Finite Element Model Updating of an 18-Story Structure using Branch-and-Bound Algorithm with Epsilon-Constraint

Yu Otsuki[1], Dan Li[1], Santanu S. Dey[2], Masahiro Kurata[3], Yang Wang[1, 4, *]

[1] *School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, USA*

[2] *School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA*

[3] *Disaster Prevention Research Institute, Kyoto University, Uji, Kyoto, Japan*

[4] *School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA*

**yang.wang@ce.gatech.edu*

Abstract: This paper studies the finite element (FE) model updating of an 18-story experimental structure. FE model updating requires solving optimization problems that are generally non-convex and have unknown number of local optima. For such problems, neither randomized local optimization algorithms nor stochastic search algorithms can guarantee global optimality. To obtain the global optimum and improve the accuracy of FE model updating, this paper proposes the branch-and-bound (B&B) algorithm for solving non-convex optimization problems in FE model updating. The paper focuses on the modal property difference formulation that minimizes the difference between experimental and simulated eigenvalues and eigenvectors. We propose a reformulation of the modal property difference approach using epsilon-constraint, in order to enable the application of the B&B algorithm in FE model updating. The proposed approach is first investigated in simulation and compared with the interior-point method and the genetic algorithm. The model updating results using the B&B algorithm are next validated by the shaking table test data of an 18-story steel frame structure.

Keywords: finite element model updating, branch-and-bound algorithm, non-convex optimization, global optimization, epsilon-constraint method, shaking table test

Declarations

Funding: This research was partially funded by the National Science Foundation (CMMI-1634483). The first author received scholarship support from the Nakajima Foundation.

Conflicts of interest: Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the view of the sponsors.

Availability of data and material: The datasets used during the current study are available from the corresponding author on reasonable request.

Code availability: Part of the code that supports the findings of this study is available at https://github.com/ywang-structures/Structural-Model-Updating

1 Introduction

With increasingly available experimental data from as-built structures, techniques for finite element (FE) model updating have been developed and widely adopted over the past few decades. FE model updating refers to the fine-tuning of the parameter values in an FE model to better match the behavior of an as-built structure. Two broad categories in FE model updating are the deterministic and probabilistic approaches, while this research focuses on the deterministic approach in frequency-domain [1-3]. The deterministic model updating approach seeks an optimal solution by solving optimization problems based on experimentally extracted modal properties of as-built structures. One common method in frequency-domain deterministic model updating is the modal property difference formulation, which directly minimizes the difference between experimental and simulated modal properties. Early researchers have attempted to minimize the difference of eigenvalues/frequencies in FE model updating, and realized that considering only the eigenvalue difference may not lead to accurate model updating results [4]. Later researchers attempted to include other modal properties in the optimization objective, such as the modal assurance criterion (MAC) values [1, 5], eigenvectors [6, 7], and modal flexibilities [8]. Another formulation widely used in frequency-domain deterministic model updating is the modal dynamic residual formulation, which minimizes the residuals of eigenvalue equations from structural dynamics [9-11]. On the other hand, uncertainties exist in FE model updating due to the imperfection of FE models and the nature of experimental measurements. To this end, probabilistic model updating finds multiple solutions in a probabilistic framework and quantifies uncertainties in the updated parameters [12, 13]. The effect and treatment of uncertainties on FE model updating have been investigated by researchers [14, 15].

In general, optimization problems in FE model updating are non-convex with unknown number of local optima [16]. For such problems, depending on the search starting points, off-the-shelf local optimization algorithms may be trapped at local optima that are far away from the global optimum. In the context of FE model updating, such local optima may lead to approximate models that are unable to accurately describe the response of as-built structures. Researchers in FE model updating have attempted to seek a better solution through either randomized local search or stochastic global search. One commonly used gradient-based local optimization algorithm is the interior-point method which can deal with nonlinear objective function and constraints [17]. Randomized search from numerous starting points is reported to find better local optima when coupled with a local optimizer [18, 19]. In addition, stochastic search algorithms have been widely investigated in FE model updating, such as genetic algorithm, simulated annealing,

evolutionary algorithm, and the artificial bee colony algorithm [20-22]. These stochastic search algorithms are less likely to be trapped in local optima but none can guarantee global optimality of the solution.

In contrast to both randomized local search and stochastic global search approaches, some recent development in deterministic global optimization algorithms shows promising performance by taking advantage of the mathematical structure of the problem and guarantees global optimality of the solution. Li *et al* [23] applied the sum of squares (SOS) deterministic global optimization algorithm on the modal dynamic residual formulation, owing to the polynomial nature of the objective function and constraints. While general global optimization algorithms are computationally expensive, the computational efficiency of the SOS algorithms has been improved utilizing the sparsity of structural matrices [24] and facial reduction technique [25]. Nevertheless, the SOS algorithms cannot be applied to optimization problems other than polynomial optimization problems with more variables in the objective function than in the constraints, which have limited engineering applications. Current validation studies have also been limited to problems with very small size.

