first-order strain discontinuities are present at structural interfaces. Therefore, an optional step is to automatically detect interface discontinuities using SGR. The computational cost associated with SGR is relatively low. It requires solving a small linear least-squares regression for each patch adjacent to boundary or interface nodes. The computations only occur once no matter how many shape variables are used to parameterize the design.

The fourth step is to use the reconstructed spatial derivatives to formulate the convective terms that appear in the local CSE boundary conditions. This step has negligible computational cost. After formulating the local CSE boundary conditions, the fifth step is to solve the local CSE system. Once the local design derivative solution is computed, the final step is to transform the local design derivative solution into a total design derivative solution. This last step is only necessary at finite element nodes where the total design derivative is desired. The transformation involves adding the convective term, which contrains spatial derivates. Therefore, SGR must also be conducted to approximate the spatial derivatives of the response variables at locations where the total design derivative is desired.

The examples that were presented provide insight to how the user might control the accuracy and computational efficiency of the method. In particular, the convergence results for the axial bar problem and the rectangular Kirkhoff plate demonstrate that the accuracy of the design derivative solution is driven by the accuracy of SGR calculations. The axial bar problem demonstrates that the rate of convergence for the local CSA with SGR can only be as high as the rate of conergence of the original analysis. In order to achieve this, two things must be done. First, the spatial derivatives in the local CSE boundary conditions must be approximated with an order of accuracy that is at least equivalent to the rate of convergence. This can be done by using a Taylor series order that is equal to the order of the analysis method plus the order of the highest-order derivative in the local CSE boundary conditions less one. Second, the nodal density of the patches used in SGR must be sufficient, becuase otherwise the calculations suffer from truncation error introduced by points that are relatively far away from the expansion point. However, in the limit as the nodal density of patches goes to infinity, the local design derivative solution converges asymptotically to the case of using analytically derived boundary conditions. The rectangular membrane example in Section 5.2 demonstrated that mesh refinement along design dependent boundaries increases the nodal density of SGR patches and thereby improves the accuracy of the design derivative solution. It is recommended that the convective term associated with each design variable be used as a metric for mesh refinement. The contribution of the local CSE boundary conditions at a given finite element node to the design derivative solution can be quantified by the relative magnitude of the convective term at said finite element node. This involves the product of the magnitudes of the spatial derivative and design velocity terms. Therefore, to converge the accuracy of the method, the mesh should be refined near nodes where the convective terms are high. In addition, the mesh should be refined near nodes inside the domain where the total design derivative solution is desired.

As with any numerical method, there exists a tradeoff between accuracy and efficiency. Refining the mesh, increasing the nodal density of patches, solving the least squares systems, and solving the local CSE systems associated with multiple design variables all increase the computational cost of the method. However, for practical problems with many design variables, the method can be extended to an adjoint formulation (see Appendix B). The adjoint equation can be solved using an already decomposed stiffness matrix and a reduced right-hand-side loading, which makes the expense of solving the adjoint equation negligble compared to solving the analysis system numerous times to conduct finite difference calculations. While the computational cost associated with SGR may seem high, the local CSE boundary conditions contain the same spatial derivatives for each design variable. Therefore, even for problems with hundreds or thousands of shape variables, the spatial derivatives only need to be approximated using SGR one time following the analysis. Relatively speaking, this computational cost becomes much more affordable. Furthermore, the number of layers in a patch should only be large enough to include more data points than there are unknown coefficients in the Taylor series order. This provides the most accurate result, while also wielding the most computationally efficient system of equations to calculate. In addition, at some point using mesh refinement to increase the nodal density of patches and increasing the Taylor series order yield diminishing returns. In the case of mesh refinement, this becomes problem dependent, where local curvature and the design order of accuracy are both factors. In the case of the Taylor series order, when a sufficient number of nodes are present, there is typically none to little advantage in increasing the order beyond the value previously defined. Doing so simply increases the computational cost of solving the least squares system, while improving the accuracy very little, due to the limitation imposed by the accuracy of the original analysis.

