Interval-Based Parameter Identification for Structural Static Problems*

Naijia Xiao

School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
naijia.xiao@gatech.edu

Francesco Fedele

School of Civil and Environmental Engineering and School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
fedele@gatech.edu

Rafi L. Muhanna

School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
rafi.muhanna@gatech.edu

Abstract

We present an interval-based approach for parameter identification in structural static problems. Our inverse formulation models uncertainties in measurement data as interval and exploits the Interval Finite Element Method (IFEM) combined with adjoint-based optimization. The inversion consists of a two-step algorithm: first, an estimate of the parameters is obtained by a deterministic iterative solver. Then, the algorithm switches to the interval extension of the previous solver, using the deterministic estimate of the parameters as an initial guess. The formulation is illustrated in solutions of various numerical examples showing how the guaranteed interval enclosures always contain Monte Carlo predictions.

1   Introduction

Parameter identification estimates model parameters of a physical system from available measurements of the system response. It belongs to the class of inverse problems (e.g., [17, 30, 33]). For example, wave tomography is used in geophysics for seismic waveform inversion [12]; in biomedical engineering, optical tomography is used

---

to detect breast cancer tissue via fluorescence [8, 10]; in civil engineering, inversion techniques are used for structural health monitoring or damage detection in safety evaluation [5, 14]. In such problems, the system response is predicted based on initial guessed model parameters, and it is then compared with the actual measurement data. Then, iterative corrections of the model parameters lead to a solution, which minimizes the difference between the predicted system response and measurement data in a least-square or maximum-likelihood sense.

Inevitably, data contain errors caused by measurement devices or unfriendly environmental conditions during data acquisition. Such uncertainties can be modeled using probability theory (e.g., [1, 36, 37]). For example, Kalman filtering (see [4, 20, 35]) provides error estimates on the model parameters based on noisy measurements of the response of a time-evolving system (e.g., [38]). Clearly, probability approaches have their limitations, since they require a prior assumption on the nature of the uncertainty, which is usually modeled as a random Gaussian variable. However, such an assumption is too optimistic or not realistic. In practice, there are often not enough measurements to reliably assess the statistical nature of the associated uncertainties. Instead, we only know bounds on the uncertain variable and some partial information about its probabilities. In this setting, non-probability theories such as fuzzy sets [9, 15], evidence theory [19], and intervals [7, 21] are useful for modeling uncertainties.

In this work, we exploit the Interval Finite Element Method (IFEM) [25, 31] combined with adjoint-based optimization [8, 10] to provide a new algorithm that guarantees interval enclosure of the model parameters from inversion of noisy measurements modeled as intervals. The paper is organized as follows. First, IFEM is reviewed, and new decomposition strategies are presented to limit overestimation due to multiple occurrences of the same variable in the IFEM matrix equations. Then, the deterministic inverse algorithm is formulated using adjoint-based methods, and an extension of the algorithm to intervals is presented. Finally, several numerical examples are discussed to validate the performance of our method.

2   Interval Finite Element Method

Interval Finite Element Method (IFEM) uses intervals to describe uncertain variables and follows the general procedure of conventional Finite Element Method (FEM). Intervals are extension of real numbers. Instead of representing one single point in the real axis, an interval denotes a set of real numbers, which are described by its endpoints,

$$\mathbf{x} = [\underline{x}, \overline{x}] = \{x \,|\, \underline{x} \le x \le \overline{x}, \, x \in \mathbb{R}\}, \tag{1}$$

where $\mathbf{x}$ denotes the interval, $\underline{x}$ and $\overline{x}$ denote its lower and upper bounds, respectively, and bold symbols denote interval quantities. Alternatively, an interval can be represented by its *midpoint* $x_{\mathrm{mid}} = (\underline{x} + \overline{x})/2$ and *radius* $x_{\mathrm{rad}} = (\overline{x} - \underline{x})/2$. The *width* of an interval is defined as $x_{\mathrm{wid}} = (\overline{x} - \underline{x}) = 2x_{\mathrm{rad}}$. Intervals with non-zero midpoint values can be brought into the form of $\mathbf{x} = x_{\mathrm{mid}}(1 + \boldsymbol{\delta}_x)$, where $\boldsymbol{\delta}_x$ has a zero midpoint. The width of $\boldsymbol{\delta}_x$ in percentage is usually referred to as the uncertainty level of $\mathbf{x}$. For a detailed discussion of interval arithmetic and extensions to interval matrices and functions, we refer to [2, 22, 23].

Overestimation due to dependency is the curse in any application of interval arithmetic (see [24, 25]). To reduce it, we propose a new decomposition strategy for the

stiffness matrix $\mathbf{K}$ and the nodal equivalent load $\mathbf{f}$ of a structural system governed by the equilibrium condition $\mathbf{Ku} = \mathbf{f}$. Here, $\mathbf{K}$ and $\mathbf{f}$ are decomposed as

$$\mathbf{K} = A \operatorname{diag}(\Lambda \boldsymbol{\alpha}) A^T, \quad \mathbf{f} = M \boldsymbol{\delta}, \tag{2}$$

where A, Λ, and M are scalar matrices; $\boldsymbol{\alpha}$ and $\boldsymbol{\delta}$ are interval vectors containing all the uncertainties in the system; and $\operatorname{diag}(\mathbf{v})$ maps a vector $\mathbf{v}$ into a diagonal matrix, whose diagonal is $\mathbf{v}$. In this way, we separate deterministic and uncertain terms, and multiple occurrences of the same variable are avoided. In practice, the decomposition in Eq. (2) is done in two steps. In the first step, the element stiffness matrix $\mathbf{K}_e$ and the element nodal equivalent load $\mathbf{f}_e$ are decomposed into A_e, Λ_e, M_e, $\boldsymbol{\alpha}_e$, and $\boldsymbol{\delta}_e$ using Eq. (2) in the local reference system. In the second step, A_e, Λ_e, and M_e are assembled into A, Λ, and M in the global reference system.

In particular, for an element with uncertain material properties

$$\mathbf{K}_e = \int_{\Omega} B_e^T \mathbf{E}_e B_e \, d\Omega, \tag{3}$$

where the integration domain Ω is the entire element, B_e is the scalar strain-displacement matrix at arbitrary locations inside the element, and $\mathbf{E}_e$ is the interval constitutive matrix, which is a function of material uncertainties. To reduce overestimation due to dependency, $\mathbf{K}_e$ is decomposed as

$$\mathbf{K}_e = A_e \operatorname{diag}(\Lambda_e \boldsymbol{\alpha}_e) A_e^T, \tag{4}$$

where A_e and Λ_e are scalar matrices, and the interval vector $\boldsymbol{\alpha}_e$ contains all the uncertainties of the element.

From Eq. (3), numerical integration yields

$$\mathbf{K}_e = \sum_{j=1}^{m} w_j J(\xi_j) B_e^T(\xi_j) \mathbf{E}_e(\xi_j) B_e(\xi_j), \tag{5}$$

where m is the number of integration points used, ξ_j and w_j are respectively the coordinates and weights of the integration points, and J is the determinant of the Jacobian between local and global reference systems. The scalar matrices A_e and Λ_e and the interval vector $\boldsymbol{\alpha}_e$ in Eq. (4) are given by

$$A_e = \left\{ B_e^T(\xi_1)\Phi_e \quad \cdots \quad B_e^T(\xi_m)\Phi_e \right\},$$

$$\Lambda_e = \left\{ \begin{matrix} w_1 J(\xi_1)\varphi_e & & \\ & \ddots & \\ & & w_m J(\xi_m)\varphi_e \end{matrix} \right\}, \qquad \boldsymbol{\alpha}_e = \left\{ \begin{matrix} \mathbf{E}(\xi_1) \\ \vdots \\ \mathbf{E}(\xi_m) \end{matrix} \right\}, \tag{6}$$

where $\mathbf{E}(\xi_j)$ denote interval Young's modulus at the j-th integration point. Further, Φ_e and φ_e are obtained from the interval constitutive matrix, which is decomposed as

$$\mathbf{E}_e(\xi_j) = \Phi_e \operatorname{diag}\{\varphi_e \mathbf{E}(\xi_j)\} \Phi_e^T. \tag{7}$$

The decomposition of the element nodal equivalent load $\mathbf{f}_e$ is done exploiting the M-$\boldsymbol{\delta}$ method [26], viz. $\mathbf{f}_e = M_e \boldsymbol{\delta}_e$. Here,

$$\mathbf{f}_e = \sum_{j=1}^{n} N^T(\xi_j) \mathbf{f}_c(\xi_j) + \int_{\Omega} N^T(\xi) \mathbf{f}_d(\xi) \, d\Omega, \tag{8}$$

where n is the number of concentrated loads acting on the element, $N(\xi)$ is the displacement interpolation matrix for the element, $\mathbf{f}_c(\xi_j)$ are the concentrated loads under consideration, Ω is the integration domain in which the distributed load $\mathbf{f}_d(\xi)$ is nonzero.

A further simplification can be obtained by rewriting $\mathbf{f}_c(\xi) = L_c(\xi)\boldsymbol{\delta}_e$ and $\mathbf{f}_d(\xi) = L_d(\xi)\boldsymbol{\delta}_e$ as function of the load uncertainty vector $\boldsymbol{\delta}_e$, where $L_c(\xi)$ and $L_d(\xi)$ are scalar matrices. Then from Eq. (8)

$$\mathbf{f}_e = \sum_{j=1}^{n} N^T(\xi_j) L_c(\xi_j)\boldsymbol{\delta}_e + \int_{\Omega} N^T(\xi) L_d(\xi)\boldsymbol{\delta}_e \, d\Omega \tag{9}$$

$$= \left\{ \sum_{j=1}^{n} N^T(\xi_j) L_c(\xi_j) + \int_{\Omega} N^T(\xi) L_d(\xi) \, d\Omega \right\} \boldsymbol{\delta}_e = M_e \boldsymbol{\delta}_e \,.$$

Here, M_e is the matrix within braces, which depends on the displacement interpolation matrix $N(\xi)$ and load distribution functions $L_c(\xi)$ and $L_d(\xi)$.

