SAND2016-0736J

ECCOMAS Congress 2016
VII European Congress on Computational Methods in Applied Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou, V. Plevris (eds.)
Crete Island, Greece, 5–10 June 2016

A NEW INVERSE PROBLEM STRATEGY BASED ON FORWARD MODEL EVALUATIONS: GRADIENT-BASED OPTIMIZATION WITHOUT ADJOINT SOLVES

Miguel A. Aguiló

Computational Solid Mechanics and Structural Dynamics
Sandia National Laboratories, PO Box 5800, MS 0380 Albuquerque, NM 87185-0380, USA
e-mail: maguilo@sandia.gov

Keywords: Inverse problem, Compliance error minimization, Gradient-based optimization, Newton's method, Nonlinear conjugate gradient, Heat transfer.

Abstract. *This paper presents a new nonlinear programming formulation for the solution of inverse problems. First, a general inverse problem formulation based on the compliance error functional is presented. The proposed error functional enables the computation of the Lagrange multipliers, and thus the first order derivative information, at the expense of just one model evaluation. Therefore, the calculation of the Lagrange multipliers does not require the solution of the computationally intensive adjoint problem. This leads to significant speedups for large-scale, gradient-based inverse problems.*

Second, if second order optimization algorithms are available, Newton's method can be applied to the first order necessary optimality conditions. Newton's method relies on accurate second order derivative information to compute descent directions during optimization. This paper presents two Hessian formulations for inverse problems based on the compliance error functional. The first approach relies on the mathematical properties of the compliance error functional to compute the nonlinear Hessian operator by performing one additional model evaluation. This yields substantial speedups during the Newton iterations. The second approach relies on a linear programming Hessian formulation based on the compliance error functional. This Hessian formulation further speedups the analysis since the first and second order derivative information are computed at the expense of one model evaluation during each optimization iteration.

Third, examples in heat transfer are presented to demonstrate the effectiveness of the compliance error functional. The compliance error minimization formulation is compared to the data misfit formulation for inverse problems. Results will show that the compliance error minimization formulation outperforms the data misfit formulation.

1 INTRODUCTION

This paper investigates the problem of large-scale parameter estimation, i.e. inverse problem, in the context of a particular problem: determining the thermal conductivity properties of a region of interest given an 'observed' temperature field. Specifically, a new inverse problem formulation based on the compliance error functional is presented. A general framework for the application of first and second order optimization algorithms and the compliance error functional is presented, This general inverse problem framework enables application of the compliance error minimization formulation to multiple problem of interests in engineering.

Multiple inverse problem techniques have been proposed in the literature for the solution of inverse problems. For instance, many researchers have applied data misfit error functionals for the solution of inverse problems in multiple physics settings [2, 16, 26, 34]. The objective of data misfit based inverse problems is to characterize the parameter of interest by minimizing the discrepancy between the simulation and the experimental data (target/observed data). An alternative to data misfit functionals is the modified error in the constitutive equation (MECE) functional. The objective of MECE based inverse problems is to characterize the parameter of interests by minimizing the discrepancy in the constitutive equations [9, 10, 19, 20, 21, 22, 23, 24]. This formulation leads to a coupled system of equations that is solved for every new set of materiel parameters. Allix *et al.* [4] performed several numerical studies that showed that the MECE functional improved the convexity of the objective function. Gockenbach *et al.* [12, 13, 14] showed that the energy norm term of the MECE functional is convex for elliptic boundary value problems when full field measurements are available. However, in a subsequent study, Gockenbach [15] showed that inverse problem formulations based on energy norm minimization can lead to inaccurate estimates if the 'observed' data is noisy. Moreover, he showed that the data misfit functional was less sensitive to noisy data than the energy norm functional. Lastly, the virtual field method (VFM) is a recent inverse problem approach developed for extracting constitutive parameters from full-field measurements [32]. The VFM is based on the principle of virtual work and a kinematically admissible virtual field that is computed *a-priori*. The choice of the kinematically admissible virtual field is key in order to improve the performance of the VFM. Thus, most research is focus on improving the predictive capabilities of the kinematically admissible virtual field [17].

The main advantage of the compliance error functional is that it enables the computation of the Lagrange multipliers, and thus the first order derivative information, at the expense of just one model evaluation. Thus, the calculation of the Lagrange multipliers does not require the solution of the computationally intensive adjoint problem. This leads to significant speedups for large-scale, gradient-based inverse problems since the adjoint problem is not necessary to compute the gradient operator at each optimization iteration. Furthermore, the compliance error formulation simplifies implementation in production software libraries since it only relies on forward model evaluations during optimization.

If second order optimization algorithms are available, Newton's method can be applied to the first order necessary optimality conditions. Newton's method requires reliable second order derivative information to compute accurate descent directions during optimization. If the data misfit functional is applied, at least 4 model evaluations are required at every optimization iteration to compute the first and second order derivative information. This paper presents two Hessian formulations based on the compliance error functional that are tailored for second order optimization algorithms. The first Hessian formulation relies on the mathematical properties of the compliance error functional to reduce the number of model evaluations needed to compute descent directions during optimization. This nonlinear Hessian formulation reduces the number of model evaluations from two to one during each Newton iteration. Consequently, substantial speedups are obtained. The second formulation relies on a linear programming Hessian formulation to further speedup the Newton iterations. By using this linear Hessian formulation based on the compliance error functional, the first and second order derivative information can be computed at the expense of one model evaluation at each optimization iteration. Results will show that the compliance error functional leads to substantial speedups over the the data misfit functional.

This paper is organized as follows: Section 2 presents the general inverse problem formulation based on the compliance error and data misfit functionals. Here, a general inverse problem framework based on the compliance error functional is presented for first and second order optimization algorithms. Section 3 presents an inverse problem example in heat transfer to showcase the advantages associated with the compliance error functional. Finally, Section 4 provides conclusions.

2 FORMULATION

2.1 Preliminaries

Let Z, U, and Y be Banach spaces, where both Z and U are reflexive, *i.e.* $z \sim z\, \forall\, z \in Z$ and $u \sim u\, \forall\, u \in U$. Furthermore, let $J\colon U \times Z \to \mathbb{R}$ and $g\colon U \times Z \to Y$. Lets now consider the optimization problem

$$\underset{(u,z)\,\in\, U\times Z}{\text{minimize}} \quad J(u,z) \quad \text{s.t.} \quad g(u,z) = 0, \tag{1}$$

where $u \in U_{ad} \subset U$ and $z \in Z_{ad} \subset Z$. U_{ad} and Z_{ad} denote admissible subsets of the state and control spaces, respectively. If the following conditions are met:

1. $Z_{ad} \subset Z$ is convex, bounded and closed;
2. $U_{ad} \subset U$ is convex, closed, and contains a feasible point; i.e. $g(u,z) = 0$ has a bounded solution operator, $\hat{u}(z)\colon Z \to U$;
3. the mapping $(u,z) \mapsto g(u,z)$ is continuous under weak convergence; and
4. J is sequentially lower semicontinuous;

there exists a solution to the optimization problem defined in Equation 1 [6, 28]. However, the uniqueness of the solution is problem dependent.

First order necessary optimality conditions are necessary to solve Equation 1 using first order optimization algorithms. However, if Newton's method is applied for the solution of Equation 1, first order necessary optimality conditions and second order sufficient conditions are necessary to find an optimal solution. These conditions involve the gradient of the objective function being zero at the optimal solution and the Hessian operator being positive semidefinite at the optimal solution. These conditions can be derived from Lagrangian multiplier theory [33].

2.2 Inverse problem

Let $\Omega \subseteq \mathbb{R}^d$, $d \in \{1,2,3\}$ denote the computational domain with boundary $\partial\Omega$. Lets now define the Lebesgue space $\mathbb{H} = L^2(\Omega;\mathbb{R}^{\mathsf{n}})$ of measurable and square intregrable functions endowed with inner product $\langle \phi, \psi \rangle_{\mathbb{H}} = \int_\Omega \phi\, \psi$ for $\phi, \psi \in \mathbb{H}$ and norm $\|\phi\|_{\mathbb{H}} = \langle \phi, \phi \rangle_{\mathbb{H}}^{1/2}$. Lets also define finite dimensional spaces $\mathbb{U} = \{\text{span}\{\phi^a\}_{a=1}^{A} \,|\, \phi \in \mathbb{H}\} \subset U$, $\mathbb{Z} = \{\text{span}\{\psi^b\}_{b=1}^{B} \,|\, \psi \in \mathbb{H}\} \subset Z$, and $\mathbb{Y} = \{\text{span}\{\chi^c\}_{c=1}^{C} \,|\, \chi \in \mathbb{H}\} \subset Y$. This enables the following finite dimensional approximations for the state, control, and Lagrange multipliers $\mathbf{u} = \sum_a^A \tilde{u}^a \phi^a \,|\, \tilde{u} \in \mathbb{R}$, $\mathbf{z} = \sum_b^B \tilde{z}^b \psi^b \,|\, \tilde{z} \in \mathbb{R}$, and $\mathbf{v} = \sum_c^C \tilde{v}^c \chi^c \,|\, \tilde{v} \in \mathbb{R}$, respectively.

Lets now define a general parameter estimation (inverse) problem as

$$\begin{aligned} &\min_{(\mathbf{u},\mathbf{z})\in\mathbb{U}\times\mathbb{Z}} \quad \mathsf{J}(\mathbf{u},\mathbf{z}) \\ &\text{s.t.} \\ &\quad \mathsf{g}(\mathbf{u},\mathbf{z}) = 0, \end{aligned} \tag{2}$$

where $\mathbf{u}$ and $\mathbf{z}$ respectively denote the state and control variables, $\mathsf{J}(\mathbf{u},\mathbf{z})\colon \mathbb{U}\times\mathbb{Z} \to \mathbb{R}$ denotes the objective function and $\mathsf{g}(\mathbf{u},\mathbf{z})\colon \mathbb{U}\times\mathbb{Z} \to \mathbb{Y}$ denotes the equality constraint (physics model).

The implicit function theorem admits the definition of a solution operator $\mathbf{u}\colon \mathbb{Z} \to \mathbb{U}$ such that $\{(\mathbf{u}(\mathbf{z}),\mathbf{z}) \mid \mathbf{z} \in \mathbb{Z}\} = \{(\mathbf{u},\mathbf{z}) \in \mathbb{U}\times\mathbb{Z} \mid \mathsf{J}(\mathbf{u},\mathbf{z}) = 0\}$. This enables the redefinition of the general inverse problem in Equation 2 as

$$\min_{\mathbf{z}\in\mathbb{Z}} \quad \mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}), \tag{3}$$

where the solution operator $\mathbf{u}(\mathbf{z})$ is obtained by solving $\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0$. This formulation is known as the reduced-space formulation for partial differential equation (PDE) constrained optimization.