Some physical considerations include how best to approximate the spatial derivatives at or near structural and material interfaces, shocks, discontinuities, or places with high curvature and large gradients. First and foremost it is important to understand that mesh refinement in these areas will almost certainly be a necessity. In addition, as pointed out by Liu, S. et al. (2010), the equilibrium equations at structural and material interfaces must be differentiated with respect to the design variables in the same manner as the other boundary conditions. This serves as the derivation of the local sensitivity equilbrium equations which must be satisfied when solving the local CSE system. The resultant boundary conditions will include convective terms that require SGR to approximate spatial derivatives on either side of the interface, which capture the presence of discontinuities. This requires SGR calcuations that utilize patches adjacent to the interface on opposite sides. The patches should not cross over the interface, because Taylor series expansions are continuous equations that will not be able to capture the discontinuity. Discontiuities, such as shocks, should be treated in a similar manner to those present along the boundaries. That is, the SGR patches should be defined adjacent to the discontinuity, but never across.

In summary, this nonintrusive method provides a unique alternative to conduct DSA. Unlike the discrete analytic method and the conventional total CSA and local CSA methods, local CSA with SGR is capable of being implemented with black box tools. It has a general formulation, which is the popularizing attribute of the finite difference method, and it provides the user with unique and robust control over the tradeoff between accuracy and efficiency.

6.3 Future Work and Research

This research was motivated by the need for a nonintrusive analytic DSA method, which can be used to compute shape design derivatives of practical high-fidelity models analysed with black box codes. Such a method was developed, and local CSA with SGR has been established as a feasible method that can meet these needs. The method has been successfully demonstrated on many practical models, but mostly of low to medium fidelity. Therefore, recommendations for future work include applying the method to even higher fidelity models and continuing to investigate the physical and numerical behaviors that have been observed on the models presented in this research. One recommendation for improving SGR approximations is to use weighted least squares, so that data points closest to the expansion point have the most influence on the derivative approximations. This has not been tested in this research, but should be investigated in future work. One might also investigate simultaneous solution of all least-squares regressions by to reduce "pixelation" by sharing the enforced essential boundary conditions among the patches. In addition, the primary focus of this research has been on structural analysis, and so, as the research moves forward, it is critical to give more attention to DSA of computational fluid dynamics (CFD). The formulation of local CSA with SGR should hold for CFD, but different issues regarding implementation may arise. For instance, preliminary work in this area by Mandar Kulkharni of Virginia Tech has uncovered a significant challenge regarding the no-penetration condition. For local CSA the homogeneous no-penetration condition leads to an associated non-homogeneous local CSE boundary condition in which a non-zero flow penetration is defined. Many general purpose aerodynamic solvers may not allow this type of boundary condition to be specified. Another issue regarding the implementation is that CFD codes do not form explicit global system matrices. Therefore, future research should be planned to investigate these challenges.

In addition, an arbitrary Lagrangian-Eulerian (ALE) framework will be required to capture the rigid body motion associated with a high-fidelity transient aerolestic gust response. There certainly will be challenges associated with the implementation of CSA in an ALE framework, such as careful distinction between grid velocity in ALE and design velocity in CSA. It is also recommended that local CSA with SGR be extended to other disciplines. The examples from Chapter 5 that deal with Laplace's equation have similarities to heat convection. For this reason, it is easy to imagine extending local CSA with SGR to thermoelasticity. Dr. Woon Kim of Virginia Tech has already begun implementing CSA for electromagnetic field analysis to support the design of antenna arrays.

The convergence of accuracy has been well established in this research when mesh refinement is used. However, in many instances, the user may not have the option of mesh refinement, and he/she may be forced to use a relatively coarse mesh. In such a case, SGR may be less accurate than desired. One recommendation is to evaluate whether or not another DSA method may provide better accuracy than the low-order SGR. Future research should investigate this issue and determine if there might be ways to improve SGR accuracy other than mesh refinement. For instance, the weighted least squares approach may improve accuracy of low-order SGR. Perhaps this is an area where high-order shape functions are a better alternative, although this may jeopordize the nonintrusive nature of the methodology.

Another recommended improvement to the method is to automatically detect if and where discontinuities are present, which would enable smarter SGR patch definitions.

Bibliography

Akbari, J., Kim, N., and Ahmadi, M.T. Shape sensitivity analysis with design-dependent loadingsequivalence between continuum and discrete derivatives. *Structural and Multidisciplinary Optimization*, 2010.

Alyanak, E.J. and Kolonay, R.M. Efficient supersonic air vehicle structural modeling for conceptual design. *12th AIAA ATIO Conference*, AIAA 2012-5519, 2012.

Arora, J. and Haug, E. Methods of design sensitivity analysis in structural optimization. *AIAA Journal*, Vol. 17, No. 9, Article No. 79-4109, 1979.

Barthelemy, B. and Haftka, R.T. Accuracy analysis of the semi-analytical method for shape sensitivity calculation. *AIAA-88-2284*, 1988.