The global $\mathbf{K}$ and $\mathbf{f}$ follow from the conventional assembly strategy [6], i.e.,

$$\mathbf{K} = \sum_e T_e^T \mathbf{K}_e T_e \,, \qquad \mathbf{f} = \sum_e T_e^T \mathbf{f}_e \,, \tag{10}$$

where T_e is the transformation matrix between the global and local nodal displacement vector $\mathbf{u}$ and $\mathbf{u}_e$. Note that $\mathbf{K}_e$, $\mathbf{f}_e$, and T_e are not necessarily the same for each element. By inserting $\mathbf{K}_e = A_e \, \mathrm{diag}(\Lambda_e \boldsymbol{\alpha}_e) A_e^T$ of Eq. (4) into Eq. (10), the decomposition rule for $\mathbf{K}$ is

$$\mathbf{K} = \sum_e T_e^T A_e \, \mathrm{diag}(\Lambda_e \boldsymbol{\alpha}_e) A_e^T T_e \tag{11}$$

$$= \left\{ T_e^T A_e \quad \cdots \quad T_e^T A_e \right\} \mathrm{diag} \left\{ \begin{matrix} \Lambda_e \boldsymbol{\alpha}_e \\ \vdots \\ \Lambda_e \boldsymbol{\alpha}_e \end{matrix} \right\} \left\{ \begin{matrix} A_e^T T_e \\ \vdots \\ A_e^T T_e \end{matrix} \right\} \,.$$

Here, the vector $\boldsymbol{\alpha}_e$ lists the uncertain interpolated Young's moduli at the element integration points, and it is related to the system parameter vector $\boldsymbol{\alpha}$ via $\boldsymbol{\alpha}_e = L_\alpha \boldsymbol{\alpha}$. Comparing terms in Eqs. (2) and (11) yields the assembly rules for A and Λ

$$A = \left\{ T_e^T A_e \quad \cdots \quad T_e^T A_e \right\}, \qquad \Lambda = \left\{ \begin{matrix} \Lambda_e L_\alpha \\ \vdots \\ \Lambda_e L_\alpha \end{matrix} \right\} \,. \tag{12}$$

Again, note that A_e, Λ_e, and L_α are not necessarily the same for each element. Similarly, the decomposition rule for $\mathbf{f}$ and the assembly rule for M follow by introducing $\mathbf{f}_e = M_e \boldsymbol{\delta}_e$ into Eq. (10) and setting $\boldsymbol{\delta}_e = L_\delta \boldsymbol{\delta}$, that is,

$$\mathbf{f} = \sum_e T_e^T \mathbf{f}_e = \sum_e T_e^T M_e \boldsymbol{\delta}_e = \left\{ T_e^T M_e \quad \cdots \quad T_e^T M_e \right\} \left\{ \begin{matrix} \boldsymbol{\delta}_e \\ \vdots \\ \boldsymbol{\delta}_e \end{matrix} \right\}, \tag{13}$$

$$\Rightarrow \quad M_e = \left\{ T_e^T M_e \quad \cdots \quad T_e^T M_e \right\} \left\{ \begin{matrix} L_\delta \\ \vdots \\ L_\delta \end{matrix} \right\} = \sum_e T_e^T M_e L_\delta \,.$$

The resulting stiffness matrix $\mathbf{K}$ in Eq. (10) is still singular, as essential boundary conditions have not been applied yet. To eliminate the singularity, $\mathbf{u}$ must satisfy the additional constraint $C\mathbf{u} = 0$, with C denoting a constraint matrix [31]. Each row of C states one constraint, and the corresponding entry is set equal to 1, leaving the rest of the row null. Then the equilibrium equation follows from setting to zero the first variation of the energy functional $\mathbf{\Pi}$ of the structure

$$\mathbf{\Pi} = \frac{1}{2}\mathbf{u}^T\mathbf{K}\mathbf{u} - \mathbf{u}^T\mathbf{f} + \boldsymbol{\lambda}^T C\mathbf{u}\,. \tag{14}$$

That is,

$$\begin{Bmatrix} \mathbf{K} & C^T \\ C & 0 \end{Bmatrix} \begin{Bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix}. \tag{15}$$

where the Lagrangian multiplier $\boldsymbol{\lambda}$ enforces $C\mathbf{u} = 0$. If $\mathbf{K}$ is composed of degenerated intervals (intervals with zero width), we can establish a direct relationship between $\mathbf{u}$ and $\mathbf{f}$ by inverting the generalized stiffness matrix in Eq. (15),

$$\begin{Bmatrix} \mathbf{K} & C^T \\ C & 0 \end{Bmatrix}^{-1} = \begin{Bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12}^T \\ \mathbf{G}_{11} & \mathbf{G}_{22} \end{Bmatrix}, \qquad \Rightarrow \qquad \mathbf{u} = \mathbf{G}_{11}\mathbf{f}\,. \tag{16}$$

In other words, we find the flexibility matrix (inverse of the stiffness matrix $\mathbf{K}$) under the constraint $C\mathbf{u} = 0$.

3   Deterministic Inverse Solver

Given an interval load uncertainty vector $\boldsymbol{\delta}$ and an interval measurement vector $\tilde{\mathbf{u}}$, a deterministic solution of the model parameters $\boldsymbol{\alpha}$ is sought using midpoint values of $\boldsymbol{\delta}$ and $\tilde{\mathbf{u}}$, and all interval quantities are replaced with their midpoint values. Drawing from Fedele, et al. [11], the algorithm is derived using adjoint based optimization, and it exploits conjugate gradient type methods to find optimal estimates of the unknown parameters, as shown in the following discussions.

Assume measurements $\tilde{u}$ are collected at sampling points on the structure. Our inverse solver aims at minimizing the difference between the predicted response Hu given in terms of the nodal displacement vector u and the actual measurements vector $\tilde{u}$, under the equilibrium constraint $Ku = f$. To do so, define the objective functional

$$\Gamma = \frac{1}{2}(Hu - \tilde{u})^T S(Hu - \tilde{u}) + w^T(Ku - f) + \frac{1}{2}\gamma(\alpha^T R\alpha)\,, \tag{17}$$

where S is a diagonal matrix defining the weight for each measurement, w is the Lagrangian multiplier to enforce equilibrium [11], and the last term provides regularization for the problem if necessary. Here, γ is the regularizer weight, and R is the finite-difference matrix associated with second-order differentiation (e.g., [17, 33]).

From the decomposition in Eqs. (2) and (17), the first variation of Γ

$$\delta\Gamma = \delta u^T H^T S(Hu - \tilde{u}) + \delta w^T(Ku - f) + \delta u^T K^T w \tag{18}$$
$$+ w^T A\,\mathrm{diag}(\Lambda\delta\alpha)A^T u + \delta\alpha^T(\gamma R)\alpha$$

is null if

$$
\begin{cases}
Ku - f = 0\,; \\
Kw + (H^T S H)u - (H^T S)\tilde{u} = 0\,; \\
\Lambda^T (A^T u \circ A^T w) + (\gamma R)\alpha = 0\,,
\end{cases}
\tag{19}
$$

where $a \circ b$ denotes the element-by-element (Hadamard) product of two vectors a and b. To obtain Eq. (19) from Eq. (18), we have exploited the matrix symmetry [see Eq. (2)]

$$
K = K^T = A \operatorname{diag}(\Lambda \alpha) A^T
\tag{20}
$$

and the chain of identities

$$
w^T A \operatorname{diag}(\Lambda \delta \alpha) A^T u = w^T A(\Lambda \delta \alpha \circ A^T u) = \delta \alpha^T \Lambda^T (A^T u \circ A^T w)\,.
\tag{21}
$$

The three equations in Eq. (19) can be interpreted as: *i*) equilibrium condition of the original system with equivalent load f, *ii*) equilibrium condition for the adjoint system with equivalent load $H^T S(\tilde{u} - Hu)$, and *iii*) optimal condition that the gradient g of Γ with respect to α is zero at the solution point.

The first two equations in Eq. (19), viz. the equilibrium conditions for the original and adjoint systems, can be recast in block form

$$
\begin{Bmatrix} H^T S H & K \\ K & 0 \end{Bmatrix} \begin{Bmatrix} u \\ w \end{Bmatrix} = \begin{Bmatrix} 0 & H^T S \\ M & 0 \end{Bmatrix} \begin{Bmatrix} \delta \\ \tilde{u} \end{Bmatrix}\,,
\tag{22}
$$

using the decomposition $f = M\delta$. The unknown vectors u and w follow as

$$
\begin{Bmatrix} u \\ w \end{Bmatrix} = \begin{Bmatrix} 0 & K^{-1} \\ K^{-1} & -K^{-1}H^T S H K^{-1} \end{Bmatrix} \begin{Bmatrix} 0 & H^T S \\ M & 0 \end{Bmatrix} \begin{Bmatrix} \delta \\ \tilde{u} \end{Bmatrix}\,,
\tag{23}
$$

The corresponding objective functional Γ and its gradient g with respect to α, viz. third equation in Eq. (19), can be expressed in terms of u, w, and α as

$$
\Gamma = \frac{1}{2}(Hu - \tilde{u})^T S(Hu - \tilde{u}) + \frac{1}{2}\alpha^T (\gamma R)\alpha\,;
\tag{24}
$$

$$
g = \frac{\partial \Gamma}{\partial \alpha} = \Lambda^T (A^T u \circ A^T w) + (\gamma R)\alpha\,.
$$

The conjugate gradient method [3, 39, 40] is exploited to solve iteratively for Eq. (19). We start from a random initial guess α_1 and a descending direction d_1 along which Γ decreases. A natural choice for d_1 is the opposite gradient direction, $d_1 = -g_1$. At the i-th step, the model parameter α is updated as

$$
\alpha_{i+1} = \alpha_i + s_i d_i\,,
\tag{25}
$$

where s_i is the step size. We use the inexact line search method to find an acceptable s_i along the descending direction d_i. This should be large enough to yield a significant decrease in Γ, while not too large to deviate too far from the optimal point. We adopt the weak Wolfe criterion [16, 34]

$$
\tau_l \leq \frac{\Gamma_{i+1} - \Gamma_i}{s_i g_i^T d_i}\,, \qquad \frac{g_{i+1}^T d_i}{g_i^T d_i} \leq \tau_u\,,
\tag{26}
$$

where $0 < \tau_l < \tau_u < 1$. In the next iteration step, the descending direction d_{i+1} is determined by

$$d_{i+1} = -g_{i+1} + \theta_i d_i \,, \tag{27}$$

where the parameter θ_i can be chosen in various ways. Popular choices for θ_i include

$$\theta_i = \frac{g_{i+1}^T g_{i+1}}{g_i^T g_i}, \qquad \text{Fletcher-Reeves [13]};$$

$$\theta_i = \frac{g_{i+1}^T (g_{i+1} - g_i)}{g_i^T g_i}, \qquad \text{Polak-Ribière-Polyak [28, 29]}; \tag{28}$$

$$\theta_i = \frac{g_{i+1}^T (g_{i+1} - g_i)}{d_i^T (g_{i+1} - g_i)}, \qquad \text{Hestenses-Stiefel [18]}.$$

The algorithm stops when the gradient g and the update on α are both small enough,

$$\frac{\|g_{i+1}\|}{\|g_1\|} \le \tau \,, \quad \frac{\|\alpha_{i+1} - \alpha_i\|}{\|\alpha_i\|} \le \tau \,, \tag{29}$$

where τ is the error tolerance. In our implementation, all three choices in Eq. (28) perform comparably. For simplicity, we adopt the Polak-Ribière-Polyak rule in later numerical simulations.