Lets also assume that the objective function and equality constraint in Equation 3 are given by

$$\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \frac{\beta}{2}\|\langle \mathbf{u}(\mathbf{z}), \mathbf{A}(\mathbf{z})\mathbf{u}(\mathbf{z})\rangle_{\mathbb{H}} - \langle \hat{\mathbf{u}}, \mathbf{A}(\mathbf{z})\hat{\mathbf{u}}\rangle_{\mathbb{H}}\|_{\mathbb{H}}^2 + \mathsf{R}(\mathbf{z}), \tag{4}$$

and

$$\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \mathbf{A}(\mathbf{z})\mathbf{u} - \mathbf{f} = 0. \tag{5}$$

Here, $\mathbf{A}(\mathbf{z})\colon \mathbb{Z} \to \mathbb{U}\times\mathbb{U}$ is a non-singular, self-adjoint linear operator, $\mathsf{R}(\mathbf{z})\colon \mathbb{Z} \to \mathbb{R}$ is a regularization functional, $\mathbf{f} \in \mathbb{Y}$ is an external force and $\hat{\mathbf{u}} \in \hat{\Omega} \subset \Omega$ denotes measured data. This formulation will be denoted as the compliance error minimization (CEM) formulation for inverse problems.

2.3 First order formulation

2.3.1 Data misfit functional

Lets define the Lagrangian functional $\mathcal{L}\colon \mathbb{U}\times\mathbb{Z}\times\mathbb{Y} \to \mathbb{R}$ for the general inverse problem defined in Equation 3 as

$$\mathcal{L}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = \mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + \langle \mathbf{v}, \mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z})\rangle_{\mathbb{Y}^*,\mathbb{Y}}, \tag{6}$$

where $\mathbf{v}$ denotes the Lagrange multipliers and $\mathbb{Y}^*$ is the dual space of $\mathbb{Y}$. The objective function is given by

$$\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \frac{1}{2}\|\mathbf{u}(\mathbf{z}) - \hat{\mathbf{u}}\|_{\mathbb{H}}^2. \tag{7}$$

and the equality constraint $\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is given by Equation 5. If $\mathbf{z}^\star \in \mathbb{Z}$ is a local solution of Equation 3; then, there exists a set of Lagrange multipliers $\mathbf{v}^\star \in \mathbb{Y}$ such that the first order necessary optimality conditions are satisfied at $\mathbf{z}^\star$.

The first order necessary optimality conditions are given by

$$\mathcal{L}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = \mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + \mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*\mathbf{v} = 0 \tag{8}$$

$$\mathcal{L}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = \mathsf{J}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + \mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*\mathbf{v} = 0, \tag{9}$$

where the subscripts $\mathbf{u}$ and $\mathbf{z}$ respectively denote derivatives with respect to the state and control variables. The Lagrange multipliers are computed by solving

$$\mathbf{v} = -(\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*)^{-1}\mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z}). \tag{10}$$

Substituting Equation 10 into Equation 9 yields a reduced gradient operator of the form

$$\nabla\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \mathsf{J}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + \mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*[-(\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*)^{-1}\mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})]. \tag{11}$$

At each optimization iteration, the following sequence of steps are done to compute the reduced gradient operator and minimize the objective function

1. Solve equality $\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0$ for $\mathbf{u} \in \mathbb{U}$;
2. Solve $\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*\mathbf{v} = -\mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ for $\mathbf{v} \in \mathbb{Y}$;
3. Compute the reduced gradient operator defined in Equation 11;
4. Compute descent direction $\mathbf{s} \in \mathbb{Z}$ and set $\mathbf{z}_{k+1} = \mathbf{z}_k + \gamma\mathbf{s}_k$, $\gamma \in \mathbb{R}$.

This sequence of steps are often necessary to solve an inverse problem based on the data misfit functional [3].

2.3.2 Compliance error functional

Assume that the objective function and equality constraint for the parameter estimation problem defined in Equation 3 are given by Equations 4 and 5, respectively. Then, the first order derivative operators $\mathsf{g}_{\mathbf{u}}(\mathbf{u},\mathbf{z})^*$ and $\mathsf{J}_{\mathbf{u}}(\mathbf{u},\mathbf{z})$ are given by

$$\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^* = \mathbf{A}(\mathbf{z})^* \tag{12}$$

and

$$\mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \alpha\mathbf{A}(\mathbf{z})\mathbf{u}, \tag{13}$$

where

$$\alpha = 2(\langle\mathbf{u}(\mathbf{z}),\mathbf{A}(\mathbf{z})\mathbf{u}(\mathbf{z})\rangle_{\mathbb{H}} - \langle\hat{\mathbf{u}},\mathbf{A}(\mathbf{z})\hat{\mathbf{u}}\rangle_{\mathbb{H}}). \tag{14}$$

Recall that $\mathbf{A}(\mathbf{z})$ is assumed to be a non-singular, self-adjoint linear operator. Thus, an explicit expression for the Lagrange multipliers can be derived by substituting Equations 12 and 37 into Equation 10. This expression is given by

$$\mathbf{v} = -\alpha\beta\mathbf{A}(\mathbf{z})^{-1}(\mathbf{A}(\mathbf{z})\mathbf{u}(\mathbf{z})) = -\alpha\beta\mathbf{I}\mathbf{u}(\mathbf{z}) = -\alpha\beta\mathbf{u}(\mathbf{z}), \tag{15}$$

where $\mathbf{I}$ denotes an identity linear operator. By substituting Equation 15 into Equation 11, the reduced gradient operator is defined as

$$\nabla_{\mathbf{z}}\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \mathsf{J}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) - \alpha\beta\langle\mathbf{u}(\mathbf{z}), \mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\rangle. \tag{16}$$

Notice that the adjoint problem defined in Equation 10 is not solved to compute the Lagrange multipliers, which should speedup the optimization problem.

At each optimization iteration, the following sequence of steps are done to compute the reduced gradient operator and minimize the objective function

1. Solve equality $\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0$ for $\mathbf{u} \in \mathbb{U}$;

2. Compute Lagrange multipliers $\mathbf{v} = -\alpha\beta\mathbf{u}(\mathbf{z})$;

3. Compute the reduced gradient operator given by Equation 16.

4. Compute descent direction $\mathbf{s} \in \mathbb{Z}$ and set $\mathbf{z}_{k+1} = \mathbf{z}_k + \gamma\mathbf{s}_k$, $\gamma \in \mathbb{R}$.

The first order CEM formulation omits the adjoint model evaluations during optimization and thus facilitates implementation and enables significant speedups.

2.4 Second order formulation

2.4.1 Data misfit functional

If second order derivative information is available, Newton's method can be applied to the first order necessary optimality conditions. Then, let $\kappa \in \mathbb{R}_{+}^{*}$ and $\delta\mathbf{z} \in \mathbb{Z}$. If $\mathbf{z}^{\star} \in \mathbb{Z}$ satisfy the first order necessary optimality conditions and

$$\langle\delta\mathbf{z}, \nabla^2\mathsf{J}(\mathbf{u}(\mathbf{z}^{\star})\mathbf{z}^{\star})\delta\mathbf{z}\rangle \geq \kappa\|\delta\mathbf{z}\|_{\mathbb{H}}^2 \;\forall\; \delta\mathbf{z} \in \;\ker \mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}^{\star}),\mathbf{z}^{\star}),$$

the second-order sufficient condition is satisfied at $\mathbf{z}^{\star}$. Furthermore, $\mathbf{z}^{\star}$ is a strict local minimum of Equation 3.

The application of the trial step $\delta\mathbf{z}$ to the nonlinear Hessian operator is given by

$$\nabla^2\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = \mathcal{L}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{u} + \mathcal{L}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{z} + \mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{v}, \tag{17}$$

for $\delta\mathbf{u} \in \mathbb{U}$ and $\delta\mathbf{v} \in \mathbb{Y}$. Notice that $\delta\mathbf{u}$ and $\delta\mathbf{v}$ are necessary to compute the application of the trial step to the nonlinear Hessian operator. Thus, explicit expressions for $\delta\mathbf{u}$ and $\delta\mathbf{v}$ are necessary to effectively calculate Equation 17.

Let $\mathsf{g}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0 \;\forall\; \mathbf{z} \in \mathbb{Z}$. Then, $\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = 0 \;\forall\; (\mathbf{z},\delta\mathbf{z}) \in \mathbb{Z}\times\mathbb{Z}$, where

$$\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = \mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{u} + \mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = 0. \tag{18}$$

Here, $\delta\mathbf{u} \equiv \mathbf{u}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z}$. Solving Equation 18 for $\delta\mathbf{u}$ gives

$$\delta\mathbf{u} = -\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{-1}\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}. \tag{19}$$

Next, an explicit expression is derived for $\delta\mathbf{v}$. By definition, $\mathcal{L}_{\mathbf{u}}(\hat{\mathbf{u}}(\mathbf{z}),\mathbf{z},\mathbf{v}) = 0 \;\forall\; (\mathbf{u},\mathbf{z},\mathbf{v}) \in \mathbb{U}\times\mathbb{Z}\times\mathbb{Y}$; thus, the derivative of $\mathcal{L}_{\mathbf{u}}(\hat{\mathbf{u}}(\mathbf{z}),\mathbf{z},\mathbf{v})$ in the direction of $\delta\mathbf{z}$ gives

$$\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{u} + \mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{z} + \mathcal{L}_{\mathbf{uv}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{v} = 0, \tag{20}$$

$\forall \quad (\mathbf{z}, \delta\mathbf{u}, \delta\mathbf{z}, \delta\mathbf{v}) \in \mathbb{Z} \times \mathbb{U} \times \mathbb{Z} \times \mathbb{Y}$. Solving Equation 20 for $\delta\mathbf{v}$ gives

$$\delta\mathbf{v} = -\mathcal{L}_{\mathbf{uv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})^{-1}[\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{u} + \mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{z}], \tag{21}$$

where

$$\mathcal{L}_{\mathbf{uv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = \mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}), \mathbf{z})^* \tag{22}$$

$$\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = \mathsf{J}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}) + \mathsf{g}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z})^*\mathbf{v} \tag{23}$$

$$\mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = \mathsf{J}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}) + \mathsf{g}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z})^*\mathbf{v}. \tag{24}$$

The following sequence of steps are performed to compute the application of the trial step $\delta\mathbf{z}$ to the reduced Hessian operator at each Newton iteration

1. Solve $\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{u} = -\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{z}$ for $\delta\mathbf{u} \in \mathbb{U}$
2. Solve $\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}), \mathbf{z})^*\delta\mathbf{v} = -[\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{u} + \mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{z}]$ for $\delta\mathbf{v} \in \mathbb{Y}$
3. Compute the application of the trial step to the reduced Hessian operator

$$\nabla^2\mathsf{J}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{z} = \mathcal{L}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{u} + \mathcal{L}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{z} + \mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})\delta\mathbf{v}, \tag{25}$$

where $\mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = \mathsf{g}_{\mathbf{z}}(\hat{\mathbf{u}}(\mathbf{z}), \mathbf{z})^*$.