Barthelemy, J.F. and Bergen, F.D. Shape sensitivity analysis of wing static aeroelastic characteristics. *Journal of Aircraft*, 26 (8):712–717, 1989.

Bhaskaran, R. and Berkooz, G. Optimization of fluid-structure interaction using the sensitivity equation approach. *Fluid-Structure Interaction, Aeroelasticity, Flow-Induced Vibrations and Noise*, Vol. 1, No. 53-1, 1997.

Bhatia, K.G. and Rudisill, C.S. Optimization of complex structures to satisfy flutter requirements. *AIAA Journal*, Vol. 9, No. 8, 1971.

Bischof, C.H., Bucker, H.M., Rasch, A., Slusanschi, E., and Lang, B. Automatic differentiation of the general-purpose computational fluid dynamics package fluent. *Journal of Fluids Engineering*, 129 (5):652–658, 2007.

Blair, M., Canfield, R.A., and Roberts, R. A joined-wing aeroelastic design with geometric non-linearity. *Journal of Aircraft*, 42(4):832–848, 2005.

Borggaard, J. and Burns, J. A sensitivity equation approach to shape optimization in fluid flows. *Technical Report, Langley Research Center*, 1994.

Borggaard, J. and Burns, J. A pde sensitivity equation method for optimal aerodynamic design. *Journal of Computational Physics*, Vol. 136, 1997.

M. Brady. Laplace's equation, 2006. URL `http://www.robots.ox.ac.uk/~jmb/lectures/pdelecture5.pdf`. [Online; accessed 18-November-2013].

V. Braibant. Shape sensitivity by finite elements. *Journal of Structural Mechanics*, 14 (2): 209–228, 1986.

Braibant, V. and Fleury, C. Shape optimal design using b-splines. *Computer Methods in Applied Mechanics and Engineering*, 44 (3):247–267, 1984.

Braibant, V. and Fleury, C. An approximation concepts approach to shape optimal design. *Computer Methods in Applied Mechanics and Engineering*, 53 (3):119–148, 1985.

Burns, J.A. and Cliff, E.M. Pre and post processing for large scale computations in optimal design and control. *Tech. rep., Virginia Polytechnic Inst and State Univ Blacksburg Interdisciplinary Center of Applied Mathmatics*, 1998.

Carle, A., Fagan, M., and Green, L.L. Preliminary results from the application of automated code generation to cfl3d. *Proceedings, 12th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis*, pages 98–4807, 1998.

Charlot, L., Etienne, S., and Pelletier, D. A continuous lagrangian sensitivity equation method for incompressible flow. *Journal of Computation Physics*, 231:5989–6011, 2012.

K.K. Choi. Shape design sensitivity analysis of displacement and stress constraints. *Journal of Structural Mechanics*, 13:27–41, 1985.

Choi, K. and Kim, N.H. *Structural Sensitivity Analysis and Optimization.* Springer Science + Business Media, 2005.

Choi, K.K. and Haug, E.J. Shape design sensitivity analysis of elastic structures. *Journal of Structural Mechanics*, 11 (2):231–269, 1983.

Choi, K.K. and Seong, H.W. A domain method for shape design of built-up structures. *Computer Methods in Applied Mechanics and Engineering*, 57:1–15, 1986.

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

Chun, Y.W. and Haug, E.J. Shape optimization of a solid revolution. *Journal of Engineering Mechanics*, 109 (1):30–46, 1983.

Colin, E., Pelletier, D., and Borggaard, J. Application of a sensitivity equation method to turbulent flows with heat transfer. *International Journal of Thermal Sciences*, 44 (1): 1024–1038, 2005.

Cross, D. and Canfield, R.A. Solving continuum shape sensitivity with existing tools for nonlinear aeroelastic gust analysis. *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, Hawaii.*, AIAA 2012-1923, 2012a.

Cross, D. and Canfield, R.A. Continuum shape sensitivity with spatial gradient reconstruction of nonlinear aeroelastic gust response. *14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Indianapolis, Indiana.*, AIAA 2012-5597, 2012b.

Cross, D. and Canfield, R.A. Local continuum shape sensitivity with spatial gradient reconstruction. *Structural and Multidisciplinary Optimization Journal*, (accepted with revisions), 2013.

Dems, K. and Haftka, R. Two approaches to sensitivity analysis for shape variation of structures. *Mech. Struct. and Mach.*, Vol. 16, No. 4, 1989.