4 Interval Inverse Solver

The interval algorithm consists of two steps. In the first step, deterministic solutions u_0, w_0, and α_0 are obtained using the deterministic inverse solver described in the previous section. In the second step, these solutions are used as initial guesses for an interval-based inverse solver, a generalization to interval of the deterministic solver. This is formulated drawing from Fedele et al. [11]. In particular, given an interval load uncertainty vector $\boldsymbol{\delta}$ and interval measurements $\tilde{\mathbf{u}}$, the unknown interval $\mathbf{u}$, $\mathbf{w}$, and $\boldsymbol{\alpha}$ satisfy the interval extension of Eq. (19), that is

$$\begin{cases} \mathbf{K}(\boldsymbol{\alpha})\mathbf{u} - M\boldsymbol{\delta} = 0 \,; \\ \mathbf{K}(\boldsymbol{\alpha})\mathbf{w} + (H^T S H)\mathbf{u} - (H^T S)\tilde{\mathbf{u}} = 0 \,; \\ \Lambda^T (A^T \mathbf{u} \circ A^T \mathbf{w}) + (\gamma R)\boldsymbol{\alpha} = 0 \,, \end{cases} \tag{30}$$

where $\mathbf{K}(\boldsymbol{\alpha})$ emphasizes the dependence on the unknown parameter $\boldsymbol{\alpha}$. To solve for Eq. (30), define δ_0 and $\tilde{u}_0$ as the midpoint values of $\boldsymbol{\delta}$ and $\tilde{\mathbf{u}}$, respectively. Then δ_0, $\tilde{u}_0$, u_0, w_0, and α_0 satisfy the optimality conditions in Eq. (19). Now, introduce the auxiliary variables

$$\begin{array}{ccc} \Delta\boldsymbol{\delta} = \boldsymbol{\delta} - \delta_0 \,, & \Delta\tilde{\mathbf{u}} = \tilde{\mathbf{u}} - \tilde{u}_0 \,; \\ \Delta\mathbf{u} = \mathbf{u} - u_0 \,, & \Delta\mathbf{w} = \mathbf{w} - w_0 \,, & \Delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \alpha_0 \end{array} \tag{31}$$

to represent deviations of the interval solutions from the corresponding reference vectors. Then, the following equalities hold

$$\begin{aligned} \mathbf{K}\mathbf{u} &= K_0 u_0 + K_0 \Delta\mathbf{u} + \Delta\mathbf{K} u_0 + \Delta\mathbf{K}\Delta\mathbf{u} \,; \\ \mathbf{K}\mathbf{w} &= K_0 w_0 + K_0 \Delta\mathbf{w} + \Delta\mathbf{K} w_0 + \Delta\mathbf{K}\Delta\mathbf{w} \,; \\ A^T \mathbf{u} \circ A^T \mathbf{w} &= A^T u_0 \circ A^T w_0 + A^T u_0 \circ A^T \Delta\mathbf{w} \\ &\quad + A^T \Delta\mathbf{u} \circ A^T w_0 + A^T \Delta\mathbf{u} \circ A^T \Delta\mathbf{w} \,. \end{aligned} \tag{32}$$

These together with

$$\Delta \mathbf{K} \mathbf{u} = A \operatorname{diag}(\Lambda \Delta \boldsymbol{\alpha}) A^T \mathbf{u} = A \operatorname{diag}(A^T \mathbf{u}) \Lambda \Delta \boldsymbol{\alpha} \tag{33}$$

are used repeatedly to rewrite Eq. (30) as

$$\begin{Bmatrix} H^T S H & K_0 & C_{w0}^T \\ K_0 & 0 & C_{u0}^T \\ C_{w0} & C_{u0} & \gamma R \end{Bmatrix} \begin{Bmatrix} \Delta \mathbf{u} \\ \Delta \mathbf{w} \\ \Delta \boldsymbol{\alpha} \end{Bmatrix} = \begin{Bmatrix} 0 & H^T S \\ M & 0 \\ 0 & 0 \end{Bmatrix} \begin{Bmatrix} \Delta \boldsymbol{\delta} \\ \Delta \tilde{\mathbf{u}} \end{Bmatrix}$$
$$- \begin{Bmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & \Lambda^T \end{Bmatrix} \begin{Bmatrix} A^T \Delta \mathbf{w} \circ \Lambda \Delta \boldsymbol{\alpha} \\ A^T \Delta \mathbf{u} \circ \Lambda \Delta \boldsymbol{\alpha} \\ A^T \Delta \mathbf{u} \circ A^T \Delta \mathbf{w} \end{Bmatrix}, \tag{34}$$

where subscripts 0 denote matrices related to u_0, w_0, and α_0. In particular,

$$C_{u0} = \Lambda^T \operatorname{diag}(A^T u_0) A^T, \qquad C_{w0} = \Lambda^T \operatorname{diag}(A^T w_0) A^T. \tag{35}$$

Eq. (34) can be written in the compact form

$$K_h \Delta \mathbf{u}_h = M_h \Delta \boldsymbol{\delta}_h - A_h \boldsymbol{\Theta}(A_h^T \Delta \mathbf{u}_h), \tag{36}$$

which emphasizes the direct relationship between uncertainties of the given data $\Delta \boldsymbol{\delta}$ and $\Delta \tilde{\mathbf{u}}$ and those of the unknown vectors $\Delta \mathbf{u}$, $\Delta \mathbf{w}$, and $\Delta \boldsymbol{\alpha}$. Here, K_h, M_h, and A_h are known scalar matrices, and $\Delta \mathbf{u}_h$ contains the unknown interval vectors $\Delta \mathbf{u}$, $\Delta \mathbf{w}$, and $\Delta \boldsymbol{\alpha}$. Further, $\Delta \boldsymbol{\delta}_h$ contains the known interval vectors $\Delta \boldsymbol{\delta}$ and $\Delta \tilde{\mathbf{u}}$. $A_h^T \Delta \mathbf{u}_h$ is composed of the secondary unknown vectors $A^T \Delta \mathbf{u}$, $A^T \Delta \mathbf{w}$, and $\Lambda \Delta \boldsymbol{\alpha}$. The functional $\boldsymbol{\Theta}(\ )$ in Eq. (36) maps $A_h^T \Delta \mathbf{u}_h$ into the following interval vector

$$\boldsymbol{\Theta}(A_h^T \Delta \mathbf{u}_h) = \boldsymbol{\Theta} \left(\begin{Bmatrix} A^T \Delta \mathbf{u} \\ A^T \Delta \mathbf{w} \\ \Lambda \Delta \boldsymbol{\alpha} \end{Bmatrix} \right) = \begin{Bmatrix} A^T \Delta \mathbf{w} \circ \Lambda \Delta \boldsymbol{\alpha} \\ A^T \Delta \mathbf{u} \circ \Lambda \Delta \boldsymbol{\alpha} \\ A^T \Delta \mathbf{u} \circ A^T \Delta \mathbf{w} \end{Bmatrix}. \tag{37}$$

If the square matrix K_h is invertible, Eq. (36) can be recast into the fixed-point form

$$\Delta \mathbf{u}_h = (K_h^{-1} M_h) \Delta \boldsymbol{\delta}_h - (K_h^{-1} A_h) \boldsymbol{\Theta}(A_h^T \Delta \mathbf{u}_h), \tag{38}$$

which is solvable by a new variant of the method of Neumaier and Pownuk [27]. In particular, we introduce auxiliary variable $\mathbf{v}_h = A_h^T \Delta \mathbf{u}_h$, and the corresponding fixed-point equation follows from Eq. (38) as

$$\mathbf{v}_h = A_h^T \Delta \mathbf{u}_h = (A_h^T K_h^{-1} M_h) \Delta \boldsymbol{\delta}_h - (A_h^T K_h^{-1} A_h) \boldsymbol{\Theta}(\mathbf{v}_h). \tag{39}$$

From this, the following iterative scheme is proposed to find a guaranteed enclosure for $\mathbf{v}_h$. The iteration starts from the trivial initial guess $\mathbf{v}_h^1 = (A_h^T K_h^{-1} M_h) \Delta \boldsymbol{\delta}_h$ and proceeds in accord with

$$\mathbf{v}_h^{j+1} = (A_h^T K_h^{-1} M_h) \Delta \boldsymbol{\delta}_h - (A_h^T K_h^{-1} A_h) \boldsymbol{\Theta}(\mathbf{v}_h^j), \tag{40}$$

where superscripts of $\mathbf{v}_h$ denote iteration steps. The iteration stops when there is no change in $\mathbf{v}_h$ in two consecutive steps, and the converged result is denoted by $\mathbf{v}_h^*$. This is an outer solution for the exact fixed-point $\mathbf{v}_h$ in Eq. (39), due to the isotonic inclusion of interval operations [23]. An outer solution for $\Delta \mathbf{u}_h$ is obtained by substituting $A_h^T \mathbf{u}_h$ in Eq. (38) with $\mathbf{v}_h^*$. Then the final interval enclosures $\mathbf{u}$, $\mathbf{w}$, and $\boldsymbol{\alpha}$ are obtained by adding $\Delta \mathbf{u}$, $\Delta \mathbf{w}$, and $\Delta \boldsymbol{\alpha}$ (i.e., $\Delta \mathbf{u}_h$) to u_0, w_0, and α_0, respectively.

To further reduce overestimation, the deterministic matrices $(K_h^{-1} M_h)$ and $(K_h^{-1} A_h)$ in Eq. (38) and $(A_h^T K_h^{-1} M_h)$ and $(A_h^T K_h^{-1} A_h)$ in Eq. (39) are calculated before multiplication with the interval vectors $\Delta\boldsymbol{\delta}_h$ and $\boldsymbol{\Theta}(\mathbf{v}_h)$.

The similarities between our method and the method of Neumaier and Pownuk are observed by comparing their Eq. (4.11) in [27]

$$\mathbf{v} = \{(ACa) + (ACF)\mathbf{b} + (ACB)\mathbf{d}\} \cap \mathbf{v}, \qquad \mathbf{d} = \{(D_0 - \mathbf{D})\mathbf{v}\} \cap \mathbf{d}$$

and their Eq. (4.12)

$$\mathbf{u} = \{(Ca) + (CF)\mathbf{b} + (CB)\mathbf{d}\}$$

with our Eqs. (40) and (38) in this paper, respectively. Our method skips introducing the auxiliary variable $\mathbf{d}$ and uses a very simple initial guess $\mathbf{v}_h^1 = (A_h^T K_h^{-1} M_h)\Delta\boldsymbol{\delta}_h$.

5   Interval-Based Parameter Identification

In summary, the flowchart of our two-step interval-based inverse algorithm is given in Figure 1. Assume that a finite element model for the structure under study is given. First, we use the deterministic inverse solver introduced in Section 3 to estimate a degenerate interval solution for the unknown parameters. In the second step, the degenerate estimate is used as an initial guess for the interval-based inverse solver defined in Section 4. The numerical experiments discussed later provide strong evidence that our two-step algorithm gives interval enclosures of the exact parameters.

Note that the scalar matrices A, Λ, and M are assembled from their element counterparts A_e, Λ_e, and M_e, and the constraint matrix C accounts for essential boundary conditions. The generation of interval load uncertainty vector $\boldsymbol{\delta}$ and the interval measurement vector $\tilde{\mathbf{u}}$ are then determined following the steps below. In particular, to simulate interval measurements with perturbed midpoint (with respect to the exact values) and perturbed radius, $\tilde{\mathbf{u}}$ is computed as follows:

1. Use a structural FEM model (not necessarily that used in the inversion) to generate the exact measurement data $\tilde{u}_{exact}$.

2. The interval vector $\tilde{\mathbf{u}}_{exact}$ is set with midpoint value $\tilde{u}_{exact}$ and radius equal to the device tolerance δ_u.

3. An ensemble of perturbed measurements $\tilde{u}_i$ are generated by adding random noise to $\tilde{u}_{exact}$. The random noise is chosen smaller than the tolerance δ_u so that $\tilde{u}_i \in \tilde{\mathbf{u}}_{exact}$.