Notice that the data misfit functional requires two model evaluations per Newton iteration to accurately compute the application of the trial step to the reduced Hessian operator

2.4.2 Compliance error functional: nonlinear programming formulation

Assume that the objective function and equality constraint are given by Equations 4 and 5, respectively. Then, the second order derivative operators $\mathcal{L}_{\mathbf{uv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})$, $\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})$, and $\mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v})$ are given by

$$\mathcal{L}_{\mathbf{uv}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = \mathbf{A}(\mathbf{z})^* \tag{26}$$

$$\mathcal{L}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = 2\beta[\alpha\mathbf{A}(\mathbf{z})\delta\mathbf{u} + 2\mathbf{A}(\mathbf{z})\mathbf{u}(\mathbf{z})\langle\mathbf{u}(\mathbf{z}), \mathbf{A}(\mathbf{z})\delta\mathbf{u}\rangle_{\mathbb{H}}] \tag{27}$$

$$\mathcal{L}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z}, \mathbf{v}) = 2\beta[\alpha(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z})\mathbf{u}(\mathbf{z}) + \gamma\mathbf{A}(\mathbf{z})\mathbf{u}(\mathbf{z})] + (\mathbf{A}(\mathbf{z})^*\mathbf{v})\delta\mathbf{u}, \tag{28}$$

where

$$\gamma = \langle\mathbf{u}(\mathbf{z}), (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z})\mathbf{u}(\mathbf{z})\rangle_{\mathbb{H}} - \langle\hat{\mathbf{u}}, (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z})\hat{\mathbf{u}}\rangle_{\mathbb{H}} \tag{29}$$

Recall that $\mathbf{A}(\mathbf{z})$ is a non-singular, self-adjoint linear operator. This enables the derivation of an explicit expression for $\delta\mathbf{v}$ by substituting Equations 15, 26, 27 and 28 into Equation 21, which is given by

$$\delta\mathbf{v} = -2\beta[\alpha\delta\mathbf{u} + 2\mathbf{u}(\mathbf{z})\langle\mathbf{u}(\mathbf{z}), \mathbf{A}(\mathbf{z})\delta\mathbf{u})\rangle_{\mathbb{H}} + \gamma\mathbf{u}(\mathbf{z})]. \tag{30}$$

Here, $\delta\mathbf{u}$ is given by Equation 19.

To compute the application of the trial step to the reduced Hessian operator, the second order derivative operators $\mathcal{L}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})$, $\mathcal{L}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})$, and $\mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})$ are required. These second order derivative operators are explicitly given by

$$\mathcal{L}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = 2\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z}) - (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}] \tag{31}$$

$$\mathcal{L}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = \beta\gamma[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z}) - (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}] + \mathsf{R}_{\mathbf{zz}}(\mathbf{z}) \tag{32}$$

$$\begin{aligned}\mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v}) = -2\beta[\alpha(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\delta\mathbf{u} + 2(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z})\langle\mathbf{u}(\mathbf{z}),\mathbf{A}(\mathbf{z})\delta\mathbf{u}\rangle_{\mathbb{H}} \\ + \gamma(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z})].\end{aligned} \tag{33}$$

Here, Equations 15 and 30 were used to derived and simplified Equations 31 and 33.

The application of the trial step to the reduced Hessian operator is then obtained by substituting Equations 31-33 into Equation 25. After some simplifications, the application of the trial step to the reduced Hessian operator is given by

$$\begin{aligned}\nabla^2\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = &-2\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z}) + (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}]\langle\mathbf{u}(\mathbf{z}),\mathbf{A}(\mathbf{z})\delta\mathbf{u}\rangle \\ &- \beta\gamma[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\mathbf{u}(\mathbf{z}) + (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}] + \mathsf{R}_{\mathbf{zz}}(\mathbf{z})\delta\mathbf{z} \\ &- 2\beta\alpha(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u}(\mathbf{z}))\delta\mathbf{u}.\end{aligned} \tag{34}$$

Thus, the following sequence of steps are applied to compute the application of the trial step $\delta\mathbf{z}$ to the reduced Hessian operator at each Newton iteration

1. Solve $\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{u} = -\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}$ for $\delta\mathbf{u} \in \mathbb{U}$
2. Compute $\nabla^2\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}$ as defined by Equation 34.

The proposed compliance error minimization formulation enables the calculation of the second order derivative information at the expense of one model evaluation per Newton iteration. Contrary, the data misfit formulation presented in Section 2.4.1 requires two FEM evaluations per Newton iteration. Thus, significant computational savings are attained during optimization.

2.4.3 Compliance error functional: linear programming formulation

To circumvent the computational demands associated with the calculation of second order derivative information, quasi-Newton methods have been innovated to effectively approximate this information during optimization [6, 8, 28]. These methods have been successfully applied to many applications [29, 30]. In this work, a different approach to quasi-Newton methods is explored by applying a linear Hessian formulation based on the compliance error functional.

Lets define the application of the trial step to the reduced linear Hessian operator as

$$\nabla^2 J(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} \equiv \mathcal{L}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{z} = [J_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + g_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*\mathbf{v}]\delta\mathbf{z}. \tag{35}$$

Notice that the derivative operators $\mathcal{L}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{u}$ and $\mathcal{L}_{\mathbf{zv}}(\mathbf{u}(\mathbf{z}),\mathbf{z},\mathbf{v})\delta\mathbf{v}$ are omitted in Equation 35 since these nonlinear terms vanishe. Thus, the solution of Equation 19 is not required to compute the application of the trail step to the reduced linear Hessian operator during the Newton iterations. Results will demonstrate that the linear Hessian formulation leads to significant speedups while preserving solution accuracy.

3 EXAMPLE IN HEAT TRANSFER

The Intrepid PDE discretization package from Trilinos [18] was used to build the finite element models. The direct solver routine from MATLAB [27] scientific package were used in the numerical studies performed herein. The optimization algorithms in this work were implemented in C++ and used to generate the results presented herein. Readers are encourage to explore any optimization library of their preference [1, 11, 18, 37]. Finally, all calculations were performed on a Linux workstation with a 2.93 GHz Intel(R) Core Xeon(R) processor and 24 GB of RAM.

To synthesize the 'observed' temperature field, a finer grid with 80,000 triangles was used to generate the experimental temperature field. The experimental temperature field was then projected onto a computational grid 20,000. This was done to avoid using the same computational mesh used to generate the 'observed' temperature field during optimization. Different levels of random Gaussian noise were also considered, $\Delta \sim N(0, \widehat{\sigma})$, to test the tolerance of the proposed formulation to corrupt data. A Gaussian distributed set of random numbers has 65%, 95%, and 99.7% certainty of respectively being within one, two, and three standard deviations from the mean. Lets thus assume that the Gaussian random noise generated to test the compliance error minimization formulation is 95% certain of being within $\theta\%$ of the actual data. Then, the perturbation applied to the 'observed' temperature field is scaled from interval $(-2\widehat{\sigma}, 2\widehat{\sigma})$ to $(-\theta\%, \theta\%)$. This produces a perturbation parameter of the form $\varepsilon^{\theta} = \frac{1}{2}(\frac{\theta}{100})$ [5]. Therefore, the corrupt data can then be generated as follows

$$\hat{\mathbf{u}}_i^{\theta} = \mathbf{u}_i(1 + \varepsilon_i^{\theta}), \text{ for } i = 1, \ldots, n_{\mathbf{u}} \tag{36}$$

where $\theta \in \Theta = \{1\%, 3\%, 5\%\}$ and $n_{\mathbf{u}}$ denotes the number of states.

Finally, the compliance error minimization formulation was compared to the data misfit formulation. The objective was to highlight the effectiveness of the CEM formulation against common formulation strategies for inverse problems. For completeness, the corresponding first and second order derivative operators for the data misfit functional are defined herein

$$\mathsf{J}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}), \mathbf{z}) = \beta(\mathbf{u}(\mathbf{z}) - \hat{\mathbf{u}}) \tag{37}$$

$$\mathsf{J}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{u} = \beta\delta\mathbf{u} \tag{38}$$

$$\mathsf{J}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}), \mathbf{z}) = \mathsf{J}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{u} = \mathsf{J}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{z} = \mathsf{J}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}), \mathbf{z})\delta\mathbf{z} = \mathbf{0} \tag{39}$$

3.1 Problem formulation

Lets consider the following parameter estimation (inverse) problem in heat transfer

$$\begin{aligned} &\min_{\mathbf{z}\in\hat{\mathbb{Z}}} \quad \frac{\beta}{2}\|\langle\mathbf{u}, \mathbf{A}(\mathbf{z})\mathbf{u}\rangle_{\mathbb{H}} - \langle\hat{\mathbf{u}}, \mathbf{A}(\mathbf{z})\hat{\mathbf{u}}\rangle_{\mathbb{H}}\|_{\mathbb{H}}^2 + \mathsf{R}(\mathbf{z}) \\ &\text{s.t.} \\ &\quad \mathbf{A}(\mathbf{z})\mathbf{u} = \mathbf{f}(\mathbf{x}) \quad \text{in} \quad \Omega \\ &\qquad\quad \mathbf{u} = 0 \quad\ \ \text{on} \quad \partial\Omega, \end{aligned} \tag{40}$$

where $\hat{\mathbb{Z}} = \{\mathbf{z} \in \mathbb{Z} : \mathscr{L} \leq \mathbf{z} \leq \mathscr{U}\}$. Here, $\mathscr{L}$ denotes the control lower bounds and $\mathscr{U}$ denotes the control upper bounds. For a steady-state heat equation, $\mathbf{z}$ is the coefficient of thermal conductivity, $\mathbf{u}$ is the temperature field, $\hat{\mathbf{u}} \in \Omega^m \subseteq \Omega$ are the temperature measurements. $\mathbf{f}(\mathbf{x})$ is a heat source given by

$$\mathbf{f}(\mathbf{x}) = A * \sin(\omega \mathbf{x}) \cos(\omega \mathbf{x}), \tag{41}$$

where $A \in \mathbb{R}$ is a given amplitude, $\omega \in \mathbb{R}$ denotes angular frequency and $\mathbf{x} \in \Omega$ denotes a position is space. $\mathbf{A}(\mathbf{z})\colon \mathbb{Z} \to \mathbb{U} \times \mathbb{U}$ is a linear operator that depends on the coefficient of thermal conductivity and $\beta \in \mathbb{R}_+$ denotes a penalty parameter. The finite dimensional approximations for the state, control, and Lagrange multipliers were previously defined in Section 2.2.

3.1.1 Regularization

The regularization functional $\mathsf{R}(\mathbf{z})\colon \mathbb{Z} \to \mathbb{R}$ in Equation 40 is given by

$$\mathsf{R}(\mathbf{z}) = \frac{\zeta}{2\theta} \int_\Omega \left(\langle \nabla \mathbf{z}, \nabla \mathbf{z} \rangle_{\mathbb{H}} + \nu \right)^\theta d\Omega. \tag{42}$$

This expression gives the flexibility to employ total variation regularization when ($\theta = 1, \nu = 0$) and a modified form of the total variation regularization when ($\theta = 1/2$, $0 < \nu \leq 1$). The parameter $0 < \zeta \leq 1$ is a penalty coefficient. In this work, the modified form of the total variation regularization functional was preferred over Tikhonov regularization due to its ability to capture sharp discontinuities in inverse problems settings. The interested reader is referred to [36] to explore other regularization methodologies that could be employed for inverse problems.

3.1.2 Helmholtz filter

The aim of this section is to present the Helmholtz filter as an alternative to regularization methods, e.g. Equation 42. Regularization functionals are often (if not always) explicitly incorporated into objective functions to solve ill-posed inverse problems. The function of regularization functionals is to penalize the objective function and enhance the smoothness of the control field. Thus, regularization functionals aim to bound the objective function and prevent undesired data overfitting.