Dems, K. and Mroz, Z. Variational approach by means of adjoint systems to structural optimization and sensitivity analysis. part i - variation of material parameters within fixed domain. *International Journal of Solids and Structures*, 19 (8):677–692, 1983.

Dems, K. and Mroz, Z. Variational approach by means of adjoint systems to structural optimization and sensitivity analysis. part ii - structure shape variation. *International Journal of Solids and Structures*, 20:527–552, 1984.

Dems, K. and Mroz, Z. Variational approach to first- and second-order sensitivity analysis of elastic structures. *International Journal for Numerical Methods in Engineering*, Vol. 21, 1985.

Duvigneau, R. and Pelletier, D. On accurate boundary conditions for a shape sensitivity equation method. *International Journal for Numerical Methods in Fluids.*, Vol. 50, 2006.

Etienne, S. and Pelletier, D. General approach to sensitivity analysis of fluid-structure interactions. *Journal of Fluids and Structures*, Vol. 21, No. 2, 2005.

Etienne, S., Hay, A., and Garon, A. Sensitivity analysis of unsteady fluid-structure interaction problems. *45th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA-2007-332, 2007.

Fernandez, M. and Mouchabir, M. Investigation of sensitivity analysis of fluidstructure interaction systems. *the First M.I.T. Conference on Computational Fluid and Solid Mechanics*, 11, 2001.

Ghattas, O. and Li, X. Domain decomposition methods for sensitivity analysis of a nonlinear aeroelasticity problem. *International Journal of Computational Fluid Dynamics*, 11:113–130, 1998.

Haftka, R. and Gurdal, Z. *Elements of Structural Optimization.* Kluwer Academic Publishers, 1992.

Haftka, R.T. and Adelman, H.M. Recent developments in structural sensitivity analysis. *Structural Optimization I*, 1989.

Haftka, R.T. and Grandhi, R.V. Structural shape optimizationa survey. *Computer Methods in Applied Mechanics and Engineering*, 57:91–106, 1986.

Haug, E.J. and Arora, J.S. Design sensitivity analysis of elastic mechanical systems. *Computer Methods in Applied Mechanics and Engineering*, Vol. 15, pp. 35-62, 1978.

Haug, E.J. and Rousselet, B. Design sensitivity analysis in structural mechanics. i. static response variations. *Journal of Structural Mechanics: An International Journal*, 8 (1): 17–41, 1980a.

Haug, E.J. and Rousselet, B. Design sensitivity analysis in structural mechanics. ii. eigenvalue variations. *Journal of Structural Mechanics: An International Journal*, 8 (2):161–186, 1980b.

Haug, E.J. and Rousselet, B. Design sensitivity analysis in structural mechanics. iii. effects of shape variation. *Journal of Structural Mechanics: An International Journal*, 10 (3): 273–310, 1982.

Haug, E.J., Choi, K.K., and Komkov, V. Design sensitivity analysis of structural systems, mathematics in science and engineering. 177, 1986.

Hristova, H., Etienne, S., Pelletier, D., and Borggaard, J. A general sensitivity equation formulation for unsteady laminar flows. *In: 34th AIAA Fluid Dynamics Conference and Exhibit*, 2004.

Issac, J.C. and Kapania, R.K. Aeroelastic sensitivity analysis of wings using automatic differentiation. *AIAA Journal*, 35 (3):519–525, 1997.

Issac, J.C., Kapania, R.K., and Barthelemy, J.M. Sensitivity analysis of flutter responses of a wing to shape and modal paramters. *AIAA Journal*, 33 (10):1983–1986, 1995.

A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3: 233–260, 1988.

F.P. Johnson. Sensorcraft–tomorrow's eyes and ears of the warfighter. *AIAA Modeling and Simulation Technologies Conference and Exhibit*, AIAA-2001-4370, 2001.

Kapania, N.R., Terracciano, K., and Taylor, S. Modeling the fluid flow around airfoils using conformal mapping. 2008.

Kapania, R.K., Bergen, F.D., and Barthelemy, J.M. Sensitivity analysis of a wing aeroelastic response. *Journal of Aircraft*, 30 (4):496–504, 1993.

Liu, S. and Canfield, R.A. Continuum shape sensitivity for nonlinear transient aeroelastic gust response. *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, Colorado.*, AIAA 2011-1971, 2011.

Liu, S. and Canfield, R.A. Continuum shape sensitivity method for fluid flow around an airfoil. *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, Hawaii.*, AIAA 2012-1426, 2012.