4. Perturbed interval measurement vectors $\tilde{\mathbf{u}}_i$ are generated using $\tilde{u}_i$ as midpoint and device tolerance δ_u as radius. $\tilde{\mathbf{u}}_i$ contains $\tilde{u}_{exact}$, i.e., $\tilde{u}_{exact} \in \tilde{\mathbf{u}}_i$.

5. The measurement vector $\tilde{\mathbf{u}}$ is obtained as the intersection of all the $\tilde{\mathbf{u}}_i$ in the ensemble. As a result, $\tilde{\mathbf{u}}$ contains a random perturbation, and it still contains $\tilde{u}_{exact}$, i.e., $\tilde{u}_{exact} \in \tilde{\mathbf{u}}$.

In the deterministic solver, to illustrate the robustness of our algorithm, the initial guess is set as $E = 60$ GPa for a structure made of copper, and $E = 160$ GPa for steel. Then the gradient g in Eq. (24) at the current iteration is computed from the solution vectors u and w of the original and adjoint systems [see Eq. (23)]. Further, we use the weak Wolfe criterion for the inexact line search, setting $\tau_l = 1/4$ and $\tau_u = 1/2$ in Eq. (26). The Polak-Ribière-Polyak rule in Eq. (28) is used for the update of the

Finite element modeling

Read the input file

Compute A_e, Λ_e, M_e and $\boldsymbol{\delta}_e$ for each element, as in Eq. (6) and Eq. (9)

Assemble A_e, Λ_e and M_e into A, Λ and M, as in Eq. (12) and Eq. (13)

Form constraint matrix C and γ, R for the regularizer

Determine load uncertainty $\boldsymbol{\delta}$ and measurements $\bar{\mathbf{u}}$

Deterministic adjoint inverse solver

Make an initial guess for α

Compute K, as in Eq. (2), and its inverse K^{-1}, as in Eq. (16)

Solve the original and adjoint systems for u and w, as in Eq. (23)

Compute gradient g, as in Eq. (24)

Update α as in Eq. (25)

Eq. (26) satisfied? No / Yes

Update d, as in Eq. (27)

Eq. (29) satisfied? No / Yes

Interval adjoint inverse solver

Save u, w, α as u_0, w_0, α_0

Compute C_{u0}, C_{w0}, K_0 and form matrices K_h, M_h, A_h, as in Eqs. (34) and (36)

Compute $K_h^{-1}M_h$, $K_h^{-1}A_h$ and $A_h^T K_h^{-1}M_h$, $A_h^T K_h^{-1}A_h$

Use initial guess $\mathbf{v}_h^1 = \left(A_h^T K_h^{-1}M_h\right)\Delta\boldsymbol{\delta}_h$

Compute $\mathbf{v}_h^{j+1}$ according to Eq. (40)

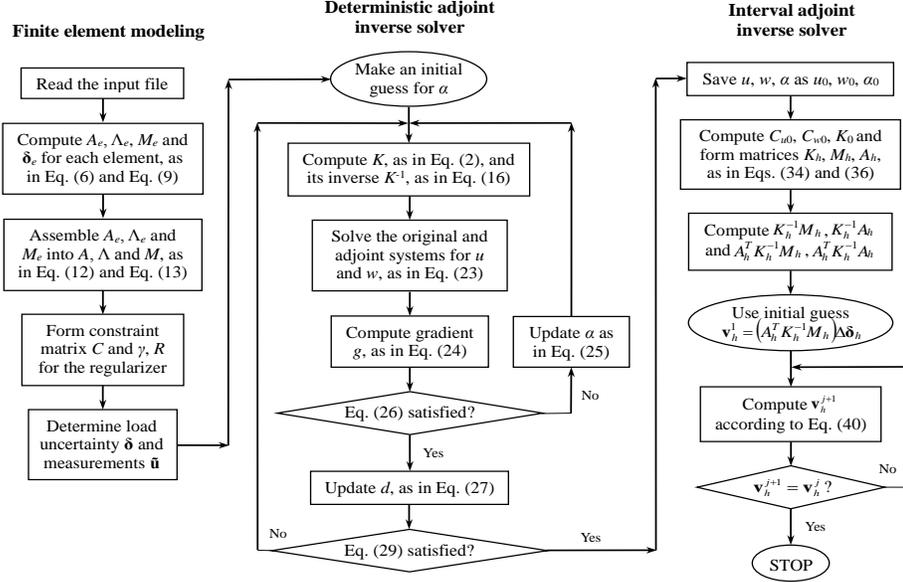$\mathbf{v}_h^{j+1} = \mathbf{v}_h^j$? No / Yes

STOP

Figure 1: Flowchart for interval-based parameter identification.

descending directions. In the stopping criterion (29), the error tolerance $\tau = 1 \times 10^{-10}$ under all circumstances.

In the interval solver, before starting the iteration, we first compute the matrices C_{u0}, C_{w0}, and K_0 in Eq. (34). Then we compute the block matrices K_h, M_h, and A_h in Eq. (36), and $(K_h^{-1}M_h)$, $(K_h^{-1}A_h)$, $(A_h^T K_h^{-1}M_h)$, and $(A_h^T K_h^{-1}A_h)$ are computed in advance to solve for $\Delta\mathbf{u}_h$ and $\mathbf{v}_h$ in Eqs. (38) and (39), respectively. As K_h, M_h, and A_h contain a significant number of null-entries, it is more efficient to perform the matrix multiplications and matrix inversions block-by-block. Then the modified version in Eq. (40) of the iterative enclosure method [27] is used to compute an enclosure of the unknown parameters $\mathbf{v}_h^*$ from the trivial initial guess $\mathbf{v}_h^1 = (A_h^T K_h^{-1}M_h)\Delta\boldsymbol{\delta}_h$.

6 Numerical Benchmark Problems

Our interval inverse algorithm is coded in INTLAB [32], an interval arithmetic extension package developed for the MATLAB environment. To test the performance of the method, we consider parameter identification of the Young's moduli of *i*) a fixed-end bar, *ii*) a truss, *iii*) a simply supported beam, and *iv*) a planar frame. Our numerical results show that our method is able to provide an interval enclosure of the exact parameters. In all solved examples, the Exact Solution (ES) represents the deterministic values for Young's moduli as given data.
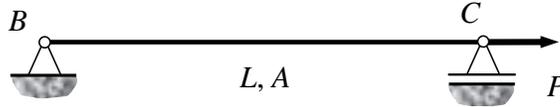
Figure 2: A pin-roller bar subject to concentrated traction at the other end.

6.1 Pin-roller bar

Consider a straight bar of length $L = 5$ m, as shown in Figure 2. The pin-roller bar is subject to a concentrated force $P = 100$ kN at one end C. The cross section of the bar is uniform, with an area $A = 0.005$ m^2. Only axial deformations are allowed, and the bar is modeled by 10 equal-length planar truss elements with uniform material properties. For each element,

$$E = 115 + 10\sin\left(\frac{7x}{L}\right) - 5\cos\left(\frac{17x}{L}\right) \text{ GPa}, \tag{41}$$

where x is the coordinate of element centroid, and the values of E are given up to four significant digits. The same 10-element model is used to generate measurement data.

Table 1: Exact and perturbed measurement data for the pin-roller bar of Figure 2. The device tolerance is the same for all measurements, $\pm 2 \times 10^{-6}$ m, and 3 sets of perturbed measurements are sampled to define the perturbed data.

| Node #	$\tilde{u}_{exact}$ (10^{-3} m)	$\tilde{\mathbf{u}}$ (10^{-3} m)		Difference (10^{-3} m)		Uncertainty (%)	
		Lower Bound	Upper Bound	Lower Bound	Upper Bound	Lower Bound	Upper Bound
1	0.09091	0.090	0.093	-0.001	0.003	-1.100	3.300
2	0.17281	0.172	0.176	-0.001	0.004	-0.579	2.315
3	0.24991	0.247	0.251	-0.003	0.002	-1.200	0.800
4	0.33208	0.331	0.333	-0.002	0.001	-0.602	0.301
5	0.41980	0.419	0.421	-0.001	0.002	-0.238	0.476
6	0.50714	0.505	0.508	-0.003	0.001	-0.592	0.197
7	0.59813	0.598	0.601	-0.001	0.003	-0.167	0.502
8	0.69694	0.696	0.700	-0.001	0.004	-0.143	0.574
9	0.79119	0.790	0.792	-0.002	0.001	-0.253	0.126
10	0.87466	0.873	0.876	-0.002	0.002	-0.229	0.229
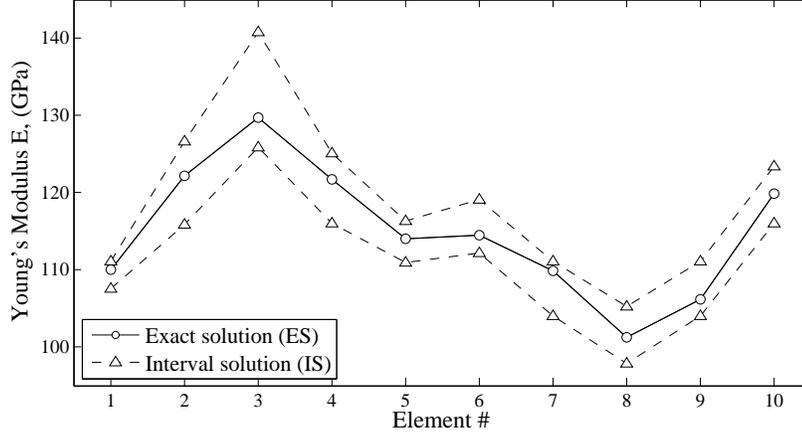
Figure 3: Interval-based identification of Young's moduli of the pin-roller bar of Figure 2: exact values (solid lines with squares) and interval solution (dashed lines with triangles), which is indistinguishable from the Monte Carlo predictions from an ensemble of 10,000 simulations (measurement uncertainty level 0.4-4%).

Axial displacements at 10 equally distributed nodes along the bar are collected into the exact measurement vector $\tilde{u}_{exact}$. The interval measurement vector $\tilde{\mathbf{u}}$ is obtained from 3 sets of perturbed measurements $\tilde{u}_j$ with device tolerance $\pm 2 \times 10^{-6}$ m. The results are listed in Table 1. Note that $\tilde{\mathbf{u}}$ contains $\tilde{u}_{exact}$, and uncertainties in $\tilde{\mathbf{u}}$ range from 0.4% to 4%.

This problem has 10 measurements and 10 unknown element Young's moduli E_j, and it has an analytical solution. Since the bar is statically determined, axial forces in each element equal the concentrated traction P at the free end. Then $\mathbf{E}_j$ depends upon the displacements $\mathbf{u}_j$, $\mathbf{u}_{j-1}$ of the neighboring nodes, viz.

$$N = \mathbf{E}_j A \frac{\mathbf{u}_j - \mathbf{u}_{j-1}}{L_e} \quad \Rightarrow \quad \mathbf{E}_j = \frac{N L_e}{A \left(\mathbf{u}_j - \mathbf{u}_{j-1} \right)}, \tag{42}$$

where $N = P = 100$ kN is the axial force, A is the cross section area, $L_e = L/10$ is the element length, and $\mathbf{u}_0 = 0$ denotes the boundary condition at the hinged end.