Instead of just applying a regularization functional, e.g. Equation 42, to solve the inverse problem in Equation 40, a Helmholtz PDE filter is also employed to filter the optimal control computed using the CEM methodology. Helmholtz PDE filters have been successfully used in topology optimization to avoid numerical artifacts and to enhance the design's smoothness and boundary description [25]. The filtered control is computed by applying a convolution operator to the optimal control. However, instead of explicitly defining the convolution integral, the filtered control can be defined implicitly as the solution of the following Helmholtz PDE

$$-r^2 \nabla^2 \tilde{\mathbf{z}} + \tilde{\mathbf{z}} = \mathbf{z}, \tag{43}$$

with Neumann boundary conditions

$$\frac{\partial \tilde{\mathbf{z}}}{\partial \mathbf{n}} = 0. \tag{44}$$

The parameter r denotes a given filter length scale and $\tilde{\mathbf{z}}$ is the filtered control, which was set to $1/3 \times 10^3$ for this study. As $\mathbf{x}/r \to \infty$, the filtering effect on the optimal control is reduced. Thus, larger values of r minimize the filtering effect on the optimal control.

In this work, the discretized Helmholtz PDE is solved after optimization. Contrary, we could have eliminated the regularization functional from Equation 40 and solved the discretized Helmholtz PDE every time a new set of trial controls was computed by the optimization algorithm. However, this would have increased the computational time. Furthermore, preliminary results performed as part of this study did not justify this approach. The quality of the solution obtained by applying the filter during optimization did not substantially compared to the solution obtained by applying the filter after optimization.

3.2 Optimality conditions

The Lagrangian functional $\mathcal{L}\colon \mathbb{U} \times \hat{\mathbb{Z}} \times \mathbb{Y} \rightarrow \mathbb{R}$ for the inverse problem defined in Equation 40 is given by

$$\mathcal{L}(\mathbf{u},\mathbf{z},\mathbf{v}) = \frac{\beta}{2}\|\langle \mathbf{u}, \mathbf{A}(\mathbf{z})\mathbf{u}\rangle_{\mathbb{H}} - \langle \hat{\mathbf{u}}, \mathbf{A}(\mathbf{z})\hat{\mathbf{u}}\rangle_{\mathbb{H}}\|^2_{\mathbb{H}} + \mathsf{R}(\mathbf{z}) + \langle \mathbf{v}, \mathbf{A}(\mathbf{z})\mathbf{u} - \mathbf{f}\rangle_{\mathbb{Y}^*,\mathbb{Y}}. \tag{45}$$

The first order necessary optimality conditions for Equation 45 are given by

$$\mathcal{L}_{\mathbf{u}}(\mathbf{u},\mathbf{z},\mathbf{v}) = \alpha\beta\mathbf{A}(\mathbf{z})\mathbf{u} + \mathbf{A}(\mathbf{z})\mathbf{v} = 0 \tag{46}$$

$$\mathcal{L}_{\mathbf{z}}(\mathbf{u},\mathbf{z},\mathbf{v}) = \alpha\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{u} + (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}] + \mathsf{R}_{\mathbf{z}}(\mathbf{z}) + (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{v} = 0, \tag{47}$$

where α is given by Equation 14. The first order derivative operator $R_{\mathbf{z}}(\mathbf{z})\colon \hat{\mathbb{Z}} \rightarrow \mathbb{Z}$ is given by

$$R_{\mathbf{z}}(\mathbf{z}) = \frac{\zeta}{2}(\theta - 1)(\langle \nabla\mathbf{z}, \nabla\mathbf{z}\rangle_{\mathbb{H}} + \nu^2)^{\theta-1}\mathbf{B}\,\mathbf{z}, \tag{48}$$

where

$$\mathbf{B} = \int_{\Omega} \nabla\psi\nabla\psi\; d\Omega. \tag{49}$$

If $\mathbf{A}(\mathbf{z})$ is a non-singular, self-adjoint linear operator, the Lagrange multipliers for an inverse problem in heat transfer are given by Equation 15. Substituting Equation 15 into Equation 47 yields the following reduced gradient operator

$$\nabla\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = -\alpha\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{u} + (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}] + \mathsf{R}_{\mathbf{z}}(\mathbf{z}). \tag{50}$$

Notice, as previously demonstrated in Section 2, that the reduced gradient operator is computed without solving the adjoint system of equations defined in Equation 10.

If second order derivative information is available, Newton's method can be applied to the first order necessary optimality conditions. The following derivative operators are then required to compute the application of the trial step to the reduced nonlinear Hessian operator

$$\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z} = (\mathbf{A}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z})\mathbf{u} \tag{51}$$

$$\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{u} = \mathbf{A}(\mathbf{z})\delta\mathbf{u} \tag{52}$$

$$\mathsf{g}_{\mathbf{z}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^*\delta\mathbf{z} = (\mathbf{A}_{\mathbf{z}}(\mathbf{z})^*\mathbf{v})\mathbf{u} \tag{53}$$

$$\mathsf{g}_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\delta\mathbf{u}=\mathbf{A}(\mathbf{z})^{*}\mathbf{v} \tag{54}$$

$$\mathsf{g}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\delta\mathbf{u}=\mathbf{0} \tag{55}$$

$$\mathsf{g}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\delta\mathbf{z}=(\mathbf{A}_{\mathbf{z}}(\mathbf{z})^{*}\delta\mathbf{z})\mathbf{v} \tag{56}$$

$$\mathsf{g}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\delta\mathbf{u}=\mathbf{0} \tag{57}$$

$$\mathsf{g}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z})^{*}\delta\mathbf{u}=(\mathbf{A}_{\mathbf{z}}(\mathbf{z})^{*}\mathbf{v})\delta\mathbf{u} \tag{58}$$

$$\mathsf{J}_{\mathbf{uu}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{u}=2\beta[\alpha\mathbf{A}(\mathbf{z})\delta\mathbf{u}+2\langle\mathbf{u},\mathbf{A}(\mathbf{z})\delta\mathbf{u}\rangle_{\mathbb{H}}\mathbf{A}(\mathbf{z})\mathbf{u}] \tag{59}$$

$$\mathsf{J}_{\mathbf{uz}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}=2\beta[\alpha(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\delta\mathbf{z})\mathbf{u}+\gamma\mathbf{A}(\mathbf{z})\mathbf{u}] \tag{60}$$

$$\mathsf{J}_{\mathbf{zz}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}=\gamma\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{u}-(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}]+\mathsf{R}_{\mathbf{zz}}(\mathbf{z})\delta\mathbf{z} \tag{61}$$

$$\mathsf{J}_{\mathbf{zu}}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{u}=2\beta[\alpha(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\delta\mathbf{u}+((\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{u}-(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}})\langle\mathbf{u},\mathbf{A}(\mathbf{z})\delta\mathbf{u}\rangle_{\mathbb{H}}], \tag{62}$$

where γ is given by Equation 29 and the Lagrange multipliers $\mathbf{v}$ are given by Equation 15. Finally, the second order derivative operator $\mathsf{R}_{\mathbf{zz}}(\mathbf{z})$ is given by

$$\mathsf{R}_{\mathbf{zz}}(\mathbf{z})=\frac{\zeta}{2}((\theta-1)(\langle\nabla\mathbf{z},\nabla\mathbf{z}\rangle_{\mathbb{H}}+\nu^{2})^{\theta-2})\langle\nabla\mathbf{z},\nabla\mathbf{z}\rangle_{\mathbb{H}}\mathbf{B}+(\langle\nabla\mathbf{z},\nabla\mathbf{z}\rangle_{\mathbb{H}}+\nu^{2})^{\theta-1}\mathbf{B}. \tag{63}$$

Substituting Equations 51-62 into Equation 25 yields the application of the trial step to the reduced nonlinear Hessian operator, defined in Equation 34. However, if the reduced linear Hessian formulation based on the compliance error functional is applied, the application of the trial step to the reduced linear Hessian operator is given by

$$\nabla^{2}\mathsf{J}(\mathbf{u}(\mathbf{z}),\mathbf{z})\delta\mathbf{z}=\gamma\beta[(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\mathbf{u})\mathbf{u}-(\mathbf{A}_{\mathbf{z}}(\mathbf{z})\hat{\mathbf{u}})\hat{\mathbf{u}}]+\mathsf{R}_{\mathbf{zz}}(\mathbf{z})\delta\mathbf{z}, \tag{64}$$

where $\mathsf{R}_{\mathbf{zz}}(\mathbf{z})$ is defined in Equation 63.

Finally, the discretized Helmholtz equation is given by

$$r\mathbf{B}\hat{\mathbf{z}}+\mathbf{M}\hat{\mathbf{z}}=\mathbf{z}, \tag{65}$$

where

$$\mathbf{M}=\int_{\Omega}\psi\psi\,d\Omega. \tag{66}$$

and $\mathbf{B}$ is defined by Equation 49.

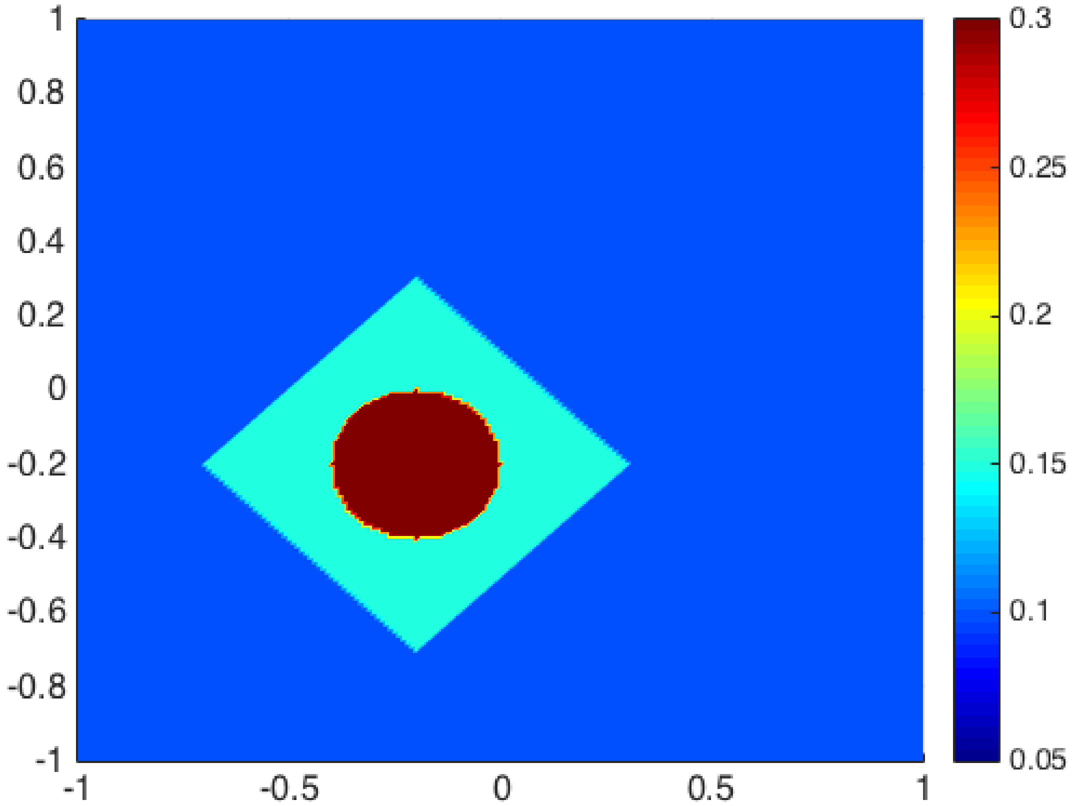


Figure 1: Target thermal conductivity field.