Liu, S. and Canfield, R.A. Equivalence of discrete analytic and continuum sensitivity methods for nonlinear differential equations. *Accepted for publication in Structural and Multidisciplinary Optimization*, 2013a.

Liu, S. and Canfield, R.A. Two forms of continuum shape sensitivity method for fluid-structure interaction problems. *Submitted to AIAA Journal*, 2013b.

Liu, S., Wickert, P.D., and Canfield, R.A. Fluid-structure transient gust response sensitivity for a nonlinear joined wing model. *51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Orlando, Florida.*, AIAA 2010-3118, 2010.

E. Livne. Equivalent plate structural modeling for wing shape optimization including transverse shear. *AIAA Journal*, 32 (6), 1994.

D.J. Lucia. The sensorcraft configurations: A non-linear aeroservoelastic challenge for aviation. *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, AIAA 2005-1943, 2005.

Lund, E., Moller, H., and Jakobsen, L. Shape design optimization of steady fluid-structure interaction problems with large displacements. *42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference and Exhibit*, AIAA 2001-1624, 2001.

Lund, E., Moller, H., and Jakobsen, L. Shape design optimization of stationary fluid-structure interaction problems with large displacements and turbulence. *Structural and Multidisciplinary Optimization*, 25, 2003.

Mandar, K., Cross, D., and Canfield, R.A. A discrete adjoint formulation using local continuum sensitivity equations. *AIAA Journal*, (Technical note in preparation), 2014.

J.R.R.A. Martins. The complex-step derivative approximation. *ACM Transactions on Mathematical Software*, 29 (3), 2003.

Morris, C.C., Allison, D.L., Schetz, J.A., and Kapania, R.K. Parametric geometry model for multidisciplinary design optimization of tailless supersonic aircraft. *AIAA Modelling and Simulation Technologies Conference*, AIAA 2012-4850, 2012.

MSC Software Corporation. Md nastran 2010: Dmap programmer's guide. *MD Nastran Documentation*, 2010a.

MSC Software Corporation. Md nastran 2010: User defined services. *MD Nastran Documentation*, 2010b.

NASA. Helios prototype. nasa dryden flight research center, 2004. URL `http://www.nasa.gov/centers/dryden/news/ResearchUpdate/Helios/index.html`. [Online; accessed 27-August-2013].

Olhoff, N. and Taylor, J.E. On structural optimization. *Journal of Applied Mechanics*, 50 (4):1139–1151, 1983.

E. Omar. Sensor integration study for sensorcraft uav mid-term review. *The Boeing Company, Seattle, Washington*, 2003.

Pandey, P. and Bakshi, P. Analytical response sensitivity computation using hybrid finite elements. *Computers and structures*, 71 (5):525–534, 1999.

O. Persson. Distmesh - a simple mesh generator in matlab, 2012. URL `http://persson.berkeley.edu/distmesh/`. [Online; accessed 15-Febrauary-2014].

Ricciardi, A.P., Patil, M.J., Canfield, R.A., and Lindsley, N. Utility of quasi-static gust loads certification methods for novel configurations. *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, Colorado.*, AIAA 2011-2043, 2011.

Sobieszczanski-Sobiesk, J. and Haftka, R.T. Multidisciplinary aerospace design optimization: survey of recent developments. *Structural and Multidisciplinary Optimization*, 14 (1), 1997.

Squire, W. and Trapp, G. Using complex variables to estimate derivatives of real functions. *SIAM Review*, 40(1):110–112, 1998.

Stanley, L. and Stewart, D. *Design sensitivity analysis: computational issues of sensitivity equation methods.* Academic Press, 2002.

Szabo, B. and Babuska, I. *Finite Element Analysis.* John Wiley and Sons, Inc., 1991.

Timoshenko, S. *Theory of Plates and Shells.* McGraw-Hill Book Company, Inc., 1940.

S. Trimble. Darpa re-thinks aircraft acquisition and operations with vulture, 2009. URL `http://www.flightglobal.com/articles/2009/11/30/335535/darpa-re-thinks-aircraft-acquisitionand-operations-with-vulture.html`. [Online; accessed 27-August-2013].

Turgeon, E., Pelletier, D., and Borggaard, J. A continuous sensitivity equation approach to optimal design in mixed convection. AIAA 99-3625, 1999.

Wickert, D.P. Least-squares, continuous sensitivity analysis for nonlinear fluid-structure interaction. *Dissertation, Air Force Institute of Technology.*, 2009.