The problem is well-posed, so no regularization is required. The initial guess $E = 60$ GPa for all the elements. To reach convergence, 60 iterations are needed in the deterministic stage, and 12 iterations in the interval stage. The estimated and exact solutions are plotted in Figure 3. In the figure, the lower and upper bounds of the estimated solution are the dashed lines with triangular markers, and the exact solution is the solid line with rectangular markers. The exact values of the Young's moduli are contained by the interval bounds.

Table 2 compares the numerical solution $\mathbf{E}_N$ from our method against the analytical solution $\mathbf{E}_A$ from Eq. (42). The upper bounds of the two solutions are identical, while the lower bounds of $\mathbf{E}_N$ are always smaller than the lower bounds of $\mathbf{E}_A$. In other words, $\mathbf{E}_N$ encloses $\mathbf{E}_A$. Exact Young's moduli and relative differences $(E_N - E_A)/E_A \times 100\%$ for the lower and upper bounds of the two interval solutions are also included in the table.

Table 2: Exact Young's moduli and predicted values for the pin-roller bar of Figure 2. Relative differences $(E_N - E_A)/E_A \times 100\%$ for the lower and upper bounds of the two interval solutions are also listed.

Element #	Exact (GPa)	$\mathbf{E}_N$ (GPa)		$\mathbf{E}_A$ (GPa)		Relative Diff. (%)	
		Lower Bound	Upper Bound	Lower Bound	Upper Bound	Lower Bound	Upper Bound
1	110.0	107.47	111.11	107.53	111.11	-0.055	0.000
2	122.1	115.84	126.58	116.28	126.58	-0.376	0.000
3	129.7	125.82	140.85	126.58	140.85	-0.601	0.000
4	121.7	115.96	125.00	116.28	125.00	-0.271	0.000
5	114.0	110.99	116.28	111.11	116.28	-0.105	0.000
6	114.5	112.17	119.05	112.36	119.05	-0.172	0.000
7	109.9	103.94	111.11	104.17	111.11	-0.215	0.000
8	101.2	97.78	105.26	98.04	105.26	-0.262	0.000
9	106.1	103.94	111.11	104.17	111.11	-0.215	0.000
10	119.8	116.06	123.46	116.28	123.46	-0.185	0.000

It is worth mentioning that the row of $(K_h^{-1} M_h)$ in Eq. (38) corresponding to Young's modulus $\mathbf{E}_j$ of the j-th element has all of the entries close to zero, except those at columns corresponding to the measurements $\mathbf{u}_j$ and $\mathbf{u}_{j-1}$ at the neighboring nodes. In addition, the two entries have similar magnitude and opposite sign. This is in agreement with the analytical solution given in Eq. (42); the modulus $\mathbf{E}_j$ of the j-th element is only a function of $\mathbf{u}_j$ and $\mathbf{u}_{j-1}$.

6.2 Simply-supported truss

The second example is a simply supported 15-bar truss, subject to concentrated loads, as shown in Figure 4. Nodes of the truss are labeled from 1 to 9, and the bars are labeled from $\underline{1}$ to $\underline{15}$. We apply horizontal load 60 kN at node 2, vertical load 100 kN at node 3, and horizontal load 30 kN and vertical load 100 kN at node 6. The bars have uniform cross sections with area $A = 0.005$ m^2. Each bar is modeled by one planar truss element with constant material property, and the corresponding Young's modulus is denoted by short bars with circular markers in Figure 5. Here we assume that bar $\underline{3}$ and $\underline{13}$ have been damaged, and their effective Young's moduli are 80 GPa and 60 GPa, respectively.

The same finite element model is used to generate the exact measurement data. To illustrate the performance of the current method under different forms of measurements, nodal displacements of bottom nodes 2 to 5, as well as strains of medium-height
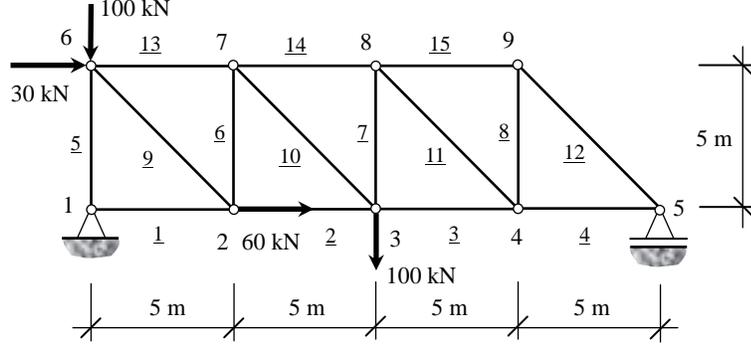
Figure 4: A simply supported truss subject to concentrated loads.

bars $\underline{5}$ to $\underline{12}$, are measured. The device tolerance is $\pm 1 \times 10^{-5}$ m for nodal displacement measurements, and $\pm 1 \times 10^{-6}$ for strain measurements. The measurement vector $\tilde{\mathbf{u}}$ is obtained from 3 sets of perturbed $\tilde{u}_j$, and the results are shown in Table 3. The uncertainties in $\tilde{\mathbf{u}}$ range from 0.06% to 2%, approximately.

This problem has 15 measurement and 15 unknowns. It is well-posed, and no regularizer is needed. The initial guess $E = 60$ GPa is used. 465 iterations are run in the deterministic stage and 12 iterations in the interval stage. In Figure 5, the obtained interval solution (IS) is compared against the exact solution (ES) and Monte

Table 3: Exact and perturbed measurement data for the simply supported truss of Figure 4. The device tolerance is $\pm 1 \times 10^{-5}$ m for nodal displacements and $\pm 1 \times 10^{-6}$ for strains. Three sets of perturbed measurements are sampled to yield the perturbed data.

|	Exact (10⁻³ m)	Perturbed data (10⁻³ m)					Exact (10⁻⁴)	Perturbed data (10⁻⁴)			
		Lower Bound	Uncer-tainty (%)	Upper Bound	Uncer-tainty (%)			Lower Bound	Uncer-tainty (%)	Upper Bound	Uncer-tainty (%)
$\mathbf{u}_2$	0.7557	0.7532	-0.321	0.7586	0.382	$\boldsymbol{\varepsilon}_5$	-2.3246	-2.3306	-0.256	-2.3210	0.155
$\mathbf{v}_2$	-5.1714	-5.1732	-0.036	-5.1591	0.238	$\boldsymbol{\varepsilon}_6$	-0.6822	-0.6827	-0.078	-0.6674	2.161
$\mathbf{u}_3$	1.3922	1.3871	-0.369	1.4021	0.711	$\boldsymbol{\varepsilon}_7$	0.9664	0.9661	-0.025	0.9777	1.167
$\mathbf{v}_3$	-7.6368	-7.6393	-0.032	-7.6349	0.025	$\boldsymbol{\varepsilon}_8$	1.0388	1.0309	-0.769	1.0456	0.648
$\mathbf{u}_4$	2.8297	2.8141	-0.551	2.8310	0.047	$\boldsymbol{\varepsilon}_9$	1.1427	1.1387	-0.346	1.1457	0.265
$\mathbf{v}_4$	-4.3003	-4.3045	-0.097	-4.2914	0.208	$\boldsymbol{\varepsilon}_{10}$	1.1028	1.1000	-0.253	1.1043	0.133
$\mathbf{u}_5$	3.2930	3.2924	-0.019	3.3089	0.482	$\boldsymbol{\varepsilon}_{11}$	-1.4241	-1.4354	-0.795	-1.4213	0.199
						$\boldsymbol{\varepsilon}_{12}$	-1.3736	-1.3748	-0.088	-1.3591	1.058
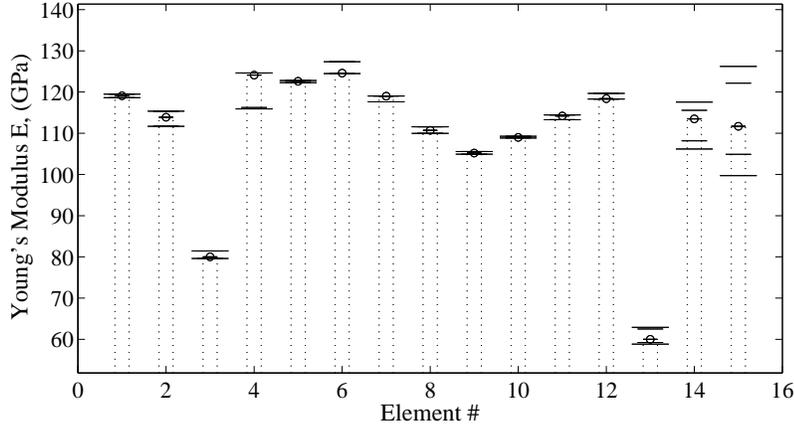
Figure 5: Interval-based identification of Young's moduli of a simply-supported truss of Figure 4: short bars with circular markers denote the exact values; the long bars denote interval prediction from our method; median-length bars with circles denote Monte Carlo predictions from an ensemble of 10,000 simulations.

Carlo (MC) predictions based on an ensemble of 10,000 simulations. In each simulation k, a random measurement vector $\tilde{u}_k$ is chosen within the interval bounds of $\tilde{\mathbf{u}}$, i.e., $\tilde{u}_k \in \tilde{\mathbf{u}}$. The corresponding solution α_k is obtained from the deterministic inverse solver formulated in section 3, and the Monte Carlo solution $\boldsymbol{\alpha}_{MC}$ is given by the minimum and maximum values of all α_k in the ensemble, that is $\boldsymbol{\alpha}_{MC} = [\min_k \alpha_k, \ \max_k \alpha_k]$.

Clearly, both IS and MC predictions enclose the exact values of the Young's moduli, and IS contains MC. It is observed that the interval enclosures of IS are very tight for elements $\underline{5}$ to $\underline{12}$, and very wide for elements $\underline{13}$ to $\underline{15}$. This is caused by the distribution of measurements. The strains of element $\underline{5}$ to $\underline{12}$ and the displacements of the bottom nodes are directly measured. Hence, the estimates on $\mathbf{E}_j$ $(j = 1, \dots, 12)$, especially $\mathbf{E}_j$ $(j = 5, \dots, 12)$, are more accurate than the estimates on $\mathbf{E}_j$ $(j = 13, \dots, 15)$.

Table 4 compares the obtained estimates on the Young's modulus $\mathbf{E}_j$ in detail. In particular, $\mathbf{E}_3$, $\mathbf{E}_4$, $\mathbf{E}_8$, $\mathbf{E}_9$, $\mathbf{E}_{13}$, and $\mathbf{E}_{14}$ are chosen for display. Solutions obtained from our method (IS) and from Monte Carlo prediction (MC) are compared against the reference solution obtained from the nonlinear programming approach (NLP). In NLP, each interval is treated as two inequality constraints, and the lower and upper bounds of the unknown parameters are obtained by solving the corresponding nonlinear programming problems. From Table 4, we observe that:

1. Our method encloses the NLP with little overestimation. In addition, the upper bounds of the estimates are exactly the same as those obtained from NLP.