3.3 Results: full-field temperature field

The compliance error minimization and the data misfit formulations were applied to the inverse problem defined in Equation 40. The amplitude for the heat source was set to 1×10^2 and the angular frequency was set to 4π and 16π for the numerical studies performed with the data misfit and compliance error functionals, respectively. The target thermal conductivity field is shown in Figure 1. The thermal conductivity coefficients lower and upper bounds were respectively set to 0.01 and 1.0 during optimization. The regularization parameter ζ was set to 1.0 for all the numerical studies performed herein. To quantify the computational efficacy of the compliance error functional, the corresponding speedups, S, are computed.

3.3.1 First order formulation

The Perry-Shanno nonlinear conjugate gradient was used to solve the inverse problem in heat transfer [31, 35]. The optimization algorithm stopped when $\mathsf{J}(\mathbf{u}(\mathbf{z}), \mathbf{z})_k$, $\|\nabla_{\mathbf{z}}\mathsf{J}(\mathbf{u}(\mathbf{z}), \mathbf{z})_k\|$ or $\|\mathsf{s}_k\|$ was below a predefined tolerance of 1×10^{-4}. A backtracking line search with cubic step interpolation was applied to enhance the global convergence capability of the nonlinear conjugate gradient algorithm. The line search contraction parameter was set to 0.5 and the step lower bound was set to 1×10^{-5}. Furthermore, the maximum number of line search iterations was set to 5.

The regularization parameters used for the numerical studies based on the data misfit func-

Noise	DMF	CEM	S
0%	132.7	1.08	122.87
1%	132.25	1.15	119.35
3%	103.18	1.07	96.43
5%	63.2	1.19	53.11

Table 1: CPU time (seconds) and corresponding speedups obtained using the first order formulation strategy based on the compliance error minimization formulation. Here DMF denotes data misfit functional, CEM denotes compliance error minimization and S denotes speedup.

tional were $(\zeta, \nu) = (5 \times 10^{-4}, 1 \times 10^{-4})$ for $\theta = \{0\%, 1\%\}$ and $(\zeta, \nu) = (5 \times 10^{-3}, 1 \times 10^{-4})$ for $\theta = \{3\%, 5\%\}$. Contrary, the regularization parameters used for all the numerical studies based on the CEM formulation were $(\zeta, \nu) = (1 \times 10^{-8}, 1 \times 10^{-8})$.

Table 1 shows the central processing unit (CPU) time obtained for the numerical studies based on first order formulation strategies. Notice that the compliance error formulation produced noticeable speedups over the data misfit formulation (DMF), regardless of the noise level. Clearly, the CEM strategy computationally outperformed the data misfit formulation strategy. Figure 2 shows that the CEM strategy required less than 10 optimization iterations to converged to an optimal solution in all the numerical studies. Contrary, the data misfit formulation strategy needed over 1000 iterations to meet one of the required convergence criterion. Figure 2 also shows the objective function values computed for the first order formulation strategies. Here, the reader can appreciate that the compliance error functional lead to faster convergence rates. Contrary, the data misfit functional displayed produced slower convergence rates for all the noise levels.

Figure 3 shows the optimal thermal conductivity field computed using the data misfit first order formulation strategy. Figure 3 shows the optimal thermal conductivity field was computed for all the noise level. However, the optimal solution was sensitive to the noise level, which was expected. Figure 4 shows the optimal thermal conductivity field computed using the CEM first order formulation strategy. The compliance error minimization formulation produced optimal results regardless of the noise level. However, the CEM strategy produced non-optimal thermal conductivity fields were the Dirichlet and Neumann boundary conditions were applied. More research is needed to understand why these inaccuracies are obtained in areas were Dirichlet or Neumann boundary conditions are applied.

Finally, it was noticed in this case study that the compliance error functional was less sensitive to corruption in the experimental data. Although the Helmholtz filter could also be employed with the data misfit formulation strategy, the computational efficacy of the CEM strategy was less impacted/hindered by the corruption in the data. However, more research is required to fully understand the outcomes produce by this case study. Regardless, these results are encouraging.

3.3.2 Second order formulation

A dogleg trust region inexact Newton algorithm [7] was used to solve the inverse problem defined in Equation 40 when second order derivative information was available. The optimization algorithm once more stopped when one of the following stopping criterion was satisfied: $\mathsf{J}(\mathbf{u}(\mathbf{z}), \mathbf{z})_k < 1 \times 10^{-4}$, $\|\nabla_{\mathbf{z}} \mathsf{J}(\mathbf{u}(\mathbf{z}), \mathbf{z})_k\| < 1 \times 10^{-4}$ or $\|\mathrm{s}_k\| < 1 \times 10^{-4}$. The trust region

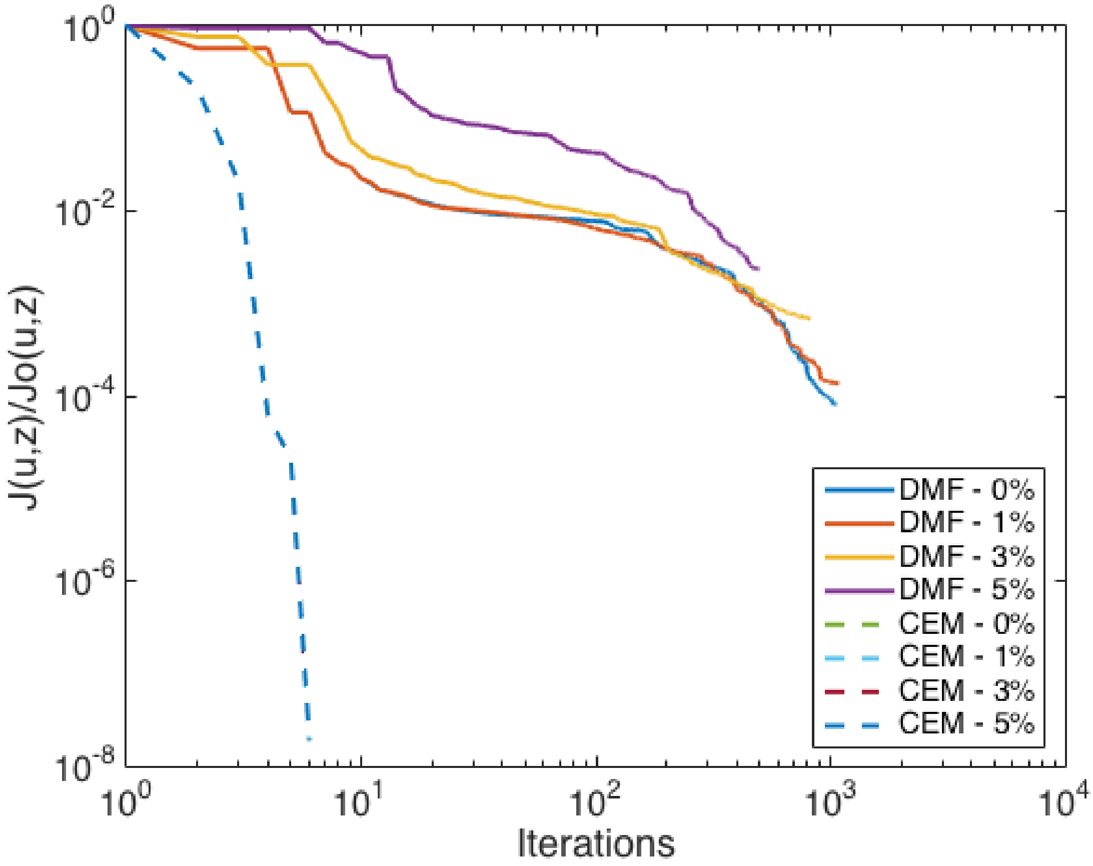
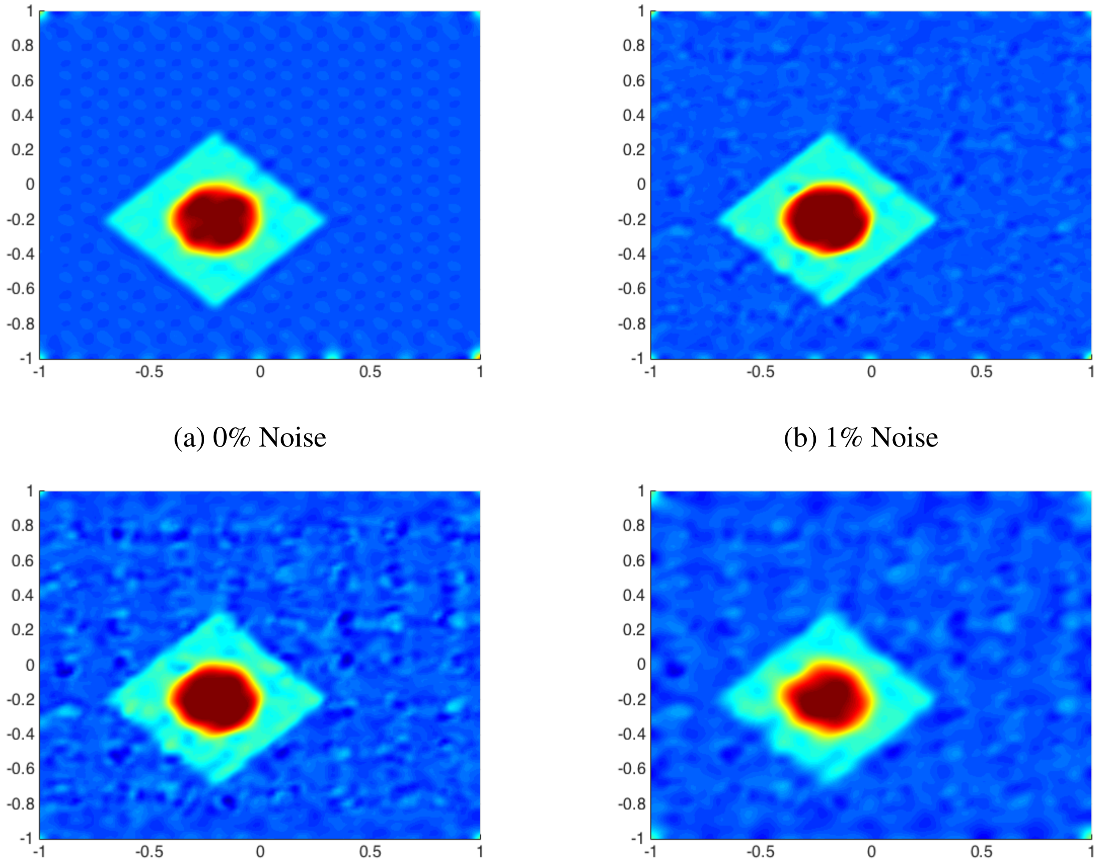


Figure 2: Objective function values computed using the first order formulation strategies.