Wickert, D.P. and Canfield, R.A. Least-squares continuous sensitivity analysis of an example fluid-structure interaction problem. *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Schaumberg, Illinois*, AIAA 2008-1896, 2008.

Wickert, D.P., Canfield, R.A., and Reddy, J.N. Continuous sensitivity analysis of fluid-structure interaction problems using least-squares finite elements. *12th AIAA/ISSMO*

Multidisciplinary Analysis and Optimization Conference, Victoria, British Columbia, Canada., AIAA 2008-5931, 2008.

Wickert, D.P., Canfield, R.A., and Reddy, J.N. Fluid-structure transient gust sensitivity using least-squares continuous sensitivity analysis. *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Palm Springs, California.*, AIAA 2009-2535, 2009.

Yang, R.J. and Botkin, M.E. The relationship between the variational approach and the implicit differentiation approach to shape design sensitivities. *The optimum shape: automated structural design*, pages 61–77, 1986.

Yang, R.J. and Choi, K.K. Accuracy of finite element based shape design sensity analysis. *Journal of Structural Mechanics*, 13 (2):223–239, 1985.

Yang, R.J. and Choi, K.K. Numerical considerations in structural component shape optimization. *Journal of Mechanisms, Transmissions, and Automation in Design*, 107 (3): 334–339, 1986.

Zienkiewicz, O.C. and Zhu, J.Z. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. *International Journal of Numerical Methods in Engineering*, Vol. 33, 1992.

Modeling Considerations

Treating Concentrated Loads with Interface Conditions

Concentrated loads must be treated as a structural interface for local CSA with SGR to be implemented successfully. Consider the clamped-clamped beam in Figure 1

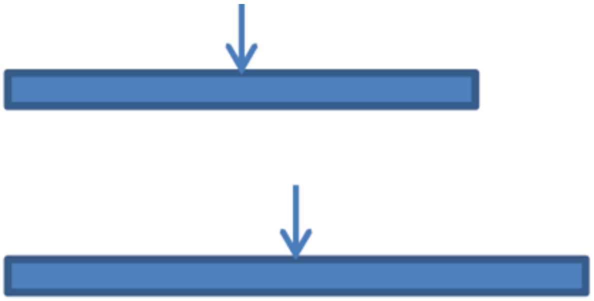


Figure 1: Reviewer 1 Clamped-Clamped Beam Example

where the design variable is the length of the beam and design velocity is given as linear over the length. In this case, the concentrated force at the center of the beam moves as design changes. The concentrated load creates a force discontinuity in the shear distribution and the force distribution, f, is not differentiable. Therefore, it is necessary to treat the discontinuity as an interface condition. This is similar to the discontinuities that occur in the joined beam model and the stiffened plate model. This is also detailed in publications by Liu and Canfield. Liu, S. et al. (2010); Liu, S. and Canfield, R.A. (2013b) For this clamped-clamped beam, let P be the magnitude of the concentrated load, and let $x_{_P}$ be the x-coordinate that locates the concentrated load. In the case for which the concentrated load moves in response to changes in the length of the beam, $x_{_P}$ is parameterized as a function of L (e.g. $x_{_P} = L/2$). Subsequently, the concentrated load has an effectively non-zero f'. In order to capture this effect, the governing equations must first be written in two domains which are adjacent at $x_{_P}$. Assuming a linearly elastic beam with homogeneous properties, the governing equations are

$$\boldsymbol{A}_L\boldsymbol{u}_1 = 0 \qquad \text{on} \qquad \Omega_1 \equiv \{x \in \mathbb{R} \;|\; 0 \leq x \leq x_{_P}\} \tag{1}$$

$$\boldsymbol{A}_L\boldsymbol{u}_2 = 0 \qquad \text{on} \qquad \Omega_2 \equiv \{x \in \mathbb{R} \;|\; x_{_P} \leq x \leq L\} \tag{2}$$

with boundary conditions

$$\boldsymbol{u}_1(0) = \left\{ \begin{array}{c} w_1(0) \\ \psi_1(0) \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\} \tag{3}$$

$$\boldsymbol{u}_2(L) = \left\{ \begin{array}{c} w_2(L) \\ \psi_2(L) \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\} \tag{4}$$

and interface boundary conditions

$$\boldsymbol{u}_1(x_{_P}) = \left\{ \begin{array}{c} w_1(x_{_P}) \\ \psi_1(x_{_P}) \end{array} \right\} = \left\{ \begin{array}{c} w_2(x_{_P}) \\ \psi_2(x_{_P}) \end{array} \right\} = \boldsymbol{u}_2(x_{_P}) \tag{5}$$

$$\sum \boldsymbol{Q}(x_{_P}) = \boldsymbol{Q}_1(x_{_P}) - \boldsymbol{Q}_2(x_{_P}) = \left\{ \begin{array}{c} M_1(x_{_P}) - M_2(x_{_P}) \\ V_1(x_{_P}) - V_2(x_{_P}) \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ -P \end{array} \right\} \tag{6}$$