2. The Monte Carlo prediction obtained from an ensemble of 10,000 simulations is contained by the NLP, thus underestimates the uncertainties.

3. All these methods contain the exact values.

Table 4: Exact and predicted Young's modulus for the simply supported truss of Figure 4. Relative error of the interval solutions from our method and Monte Carlo predictions from an ensemble of 10,000 simulations.

	Young's modulus $\mathbf{E}_3$ (GPa)				Young's modulus $\mathbf{E}_4$ (GPa)			
	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)
Proposed (IS)	80.02	-0.026	81.86	0.000	115.92	-0.253	124.63	0.000
NL Program. (NLP)	80.04	---	81.86	---	116.21	---	124.63	---
Monte Carlo (MC)	80.04	0.001	81.86	-0.001	116.22	0.003	124.62	-0.005
Exact Solution (ES)	80.40	0.453	80.40	-1.781	124.10	6.788	124.10	-0.424
	Young's modulus $\mathbf{E}_8$ (GPa)				Young's modulus $\mathbf{E}_9$ (GPa)			
	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)
Proposed (IS)	109.98	-0.010	111.56	0.000	104.92	-0.002	105.57	0.000
NL Program. (NLP)	109.99	---	111.56	---	104.92	---	105.57	---
Monte Carlo (MC)	109.99	0.000	111.56	0.000	104.92	0.000	105.57	0.000
Exact Solution (ES)	110.70	0.648	110.70	-0.769	105.20	0.265	105.20	-0.346
	Young's modulus $\mathbf{E}_{13}$ (GPa)				Young's modulus $\mathbf{E}_{14}$ (GPa)			
	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)	Lower Bound	Uncer- tainty (%)	Upper Bound	Uncer- tainty (%)
Proposed (IS)	59.61	-0.220	63.77	0.000	106.16	-0.495	117.57	0.000
NL Program. (NLP)	59.74	---	63.77	---	106.68	---	117.57	---
Monte Carlo (MC)	59.79	0.090	63.72	-0.082	106.93	0.227	117.18	-0.330
Exact Solution (ES)	60.80	1.776	60.80	-4.659	113.50	6.389	113.50	-3.458

6.3   Simply-supported beam

The third example is a simply-supported beam subject to uniformly distributed vertical load $q = 100$ kN/m, as shown in Figure 6. The beam has a length $L = 2$ m and a 5 cm $\times$ 3 cm rectangular cross section (cross section area $A = 0.015$ m^2 and moment of inertia $I = 1.125 \times 10^{-4}$ m^4). The beam is subject to lateral deformation, and 20 two-node Euler-Bernoulli beam elements are used in the finite element mesh. The stiffness matrix is computed using the three-node Gaussian quadrature rule. To generated a continuous material field, Young's moduli at the quadrature points are linearly interpolated from those at the material mesh nodes, given by the following
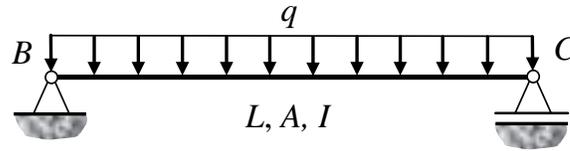
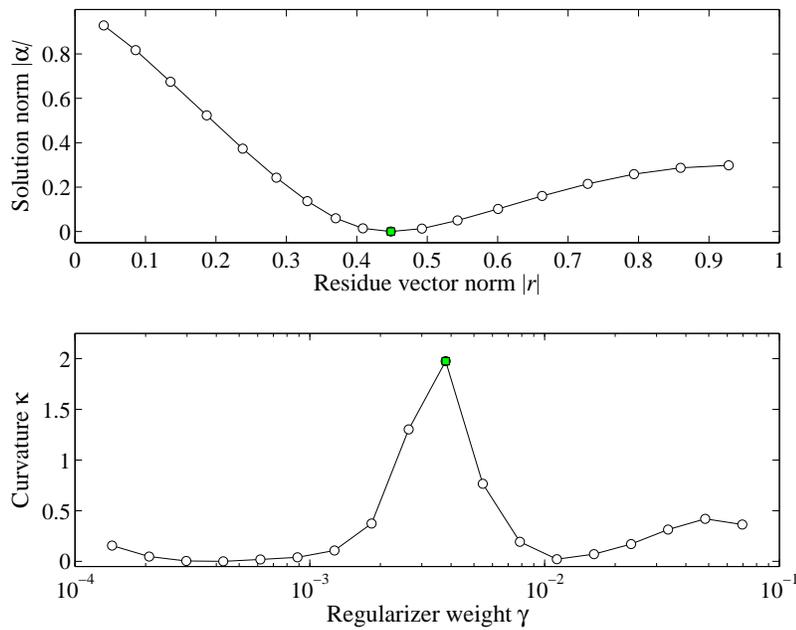Figure 6: A simply supported beam subject to uniformly distributed load.

Figure 7: The L-curve used to find the optimal regularization weight γ: (top) normalized solution vector norm $|\alpha|$ vs. normalized residue vector norm $|r|$, (bottom) curvature κ of the curve in the top subplot vs. the regularization weight γ. Each circular marker corresponds a different weight γ, and the green rectangular marker denotes the optimal weight $\gamma \approx 2 \times 10^{-3}$.

function.

$$E = 220 + 10\sin\left(\frac{6x}{L}\right) - 5\cos\left(\frac{13x}{L}\right) \text{ GPa}, \tag{43}$$

where x is the nodal coordinate, and the values are given up to four significant digits. The stiffness parameter vector $\boldsymbol{\alpha}$ has 21 components, one for each mesh node.

In the first case, a finer 80-element finite element model is used to generate the measurement data. Young's moduli are linearly interpolated from the above-mentioned 21-node material mesh. Further, 9 lateral deflections at equidistant points along the
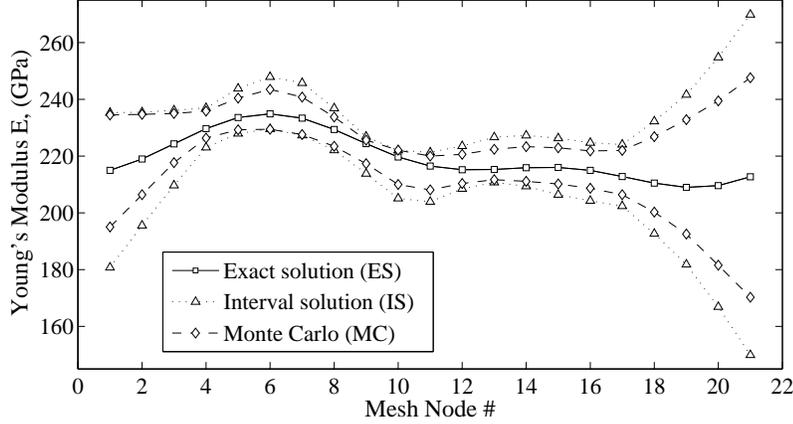
Figure 8: Interval-based identification of the Young's moduli of the simply supported beam of Figure 6 under uniformly distributed load: interval solution (IS), exact solution (ES) and Monte Carlo (MC) prediction from an ensemble of 100,000 simulations (measurement uncertainty level 0.1-1%).

beam are collected as measurements. The measurement vector $\tilde{\mathbf{u}}$, which has 9 components, is obtained from 3 sets of perturbed data $\tilde{u}_j$ with device tolerance $\pm 2 \times 10^{-6}$ m. The resulting $\tilde{\mathbf{u}}$ has uncertainties ranging from 0.1% to 1% and contains the exact measurement data.

The problem is ill-posed, since only 9 measurements are available to estimate 21 unknown parameters. This requires regularization. The regularizer weight γ should be chosen with caution: it has to be large enough to avoid useless estimates or even divergence with unbounded intervals, but not so large that the solution will be over-smooth [17]. Here, a second-order regularization matrix R is used. To determine the optimal γ, the famous L-curve method is used, as shown in Figure 7. According to Figure 7, $\gamma = 2 \times 10^{-3}$ is chosen as the regularization weight.

Then for our method, the initial guess $E = 160$ GPa is given for all components in $\boldsymbol{\alpha}$. Convergence is attained in 289 and 37 iterations in the deterministic and interval stages, respectively. The interval estimates are compared against the exact Young's moduli from Eq. (43) and Monte Carlo predictions from an ensemble of 100,000 simulations. Figure 8 shows the exact solution (ES, solid lines with rectangular markers), the interval solution (IS, dotted lines with triangular markers), and the Monte Carlo prediction (MC, dashed lines with diamond markers). Observe that IS indicates a high level of uncertainty near both ends, especially near the right end, which is attributed to the relatively small bending moment near the ends. In addition, both IS and MC guarantees to enclose ES everywhere, and IS contains MC.

In the second case, two opposing bending moments $M = 50$ kN·m are added to the ends B and C, to create a more uniform bending moment diagram for the beam. In addition, rotation angles $\boldsymbol{\theta}_B$ and $\boldsymbol{\theta}_C$ at both ends are measured. The device tolerance is now $\pm 5 \times 10^{-6}$ m for deflections and $\pm 2 \times 10^{-5}$ rad for $\boldsymbol{\theta}_B$ and $\boldsymbol{\theta}_C$. As a result, the level of uncertainty in $\tilde{\mathbf{u}}$ ranges from 0.1% to 1%, roughly the same as in the first case. IS and MC predictions are compared against the exact values ES in Figure 9. The level
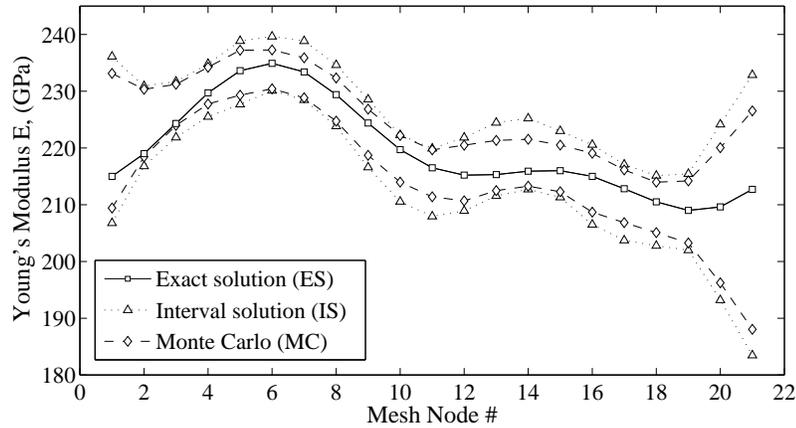
Figure 9: Interval-based identification of Young's moduli of the simply supported beam of Figure 6 under uniformly distributed load and bending moments at both ends: interval solution (IS), exact solution (ES) and Monte Carlo (MC) prediction from an ensemble of 100,000 simulations (measurement uncertainty level 0.1-1%).
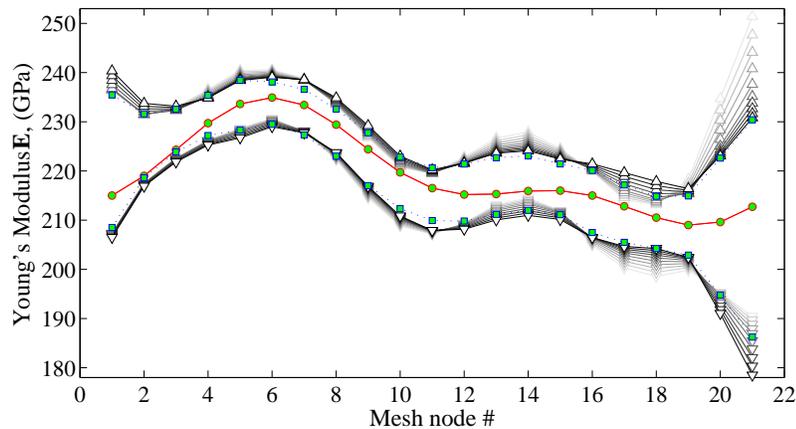