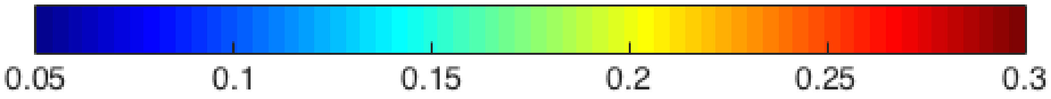
(a) 0% Noise

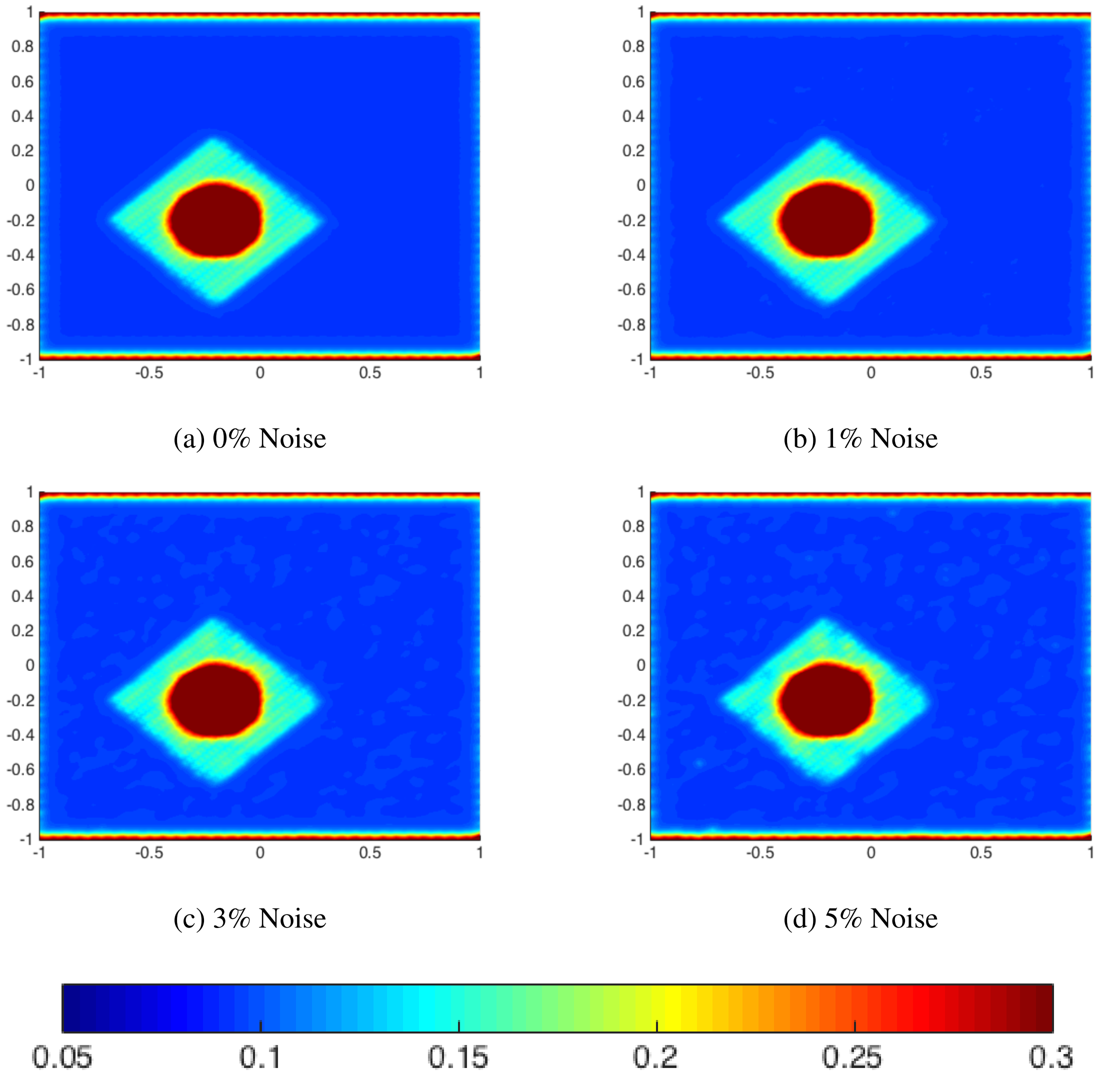
(b) 1% Noise

(c) 3% Noise

(d) 5% Noise

Figure 3: Optimal thermal conductivity field computed using the first order data misfit formulation strategy and the Perry-Shanno nonlinear conjugate gradient algorithm.





(a) 0% Noise

(b) 1% Noise

(c) 3% Noise

(d) 5% Noise

Figure 4: Optimal thermal conductivity field computed using the first order compliance error minimization formulation and the Perry-Shanno nonlinear conjugate gradient algorithm.

Noise	DMF	CEM-NLP	CEM-LP	S-NLP	S-LP
0%	123.64	8.18	5.37	15.11	23.02
1%	192.42	7.98	4.61	24.11	41.74
3%	211.29	8.06	6.07	26.21	34.81
5%	91.76	8.08	5.06	11.36	18.13

Table 2: CPU time (seconds) and corresponding speedups obtained with the second order compliance error minimization formulations. Here, NLP and LP respectively denote nonlinear and linear programming Hessian formulation and S denotes speedup.

contraction and expansion parameters were respectively set to 0.5 and 2. The maximum number of trust region sub-problem iterations was set to 5 and the minimum ratio between the actual and predicted reduction was set to 0.2.

The regularization parameters used for the numerical studies based on the data misfit formulation were $(\zeta, \nu) = (1\times10^{-3}, 1\times10^{-3})$ for $\theta = \{0\%, 1\%, 3\%\}$ and $(\zeta, \nu) = (5\times10^{-3}, 1\times10^{-3})$ for $\theta = 5\%$. Once more, the regularization parameters used for all the numerical studies based on the compliance error minimization formulation were $(\zeta, \nu) = (1 \times 10^{-8}, 1 \times 10^{-8})$.

Table 2 shows the CPU times for the numerical studies based on the second order formulations. The results on Table 2 show that the compliance error minimization formulation once more produced significant speedups. However, the CPU times gathered for the first order compliance error minimization formulations were larger than the CPU times gathered for the second order CEM formulation. Therefore, the CEM formulation based on second order derivative information did not outperform the first order CEM formulation. However, these results are specific to the parameter estimation problem in heat transfer considered herein. Thus, more studies are necessary to further understand the benefits of the compliance error minimization formulation based on second order derivative information. Lets recall that an effective preconditioning strategy could have been used to improve the performance of the Newton algorithm, which was outside the scope of the present study.

Figure 5 displays the objective function values computed using the second order formulations. Notice that the compliance error minimization formulations once more needed less than 10 optimization iterations to converged to an optimal solution. Contrary, the data misfit formulation required over 50 optimization iterations (in some cases over 100 iterations) to meet one of the convergence criterion. Why the CPU times gathered for the second order data misfit formulation are greater (for 3 out of the 4 noise levels) than the CPU times gathered for the first order data misfit formulations? This results can be counterintuitive to the reader since the numerical studies based on the first order data misfit formulation required more optimization iterations to converge. The reader should recall that second order optimization algorithms require at least 4 model evaluations (2 for the gradient and 2 for the Hessian calculation) per optimization iteration to compute a trial control. Furthermore, in every Newton iteration, 2 model evaluations are necessary to compute a descent direction. Hence, regardless of the fact that the trust region Newton algorithm converged in less optimization iterations, the number of model evaluations will always dictate the CPU times.

The same argument can be applied to the second order CEM formulation based on a nonlinear Hessian formulation. At every optimization iteration, at least 2 model evaluations (1 for the gradient and 1 for the Hessian calculation) are performed per iteration to compute a new set of control parameters. This fact can explained why the CPU times gathered for the CEM

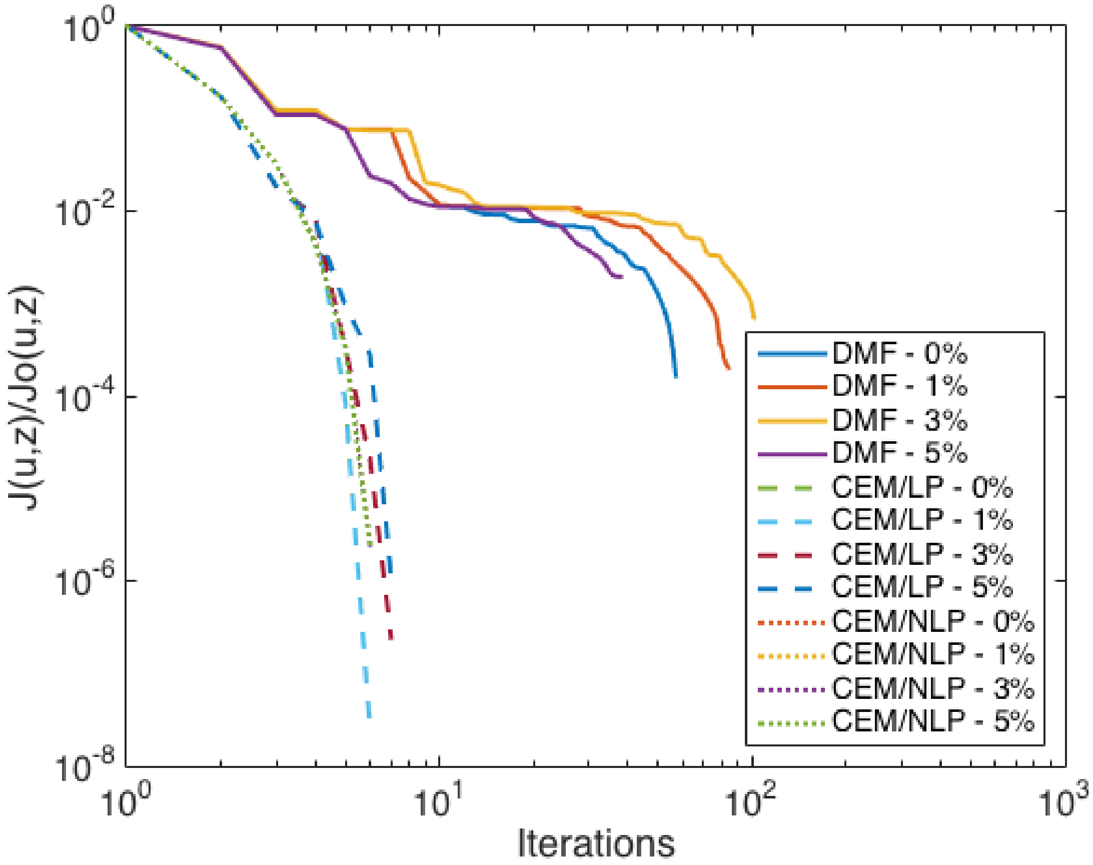
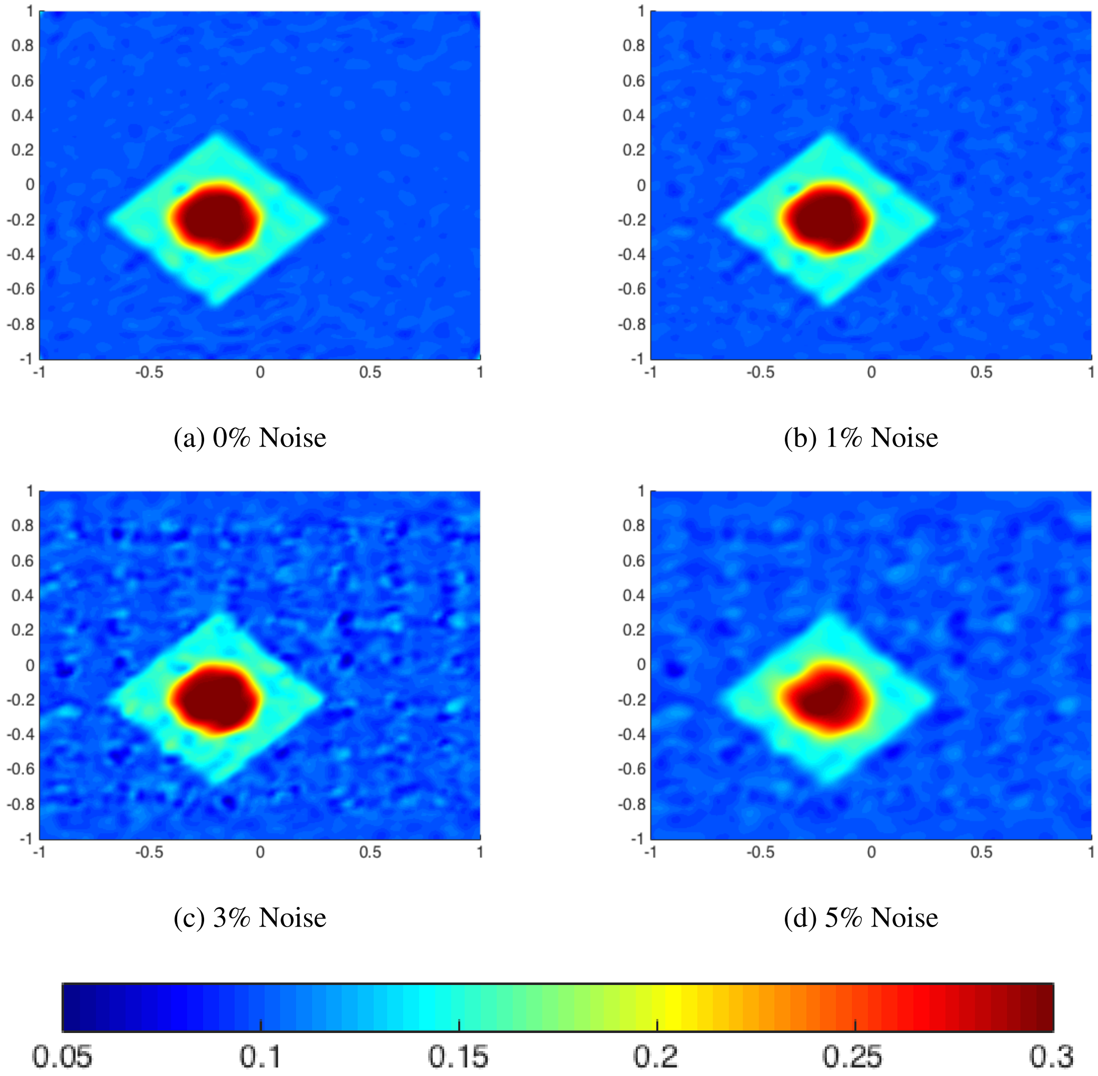