Here, $w(x)$ is transverse displacement, $\psi(x)$ is the cross-section rotation, $M(x)$ is the internal bending moment, and $V(x)$ is the internal shear force. No assumptions about the shear deformation are made (yet).

The local continuum sensitivity equations can now be derived according to Eqs. (11) through (18) of the manuscript. The design velocity corresponding to the length of the beam is

$$\mathcal{V}(x) = \frac{x}{L} \tag{7}$$

The local continuum sensitivity equations are

$$\boldsymbol{A}_L \boldsymbol{u}_1' = 0 \qquad \text{on} \qquad \Omega_1 \equiv \{x \in \mathbb{R} \ | \ 0 \le x \le x_{\scriptscriptstyle P}\} \tag{8}$$

$$\boldsymbol{A}_L \boldsymbol{u}_2' = 0 \qquad \text{on} \qquad \Omega_2 \equiv \{x \in \mathbb{R} \ | \ x_{\scriptscriptstyle P} \le x \le L\} \tag{9}$$

with boundary conditions

$$\boldsymbol{u}_1'(0) = \left\{ \begin{array}{c} w_1'(0) \\ \psi_1'(0) \end{array} \right\} = \left\{ \begin{array}{c} -w_{1,x}(0)\mathcal{V}(0) \\ -\psi_{1,x}(0)\mathcal{V}(0) \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\} \tag{10}$$

$$\boldsymbol{u}_2'(L) = \left\{ \begin{array}{c} w_2'(L) \\ \psi_2'(L) \end{array} \right\} = \left\{ \begin{array}{c} -w_{2,x}(L)\mathcal{V}(L) \\ -\psi_{2,x}(L)\mathcal{V}(L) \end{array} \right\} = \left\{ \begin{array}{c} -w_{2,x}(L) \\ -\psi_{2,x}(L) \end{array} \right\} \tag{11}$$

and interface boundary conditions

$$\boldsymbol{u}_1'(x_{\scriptscriptstyle P}) = \left\{ \begin{array}{c} w_1'(x_{\scriptscriptstyle P}) \\ \psi_1'(x_{\scriptscriptstyle P}) \end{array} \right\} = \left\{ \begin{array}{c} w_2'(x_{\scriptscriptstyle P}) \\ \psi_2'(x_{\scriptscriptstyle P}) \end{array} \right\} = \boldsymbol{u}_2'(x_{\scriptscriptstyle P}) \tag{12}$$

$$\begin{aligned}\sum \boldsymbol{Q}'(x_P) = \boldsymbol{Q}_1'(x_P) - \boldsymbol{Q}_2'(x_P) &= \left\{ \begin{array}{c} M_1'(x_P) - M_2'(x_P) \\ V_1'(x_P) - V_2'(x_P) \end{array} \right\} \\ &= \left\{ \begin{array}{c} \dot{0} \\ -\dot{P} \end{array} \right\} - \left\{ \begin{array}{c} M_{1,x}(x_P) - M_{2,x}(x_P) \\ V_{1,x}(x_P) - V_{2,x}(x_P) \end{array} \right\} \mathcal{V}(x_P) \qquad (13) \\ &= \left\{ \begin{array}{c} M_{1,x}(x_P) - M_{2,x}(x_P) \\ V_{1,x}(x_P) - V_{2,x}(x_P) \end{array} \right\} \mathcal{V}(x_P)\end{aligned}$$

The boundary conditions of the local continuum sensitivity system represent a fixed boundary condition at $x = 0$, and a free boundary with specified displacement and rotation at $x = L$. Furthermore, where the original analysis demanded continuity of $w(x)$, $\psi(x)$, and $M(x)$ with a shear discontinuity in $V(x)$ caused by the concentrated force, the local continuum sensitivity analysis demands continuity of only $w'(x)$ and $\psi'(x)$ with a discontinuity in the local design derivative of bending moment, $M'(x)$. Essentially, the concentrated force at x_P in the original analysis becomes a concentrated moment at x_P in the local continuum sensitivity analysis. Depending on the beam theory that is used, the local continuum sensitivity analysis may also include a concentrated force at x_P.