Figure 10: Influence of the reference measurement vector $\tilde{u}_0$ in the interval-based parameter identification of Young's moduli of the simply supported beam of Figure 6: exact solution (solid line with circular marker), Monte Carlo prediction from an ensemble of 100,000 simulations (dotted lines with square markers), lower and upper bounds of the interval solution (solid lines with triangular markers). The lightest lines correspond to the lower bound, viz. $\tilde{u}_0 = \inf \tilde{\mathbf{u}}$ and the darkest lines the upper bound, viz. $\tilde{u}_0 = \sup \tilde{\mathbf{u}}$.

of uncertainty at the ends is reduced significantly because of the additional bending moments at the ends and extra measurements $\boldsymbol{\theta}_B$ and $\boldsymbol{\theta}_C$. Indeed, the maximum level
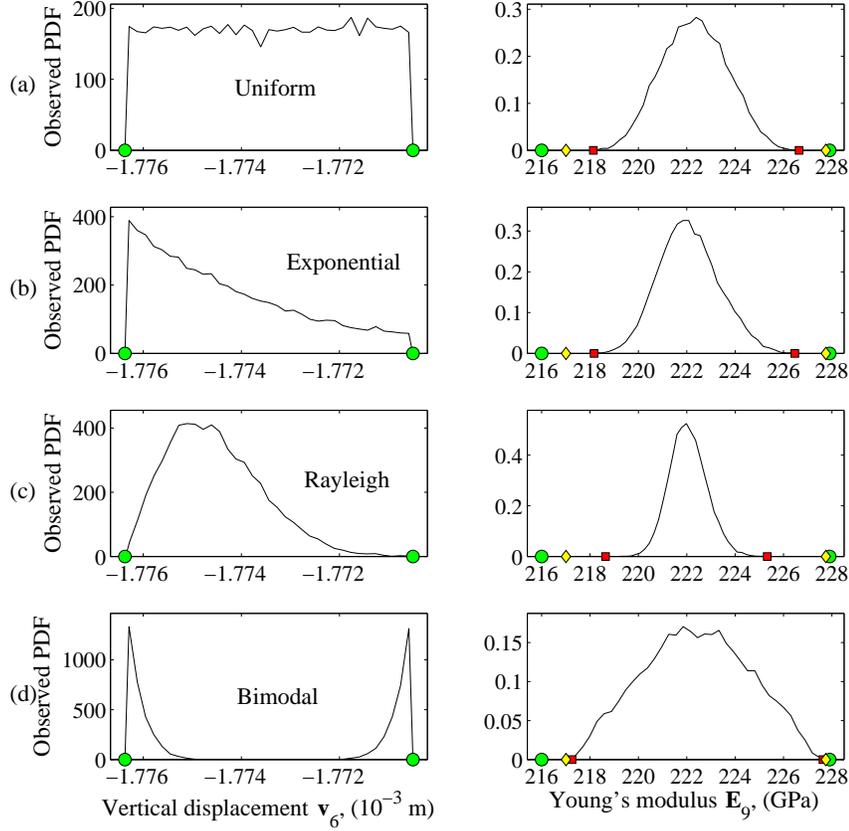
Figure 11: Comparison between the interval solution and Monte Carlo prediction of the Young's modulus $\mathbf{E}_9$ of the simply supported beam of Figure 6 from an ensemble of 100,000 simulations: (left) observed probability density function (PDF) of vertical displacement measurement $\mathbf{v}_6$ sampled from (a) uniform, (b) truncated exponential, (c) truncated Rayleigh, and (d) bimodal probability distributions (interval endpoints denoted by circular markers); (right) corresponding observed PDF of the Young's modulus $\mathbf{E}_9$, interval solution (endpoints denoted by circular markers), nonlinear programming solution (endpoints denoted by diamond markers), and Monte Carlo predicted interval $[\min(E_9) \ \max(E_9)]$ (square markers).

of uncertainty at the ends is approximately 13% on the left and 23% on the right. In the previous case of Figure 8, the uncertainty levels are much higher, approximately 25% on the left and 56% on the right. Near the mid-span, the level of uncertainty is slightly reduced from about 8% in Figure 8 to about 5% in Figure 9.

In the previous discussions, the deterministic reference vector $\tilde{u}_0$ is assumed to be the midpoint of the interval measurement vector $\tilde{\mathbf{u}}$. Figure 10 compares the interval solutions obtained from our method with different choices of $\tilde{u}_0$, from the lower bound $\tilde{u}_0 = \inf \tilde{\mathbf{u}}$ to the upper bound $\tilde{u}_0 = \sup \tilde{\mathbf{u}}$. It is observed that the midpoint values $\tilde{u}_0 = \mathrm{mid}\ \tilde{\mathbf{u}}$ yields the tightest bounds in general.

Finally, note that interval solutions enclose all possible predictions associated with different probabilistic distributions of the measurements, either symmetrical or not (see Figure 11).
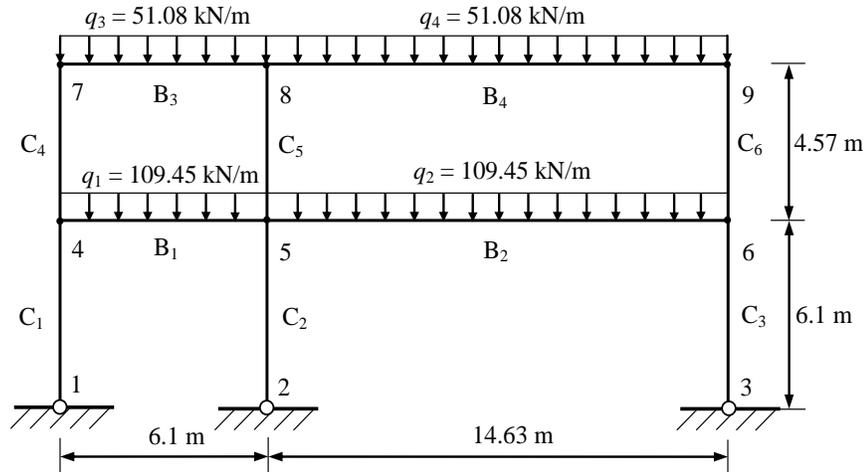

Figure 12: A two-bay two-story frame subject to uniformly distributed loads.

6.4   Two-bay two-story frame

The fourth example is a two-bay two-story planar frame hinged to the ground, subject to uniformly distributed vertical loads on each floor, as shown in Figure 12. The frame is composed of six columns and four beams, labeled as C_j $(j = 1, \ldots, 6)$ and B_j $(j = 1, \ldots, 4)$, respectively. Connecting joints and supports are labeled from 1 to 9. Uniformly distributed vertical loads q_j $(j = 1, \ldots, 4)$ are applied on B_j, where $q_1 = q_2 = 109.45$ kN/m and $q_3 = q_4 = 51.08$ kN/m.

Each member of the frame has uniform cross section and material property. The corresponding cross section area A, moment of inertia I and Young's modulus E are listed in Table 5. Ten two-node Euler-Bernoulli beam elements are used to model the frame, one for each member.

Measurement data used in the inverse algorithm is generated from the same 10-element finite element model. Only nodal displacement $\mathbf{u}_j$, $\mathbf{v}_j$, and rotation angle $\boldsymbol{\theta}_j$ at nodes 4 to 9 $(i = 4, \ldots, 9)$ are included in the measurement vector $\tilde{\mathbf{u}}$. $\tilde{\mathbf{u}}$ is obtained

from 3 sets of perturbed measurements $\tilde{u}_j$, and the corresponding device tolerance is $\pm 2 \times 10^{-5}$ m for nodal displacements and $\pm 2 \times 10^{-5}$ rad for rotation angles. The level of uncertainty in $\tilde{\mathbf{u}}$ ranges from approximately 0.1% to 1%, with the exception of $\boldsymbol{\theta}_4 = [-1.2442, -0.9825] \times 10^{-4}$ rad (22.2% uncertainty).

In this benchmark case, 18 measurements (6 nodes $\times$ 3 DOF) are used to predict the Young's moduli E of the 10 members. The problem is well-posed, and no regularizer is required. Initial guess $E = 160$ GPa is used. The results are compared with the exact values and the Monte Carlo prediction with 10,000 runs in Figure 13, following the same guidelines as in Figure 5 of the simply-supported truss. The interval solution provides an enclosure of both the exact and Monte Carlo solutions.

In Figure 13, the width of the interval estimate $\mathbf{E}_4$ for the Young's modulus of the left column C_4 on the upper floor is much wider than other estimates. The wide enclosure is mainly caused by the displacements $\mathbf{v}_4$ and $\mathbf{v}_7$ at nodes 4 and 7, viz. the vertical displacement of the column C_4. They are modeled by two intervals with about 1% uncertainty, i.e., $\mathbf{v}_4 = [-2.3599, -2.3399] \times 10^{-3}$ m and $\mathbf{v}_7 = [-3.4548, -3.4186] \times 10^{-3}$ m. To obtain a narrower interval prediction for $\mathbf{E}_4$, the accuracy of the measurements $\mathbf{v}_4$ and $\mathbf{v}_7$ is increased, and the level of uncertainty is reduced to about 0.2%, i.e., $\mathbf{v}_4 = [-2.3515, -2.3465] \times 10^{-3}$ m and $\mathbf{v}_7 = [-3.4378, -3.4288] \times 10^{-3}$ m. The results are depicted in Figure 14, showing a significant increase in the accuracy of the predicted value for $\mathbf{E}_4$. In particular, the previous estimate in Figure 13 is $\mathbf{E}_4 = [193.09, 207.39]$ GPa (7.1% uncertainty), and that in Figure 14 is $\mathbf{E}_4 = [197.72, 203.34]$ GPa (2.8% uncertainty).

7   Conclusion

An interval-based parameter identification is presented for structural static problems. Uncertainties in the system are modeled by intervals, and IFEM is exploited to handle uncertainties. Our inverse algorithm stems from an adjoint-based optimization formulation, and it provides an interval estimate of the unknown parameters (e.g., Young's moduli). The associated nonlinear interval equations are solved by a new variant of

Table 5: Geometric and material properties for the members of the two-bay two-story frame shown in Figure 12.