Figure 5: Objective function values computed using the second order formulation strategies.

formulation based on a nonlinear Hessian were greater than the CPU times gathered for the first order CEM strategy. Next, lets consider the CEM formulation based on a linear Hessian formulation. The second order compliance error minimization formulation based on a linear Hessian formulation only needs one model evaluation per optimization iteration. Therefore, why the CPU times are not closer/similar to the CPU times observed for the first order CEM formulation? In all the numerical studies done herein, the optimization algorithm required several trust region sub-problem iterations to compute an optimal trial control that met the ratio between the actual and the predicted reduction. Contrary, the first order CEM formulation did not need additional line search iterations to compute a descent direction that yielded an optimal/feasible trial control at each optimization iteration. Therefore, less model evaluations were required during optimization; thus, reducing computational effort.

Figure 5 also shows the convergence rates for the second order formulations. Notice that the CEM formulations displayed faster convergence rates than the second order data misfit formulation, similar to the results shown in Section 3.3.1. However, the convergence rates for the second order data misfit formulation displayed faster convergence rates near the optimal/feasible point. This was expected since Newton algorithms should converge quadratically near the optimal/feasible point.

Figure 6 shows the optimal thermal conductivity field computed using the second order data misfit formulation for all the noise levels. Results demonstrate that the second order data misfit formulation produced an optimal thermal conductivity field. However, results also show that the



(a) 0% Noise

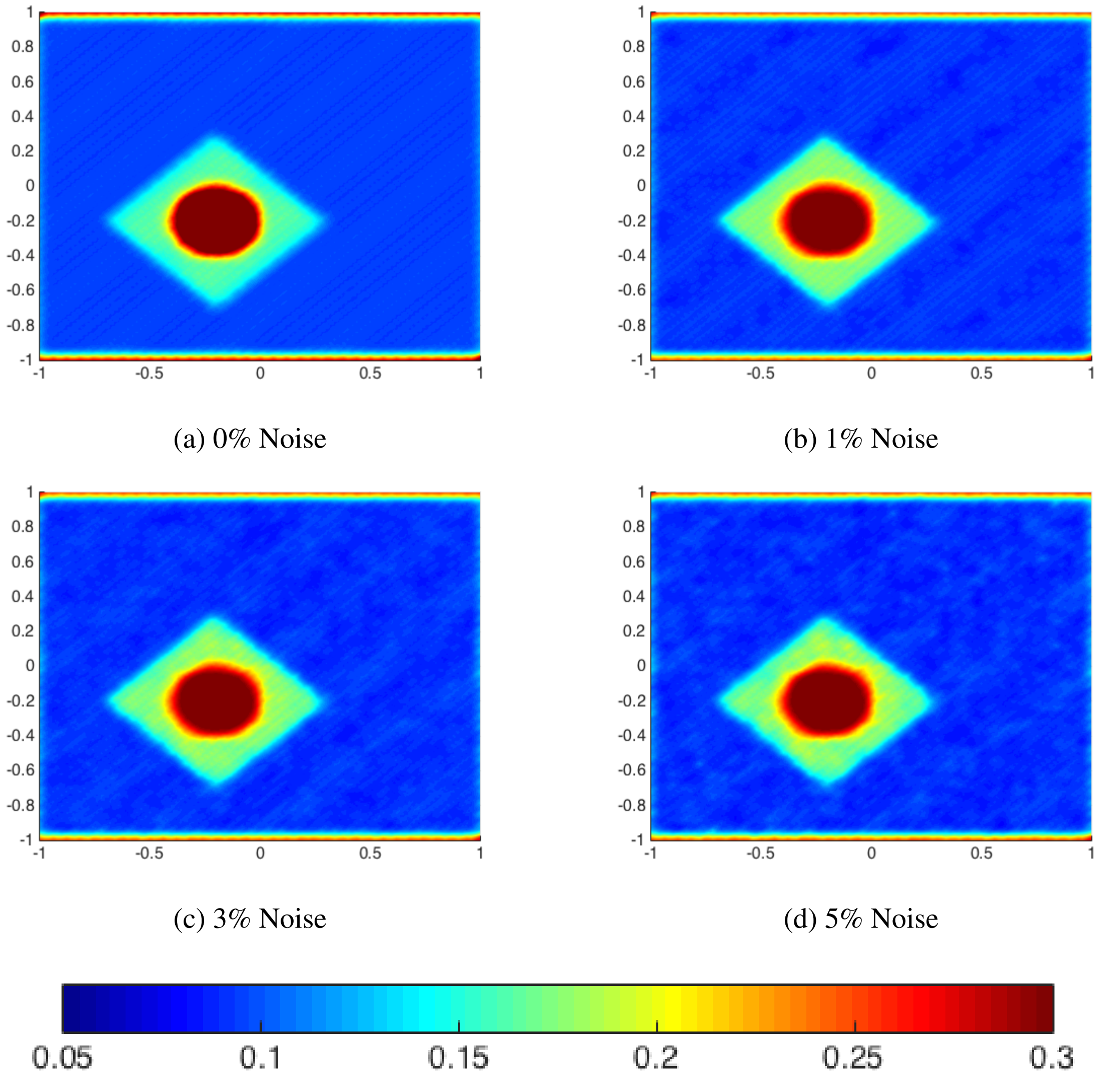
(b) 1% Noise

(c) 3% Noise

(d) 5% Noise

Figure 6: Optimal thermal conductivity field computed using the second order data misfit formulation and the inexact dogleg trust region Newton algorithm.

optimal solution was sensitive to corrupt data. Figures 7 and 8 show the optimal thermal conductivity field computed using the second order CEM formulations based on a nonlinear and linear Hessian formulation, respectively. Both Hessian formulations accurately characterized the location and the shape of the thermal conductivity field of interest. However, the magnitude of the thermal conductivity field around the heterogeneous conductivity field were overestimated. Furthermore, the compliance error minimization formulation once more produced non-optimal thermal conductivity values near the regions were Dirichlet and Neumann boundary conditions were applied. These results further highlight the need for more research to understand/correct the numerical artifacts computed near the regions were Dirichlet or Neumann boundary conditions were applied. However, the significant speedups observed with the second order, and first order, compliance error minimization formulations motivate future research to further improve the proposed inverse problem formulation.



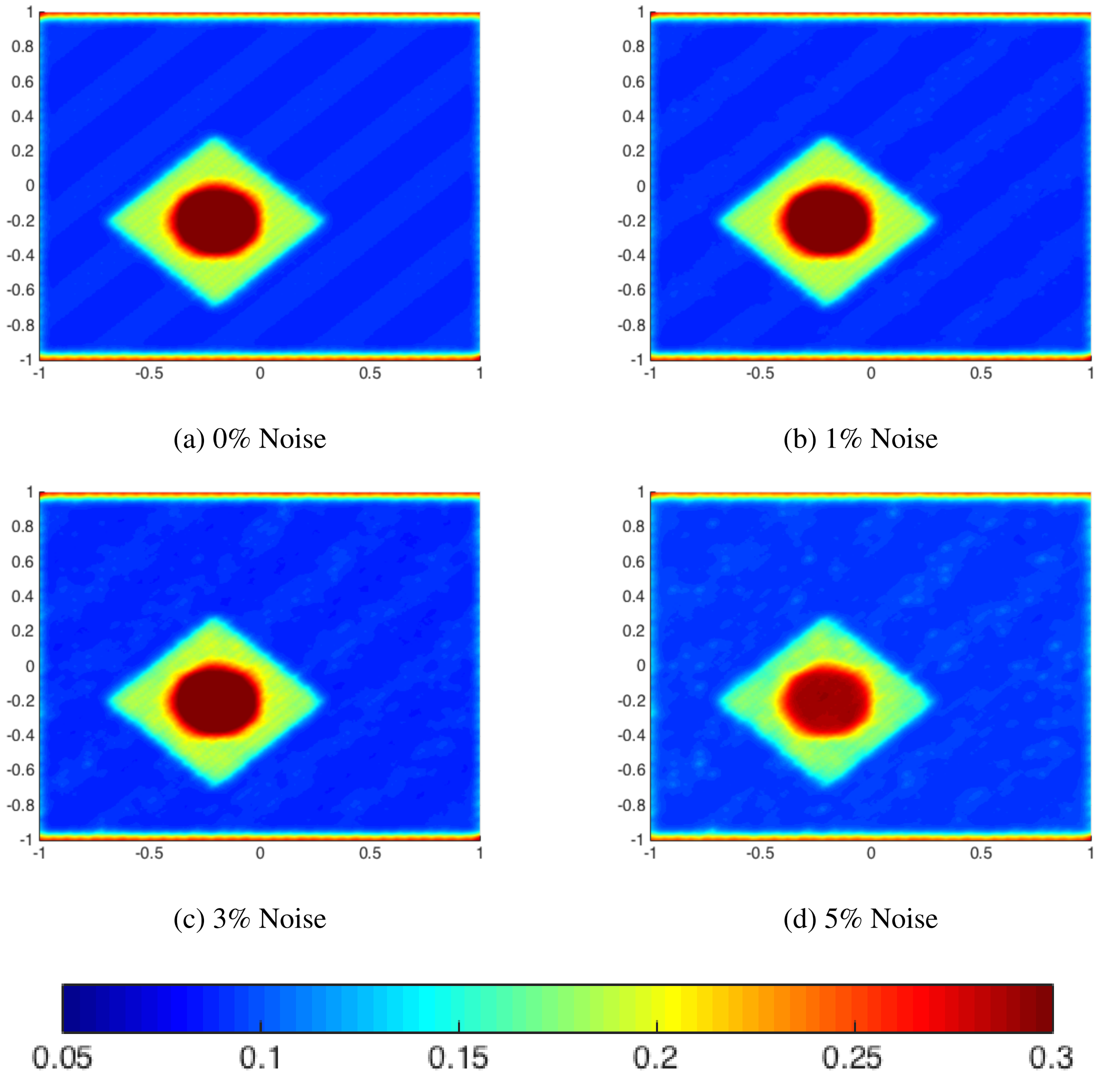
(a) 0% Noise

(b) 1% Noise

(c) 3% Noise

(d) 5% Noise

Figure 7: Optimal thermal conductivity field computed using the second order compliance error minimization formulation based on a nonlinear Hessian formulation and the inexact dogleg trust region Newton algorithm.