If a linear finite element analysis is used to solve the original system, then the spatial derivatives that appear in the boundary conditions can be approximated using spatial gradient reconstruction (SGR). As previously mentioned, this would require SGR to be conducted in the two different domains separately, not allowing any SGR patches to cross over the location of the concentrated load. Treating concentrated forces with interface conditions allows the local CSA method with SGR to be implemented successfully and capture the effects of discontinuities in the force distributions.

Design Parameterization Example

In Section 2.8.1 it is stated that for structural elasticity problems $\boldsymbol{A}'_L$ and $\boldsymbol{B}'_L$ will typically vanish from the right-hand-side of the local CSEs. Similarly, that $\boldsymbol{A}'_{NL}$ and $\boldsymbol{B}'_{NL}$ will tpically vanish for nonlinear problems. However, this is not gauranteed and depends on how the user paramterizes the design. The designer has some control over whether or not these terms vanish based on how he/she chooses to parameterize the geometry and material properties of the structural design. For example, consider a linearly tapered axial bar of length, L. The bar is fixed at one end and free at the other, and an axial end load is applied at the free end. The designer could choose to define the axial stiffness by

$$EA(x) = EA_0(1 - \lambda \frac{x}{L}) \tag{14}$$

where λ is the taper ratio and is constrained to take a value less than or equal to 1, and L is the length of the beam. In this case, the designer has chosen to parameterize the axial stiffness distribution with an explicit dependence on the length of the beam, L. Therefore, the local CSE derived with respect to the length of the beam, L will possess a nonzero A'_L and B'_L which act as additional source terms to the local CSE. Formulating these terms requires the definitions of A_L and B_L to be known. If said definitions are not obtainable, then the designer might wish to parameterize the axial stiffness without normalizing the x-coordinate by L. That is

$$EA(x) = EA_0(1 - \lambda x) \tag{15}$$

This definition requires that $\lambda L \leq 1$, and it assures that A'_L and B'_L will equal zero for the local CSE derived with respect to the length of the beam, L. The choice of parameterization changes the nature of the design space, but for this example, it is clear that both parameterizations can be used to model the same set of linear distributions of axial stiffness. It is not the contention of the author that A'_L and B'_L can always be made to vanish through careful parameterization, but rather a likely possibility.

Adjoint Formulation

Local CSA with SGR was presented in this research using a direct formulation. The direct formulation is good for providing the design derivatives at every finite element node in the discretization. However, oftentimes, the objective and constraint functions of an optimization problem are only dependent on the system response at specific locations in the domain. Therefore, for a problem that has many more design variables than it does constraint functions, an adjoint formulation typically is much more efficient. The direct formulation of local CSA with SGR can be extended to an adjoint formulation. Mandar, K. et al. (2014) present and implement a discrete adjoint formulation using local CSEs.

Assume a displacement constraint

$$\psi = \{z\}^T\{u\} \tag{16}$$

where $\{z\}$ is a virtual load vector and $\{u\}$ is a vector displacements. The local design derivative of the displacement constraint is

$$\psi = \{z\}^T\{u'\} \tag{17}$$

The equilibrium of the discretized local CSEs can be added and multiplied by the adjoint vector, $\{\lambda\}$.

$$\psi = \{z\}^T\{u'\} + \{\lambda\}^T\left([K]\{u'\} - \{F_{Local}\}\right) \tag{18}$$

The adjoint equation, which solves for $\{\lambda\}$, is

$$\{\lambda\} = -[K]^{-T}\{z\} \tag{19}$$

and, therefore, the local design derivative of the displacement constraint becomes

$$\psi' = -\{\lambda\}^T\{F_{Local}\} \tag{20}$$

The benefit of this formulation is that the adjoint equation will typically only need to be solved once for each constraint. For practical problems, the number of design variables typically is significantly larger than the number of constraints, which makes the adjoint formulation more computationally efficient than the direct formulation.

Another apparent advantage of this particular formulation is that the boundary conditions of the adjoint equation are the same as those of the original analysis. Conventional adjoint formulations require separate derivation of the adjoint boundary conditions. This is often a complicated step which is avoided here. This adjoint formulation has only been implemented for static responses of structural models. It has yet been determined if this type of formulation can be derived for transient analysis.