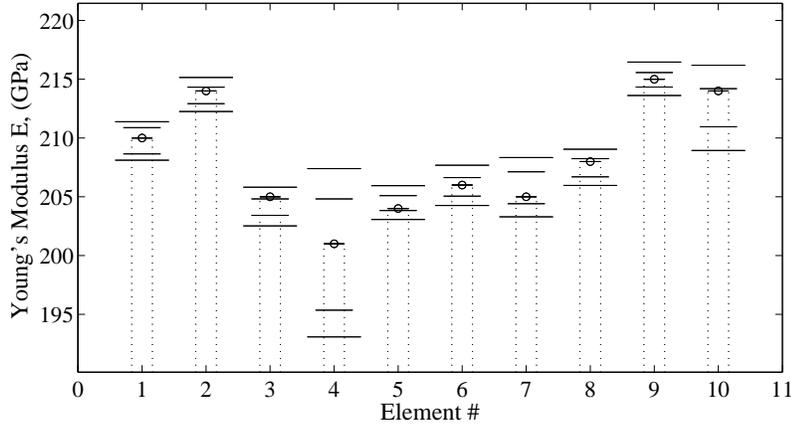|	Shape	A (10⁻⁴ m²)	I (10⁻⁸ m⁴)	E (GPa)		Shape	A (10⁻⁴ m²)	I (10⁻⁸ m⁴)	E (GPa)
C_1	W12×19	35.940	5411.00	210	B_1	W27×84	160.000	118625.96	205
C_2	W14×132	250.320	63683.41	214	B_2	W36×135	256.130	324660.51	208
C_3	W14×109	206.450	51612.70	205	B_3	W18×40	76.130	25473.36	215
C_4	W10×12	22.835	2239.32	201	B_4	W27×94	178.710	136107.68	214
C_5	W14×109	206.450	51612.70	204					
C_6	W14×109	206.450	51612.70	206					

Figure 13: Interval-based identification of Young's moduli of the two-bay two-story frame in Figure 12: short bars with circular markers denote the exact values; long bars denote interval predictions from our method; median-length bars denote the Monte Carlo predictions from an ensemble of 10,000 simulations (measurement uncertainty level 0.1-1%).
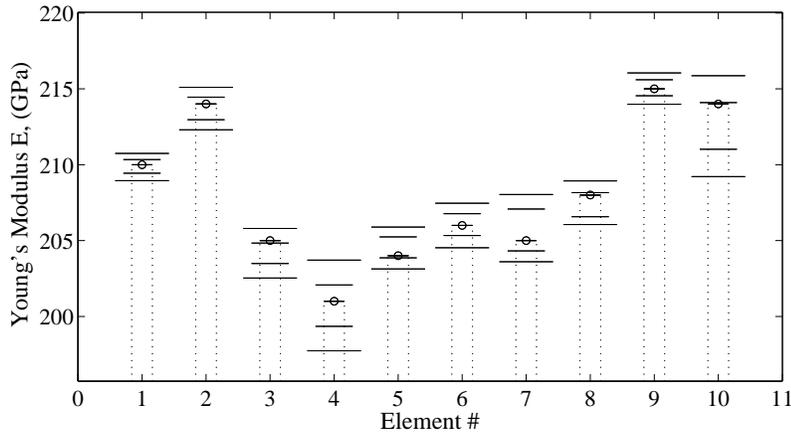

Figure 14: Interval-based identification of the Young's moduli of the two-bay two-story frame in Figure 12 using more accurate measurements in $\mathbf{v}_4$ and $\mathbf{v}_7$ than those used in Figure 13: short bars with circular markers denote the exact values; long bars denote interval prediction from our method; median-length bars denote the Monte Carlo prediction from an ensemble of 10,000 simulations (measurement uncertainty level 0.1-1%).

the iterative enclosure method. In addition, overestimation is reduced by means of a new decomposition of the IFEM matrices K and f, which limits multiple occurrences of the same variable in the IFEM equations by separating deterministic and interval

terms. The interval solution from our solver guarantees enclosure of the exact parameters, as confirmed by several numerical benchmark problems, and it always contains Monte Carlo predictions.

References

[1] Hajime Akashi, Hiroyuki Imai, and Kamal A.F. Moustafa. Parameter identification technique for multivariate stochastic systems. *Automatica*, 15(2):217–221, 1979.

[2] Gotz Alefeld and Jurgen Herzberger. *Introduction to Interval Computation*. Academic Press, 1984.

[3] Neculai Andrei. Acceleration of conjugate gradient algorithms for unconstrained optimization. *Applied Mathematics and Computation*, 213(2):361–369, 2009.

[4] Robert Grover Brown and Patrick Y.C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley and Sons, New York, 1997.

[5] Peter C. Chang, Alison Flatau, and S.C. Liu. Review paper: health monitoring of civil infrastructure. *Structural Health Monitoring*, 2(3):257–267, 2003.

[6] Robert D. Cook, David S. Malkus, Michael E. Plesha, and Robert J. Witt. *Concepts and Applications of Finite Element Analysis*. John Wiley & Sons, 2007.

[7] Xiuyun Du, Zhenan Tang, and Qiwen Xue. Interval inverse analysis of hyperbolic heat conduction problem. *International Communications in Heat and Mass Transfer*, 54:75–80, 2014.

[8] M.J. Eppstein, F. Fedele, J. Laible, C. Zhang, A. Godavarty, and E.M. Sevick-Muraca. A comparison of exact and approximate adjoint sensitivities in fluorescence tomography. *IEEE Transactions on Medical Imaging*, 22(10):1215–1223, 2003.

[9] Yildirim Serhat Erdogan and Pelin Gundes Bakir. Inverse propagation of uncertainties in finite element model updating through use of fuzzy arithmetic. *Engineering Applications of Artificial Intelligence*, 26(1):357–367, 2013.

[10] F. Fedele, J.P. Laible, and M.J. Eppstein. Coupled complex adjoint sensitivities for frequency-domain fluorescence tomography: theory and vectorized implementation. *Journal of Computational Physics*, 187(2):597–619, 2003.

[11] Francesco Fedele, Rafi L. Muhanna, Naijia Xiao, and Robert L. Mullen. Interval-based approach for uncertainty propagation in inverse problems. *Journal of Engineering Mechanics*, 141(1), 2014.

[12] Andreas Fichtner. *Full Seismic Waveform Modelling and Inversion*. Springer Science & Business Media, 2010.

[13] Reeves Fletcher and Colin M. Reeves. Function minimization by conjugate gradients. *The computer journal*, 7(2):149–154, 1964.

[14] Steven D. Glaser, Hui Li, Ming L. Wang, Jinping Ou, and Jerome Lynch. Sensor technology innovation for the advancement of structural health monitoring: a strategic program of US-China research for the next decade. *Smart Structures and Systems*, 3(2):221–244, 2007.

[15] T. Haag, J. Herrmann, and M. Hanss. Identification procedure for epistemic uncertainties using inverse fuzzy arithmetic. *Mechanical Systems and Signal Processing*, 24(7):2021–2034, 2010.

[16] Xue Han, Wenyu Sun, and Chuangyin Dang. Nonmonotone second-order Wolfe's line search method for unconstrained optimization problems. *Computers & Mathematics with Applications*, 60(9):2517–2525, 2010.

[17] Per Christian Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, Technical University of Denmark, Lyngby, Denmark, Copyright 2010 by the Society for Industrial and Applied Mathematics.

[18] Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, 1952.

[19] C. Jiang, Z. Zhang, X. Han, and J. Liu. A novel evidence-theory-based reliability analysis method for structures with epistemic uncertainty. *Computers & Structures*, 129:1–12, 2013.

[20] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(D):35–45, 1960.

[21] Hamed Haddad Khodaparast, John E. Mottershead, and Kenneth J. Badcock. Interval model updating with irreducible uncertainty using the Kriging predictor. *Mechanical Systems and Signal Processing*, 25(4):1204–1226, 2011.

[22] U.W. Kulisch and W.L. Miranker. *Computer Arithmetic in Theory and Practice*. Academic Press, ANew York, 1981.

[23] Ramon E. Moore, R. Baker Kearfott, and Michael J. Cloud. *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, 2009.

[24] Rafi L. Muhanna and Hao Zhang. Interval finite elements as a basis for generalized models of uncertainty in engineering mechanics. *Reliable computing*, 13:173–194, 2007.

[25] Rafi L. Muhanna, Hao Zhang, and Robert L. Mullen. Combined axial and bending stiffness in interval finite-element methods. *Journal of Structural Engineering*, 133(12):1700–1709, 2007.

[26] Robert L. Mullen and Rafi L. Muhanna. Bounds of structural response for all possible loading combinations. *Journal of Structural Engineering*, 125(1):98–106, 1999.

[27] Arnold Neumaier and Andrzej Pownuk. Linear systems with large uncertainties, with applications to truss structures. *Reliable Computing*, 13(2):149–172, 2007.

[28] E. Polak and G. Ribière. Note sur la convergence de directions conjugèes. *Rev. Francaise Informat Recherche Operationelle*, 16:35–43, 1969.

[29] B.T. Polyak. The conjugate gradient method in extreme problems. *USSR Computational Mathematics and Mathematical Physics*, 9:94–112, 1969.

[30] Alexander G. Ramm. *Inverse Problems: Mathematical and Analytical Techniques with Applications to Engineering*. Springer, 2005.

[31] M.V. Rama Rao, Robert L. Mullen, and Rafi L. Muhanna. A new interval finite element formulation with the same accuracy in primary and derived variables. *International Journal of Reliability and Safety*, 5(3):336–357, 2011.

[32] Siegfried M. Rump. INTLAB - INTerval LABoratory. In Tibor Csendes, editor, *Developments in Reliable Computing*, pages 77–104. Kluwer Academic Publishers, Dordrecht, 1999. `http://www.ti3.tuhh.de/rump/`.

[33] J. Carlos Santamarina and Dante Fratta. *Discrete Signals and Inverse Problems: An Introduction for Engineers and Scientists*. John Wiley & Sons, Ltd, 2005.

[34] Zhen-Jun Shi and Jie Shen. A gradient-related algorithm with inexact line searches. *Journal of computational and applied mathematics*, 170(2):349–370, 2004.

[35] D. Simon. *Optimal State Estimation: Kalman, H_∞, and Nonlinear Approaches*. John Wiley & Sons, Inc., 2006.

[36] Christopher J. Stull, Jonathan M. Nichols, and Christopher J. Earls. Stochastic inverse identification of geometric imperfections in shell structures. *Computer Methods in Applied Mechanics and Engineering*, 200(25-28):2256–2267, 2011.

[37] Ling Wang, Ye Xu, and Lingpo Li. Parameter identification of chaotic systems by hybrid Nelder-Mead simplex search and differential evolution algorithm. *Expert Systems with Applications*, 38:3238–3245, 2011.

[38] Z. Xie and J. Feng. Real-time nonlinear structural system identification via iterated unscented Kalman filter. *Mechanical Systems and Signal Processing*, 28:309–322, 2012.

[39] Gaohang Yu, Lutai Guan, and Zengxin Wei. Globally convergent Polak–Ribière–Polyak conjugate gradient methods under a modified Wolfe line search. *Applied Mathematics and Computation*, 215(8):3082–3090, 2009.

[40] Li Zhang and Junli Li. A new globalization technique for nonlinear conjugate gradient methods for nonconvex minimization. *Applied Mathematics and Computation*, 217(24):10295–10304, 2011.