(a) 0% Noise

(b) 1% Noise

(c) 3% Noise

(d) 5% Noise

Figure 8: Optimal thermal conductivity field computed using the second order compliance error minimization formulation based on a linear Hessian formulation and the inexact dogleg trust region Newton algorithm.

4 CONCLUSIONS

This paper presented a new formulation for inverse problems based on the compliance error functional. The compliance error functional enabled the computation of the Lagrange multipliers at the expense of just one model evaluation. Thus, the calculation of the Lagrange multipliers did not require the solution of the computationally intensive adjoint problem. This, leads to significant speedups since the computation of the gradient operator only requires one model evaluation per optimization iteration. Furthermore, the implementation of the CEM formulation in production finite element software is greatly simplified since only the forward model evaluation is needed to compute the gradient operator. Therefore, the implementation of the adjoint system of equations is not necessary to solve the inverse problem, greatly simplifying implementation. Likewise, computing the application of the trial step to the nonlinear Hessian operator is greatly simplified since only one additional model evaluation is necessary. This also leads to significant speedups when second order optimization algorithms and the second order CEM formulation are applied to solve the inverse problem.

This paper also presented a linear Hessian formulation based on the compliance error functional to approximate the nonlinear Hessian operator during the Newton iterations. By applying the linear Hessian formulation, optimal results were recovered, while avoiding the additional model evaluation needed to compute the nonlinear Hessian operator based on the compliance error functional. Thus, the linear Hessian formulation further reduced the computational times associated with the inverse problem in heat transfer. However, results suggested that the first order CEM formulation is more computationally effective than the second order CEM formulation. Therefore, the second order CEM formulations presented herein did not provide additional computational benefits over the first order CEM formulation. However, more studies are needed since the second order CEM formulations can be combined with an effective preconditioning strategy to further expedite second order optimization algorithms.

Results also showed that the data misfit formulation produced better results near the regions were Dirichlet and Neumann boundary conditions were applied. The compliance error functional produce perceivable numerical artifacts near the regions were Dirichlet and Neumann boundary conditions were applied. However, the interior thermal conductivity fields computed with the first and second order CEM formulations were accurate. However, the second order CEM formulations were inclined to overestimate the thermal conductivity field near the heterogeneous thermal conductivity region. Future research will focus on exploring alternate inverse problem formulations to improve the CEM strategy. For instance, the accuracy could be improved by combining the data misfit functional, either as an inequality constraint or as an objective term, with the CEM functional. The goal is to preserve the fast convergence properties of the compliance error minimization formulation and improve the accuracy through the data misfit functional. Finally, the results produced with the compliance error minimization formulation were less sensitive to corrupt data in the numerical studies performed herein.

Overall, the results obtained with the compliance error minimization formulation are encouraging since the compliance error minimization formulation produced significant speedups. Furthermore, the first order CEM formulation effectively produce accurate thermal conductivity fields. Regardless of the fact that the second order compliance error formulations were inclined to overestimate the thermal conductivity field, the geometry of the heterogenous thermal conductivity field was highly accurate. Although their is room for improvements, the gains in speed obtained with the compliance error functional are substantial. Hence, this fact motivates further investigation to continue improving the proposed inverse problem formulation.

REFERENCES

[1] B.M. Adams, W.J. Bohnhoff, K.R. Dalbey, J.P. Eddy, M.S. Eldred, D.M. Gay, K. Haskell, P.D. Hough, and L.P. Swiler. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: version 5.0 users manual. *Sandia National Laboratories, Tech. Rep. SAND2010-2183*, 2009.

[2] M. Aguiló, W. Aquino, J.C. Brigham, and M. Fatemi. An inverse problem approach for elasticity imaging through vibroacoustics. *Medical Imaging, IEEE Transactions on*, 29(4):1012–1021, 2010.

[3] M.A. Aguiló, L.P. Swiler, and A. Urbina. First-order formulations for large-scale stochastic parameter estimation within the frameworks of steady state dynamics: the elastic and viscoelastic case. *Inverse Problems*, 28(7):075003, 2012.

[4] O. Allix, P. Feissel, and H.M. Nguyen. Identification strategy in the presence of corrupted measurements. *Engineering computations*, 22(5-6):487–504, 2005.

[5] H.T. Banks, M.L. Joyner, B. Wincheski, and W.P. Winfree. Nondestructive evaluation using reduced-order computational methodology. *Inverse Problems*, 16(4):929–945, 2000.

[6] D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

[7] A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust region methods*, volume 1. SIAM, 2000.

[8] J.E. Dennis and R.B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall, Englewood Cliffs, N.J., 1983.

[9] A. Deraemaeker, P. Ladevèze, and P. Leconte. Reduced bases for model updating in structural dynamics based on the constitutive relation error. *Computer Method in Applied Mechanics and Engineering*, 191(21-22):2427–2444, 2002.

[10] P. Feissel and O. Allix. Modified constitutive relation error identification strategy for transient dynamics with corrupted data: The elastic case. *Computer Method in Applied Mechanics and Engineering*, 196(13-16):1968–1983, 2007.

[11] P.E. Gill, W. Murray, and M.A. Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM journal on optimization*, 12(4):979–1006, 2002.

[12] M. S. Gockenbach and A. A. Khan. *Mathematical Models and Methods for Real World Systems*, chapter A convex objective functional for elliptic inverse problems. Chapman & Hall CRC Taylor & Francis Group, 2006.

[13] M. S. Gockenbach and A. A. Khan. An abstract framework for elliptic inverse problems: Part 1. An output least-squares approach. *Mathematics and Mechanics of Solids*, 12(3):259–276, 2007.

[14] M. S. Gockenbach and A. A. Khan. An abstract framework for elliptic inverse problems: Part 2. An augmented Lagrangian approach. *Mathematics and Mechanics of Solids*, 14(6):517–539, 2009.

[15] M.S. Gockenbach. Numerical analysis of elliptic inverse problems with interior data. *Journal of Physics: Conference Series*, 124(1):1–12, 2008.

[16] M.S. Gockenbach, B. Jadamba, and A.A. Khan. Numerical estimation of discontinuous coefficients by the method of equation error. *Mathematics and Mechanics of Solids*, 1:343–359, 2006.

[17] M. Grdiac, F. Pierron, S. Avril, and E. Toussaint. The virtual fields method for extracting constitutive parameters from full-field measurements: a review. *Strain*, 42(4):233–253, 2006.

[18] M. Heroux, R. Bartlett, V. Howle, R. Hoekstra, J. Hu, T. Kolda, R. Lehoucq, K. Long, R. Pawlowski, E. Phipps, A. Salinger, H. Thornquist, R. Tuminaro, J. Willenbring, and A. Williams. An Overview of Trilinos. SAND2003-2927, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185-1110, 2003.

[19] R. V. Kohn and B. D. Lowe. A variational method for parameter identification. *Mathematical Modeling and Numerical Analysis*, 22(1):119–158, 1998.

[20] R.V. Kohn and A. McKenney. Numerical implementation of a variational method for electrical impedance tomography. *Inverse Problems*, 6(3):389–414, 1990.

[21] R.V. Kohn and M. Vogelius. Determining conductivity by boundary measurements. *Communications on Pure and Applied Mathematics*, 37(3):289–298, 1984.

[22] R.V. Kohn and M. Vogelius. Determining conductivity by boundary measurements II. interior results. *Communications on Pure and Applied Mathematics*, 38(5):643–667, 1985.

[23] P. Ladevèze and Chouaki. A. Updating structural dynamic models with emphasis on the damping properties. *American Institute of Aeronautics and Astronautics Journal*, 36(6):1094–1099, 1998.

[24] P. Ladevèze and D. Leguillon. Error estimates procedures in the finite element method and applications. *SIAM Journal on Numerical Analysis*, 20(3):485–509, 1983.

[25] B.S. Lazarov and O. Sigmund. Filters in topology optimization based on helmholtz-type differential equations. *International Journal for Numerical Methods in Engineering*, 86(6):765–781, 2011.

[26] H. L. Liew and P. M.. Pinsky. Recovery of shear modulus in elastography using an adjoint method with b-spline representation. *Finite Elements in Analysis and Design*, 41(7-8):778–799, 2005.

[27] MATLAB 8.0 and Statistics Toolbox 8.1, The MathWorks, Inc., Natick, Massachusetts, United States.

[28] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006.

[29] A.A. Oberai, N.H. Gokhale, M.M. Doyley, and J.C. Bamber. Evaluation of the adjoint equation based algorithm for elasticity imaging. *Physics in Medicine and Biology*, 49(13):2955, 2004.

[30] A.A. Oberai, N.H. Gokhale, and G.R. Feijóo. Solution of inverse problems in elasticity imaging using the adjoint method. *Inverse Problems*, 19(2):297, 2003.

[31] J.M. Perry. A class of conjugate gradient algorithms with a two-step variable-metric memory. *Discussion Papers*, 269, 1977.

[32] F. Pierron and M. Grédiac. *The Virtual Fields Method: Extracting Constitutive Mechanical Parameters from Full-field Deformation Measurements*. Springer New York, 2012.

[33] R. T. Rockafellar. Lagrange multipliers and optimality. *SIAM Review*, 35:183–238, 1993.

[34] R. Sampath and N. Zabaras. An object-oriented framework for the implementation of adjoint techniques in the design and control of complex continuum systems. *International Journal for Numerical Methods in Engineering*, 48(2):239–266, 2000.

[35] D.F. Shanno. On the convergence of a new conjugate gradient algorithm. *SIAM Journal on Numerical Analysis*, 15(6):1247–1257, 1978.

[36] C.R. Vogel. *Computational methods for inverse problems*. Society for Industrial and Applied Mathematics, 2002.

[37] A. Wächter and L.T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.