Another deterministic global optimization algorithm widely used in many engineering fields is the branch-and-bound (B&B) algorithm [26, 27]. The B&B algorithm computes the lower and upper bounds of the global optimum of the objective function over partitioned subregions of the entire feasible region. Here, "branch" refers to the successive subdivision of the feasible region and "bound" refers to the computation of lower and upper bounds for the global optimum. When the difference between the upper and lower bounds of the global optimum is within a specified tolerance, the algorithm terminates at the solution that is guaranteed to be within the tolerance of the global optimum. The main feature of the B&B algorithm is its capability to efficiently prune the feasible region that does not contain the global optimum. Over the decades, the B&B algorithm has been one of the most well-known methods for solving non-convex optimization problems in many engineering fields [28] such as the pooling problem [27], process design [29], path planning [30], sensor placement [31], control design [32], structural design [33], and so on. However, the application of the B&B algorithm on FE model updating has been scarce. Kurata *et al* [5] heuristically incorporated the pruning technique of the B&B algorithm into the Bayesian model updating to efficiently reduce the feasible region. The modal dynamic residual formulation in FE model updating has been used as an example to evaluate the performance of the novel B&B algorithm in simulation. The algorithm successfully found the global optimum [34]. However, experimental validation of FE model updating using the B&B algorithm has not been made yet.

To obtain the global optimum and improve the accuracy of FE model updating, this research investigates the B&B algorithm to solve the non-convex optimization problems in FE model updating. Unlike the modal dynamic residual formulation previously used for the SOS global optimization algorithm [23-25], this research focuses on the modal property difference formulation that minimizes the difference between experimental and simulated modal properties. To make the B&B algorithm applicable, this paper proposes the reformulation of the modal property difference formulation utilizing the epsilon-constraint method. The proposed method is first compared in simulation with a randomized local optimization algorithm and a stochastic global search algorithm. The method is finally validated using the shaking table test data of an 18-story steel frame structure.

The rest of the paper is organized as follows. Section 2 introduces the modal property difference formulation with the epsilon-constraint method. Section 3 reviews the B&B algorithm procedure to solve non-convex optimization problems. Section 4 presents the simulation study and the experimental validation of the proposed method by the shaking table test data of an 18-story steel frame structure. In the end, Section 5 provides conclusions and future work.

2 Model updating formulations

FE model updating attempts to identify the suitable structural parameter values such as stiffness and mass. Note that if all the stiffness and mass parameters are updated simultaneously, frequency-domain FE model updating suffers from non-uniqueness of the solution [35]. Like most studies, for brevity, we assume accurate mass information and update the stiffness parameters of a proportionally damped FE model. In addition, no support/foundation springs have been allocated in a model used in this study. The stiffness matrix of an N-degrees of freedom (DOFs) linear FE model can be represented by:

$$\mathbf{K}(\boldsymbol{\alpha}) = \mathbf{K}_0 + \sum_{j=1}^{n_{\boldsymbol{\alpha}}} \alpha_j \mathbf{K}_j \tag{1}$$

where $\mathbf{K}_0 \in \mathbb{R}^{N\times N}$ is the initial (nominal) stiffness matrix; $n_{\boldsymbol{\alpha}}$ is the number of stiffness updating variables; α_j is the j-th entry of the updating vector variable $\boldsymbol{\alpha} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$, which represents the relative change of a stiffness parameter from the initial value; $\mathbf{K}_j \in \mathbb{R}^{N\times N}$ is the influence matrix corresponding to α_j.

The formulation considered in this research is based on the generalized eigenvalue problem in structural dynamics:

$$[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]\{\boldsymbol{\psi}_i\} = \mathbf{0}, \quad i = 1 \dots n_{\text{modes}} \tag{2}$$

where $\mathbf{M} \in \mathbb{R}^{N \times N}$ denotes the mass matrix; $\lambda_i \in \mathbb{R}$ and $\boldsymbol{\psi}_i \in \mathbb{R}^N$ are the *i*-th eigenvalue and eigenvector, respectively; n_{modes} denotes the number of modes. Note that λ_i and $\boldsymbol{\psi}_i$ implicitly depend on $\boldsymbol{\alpha}$, and thus can be denoted as $\lambda_i(\boldsymbol{\alpha})$ and $\boldsymbol{\psi}_i(\boldsymbol{\alpha})$. In general, eigenvectors obtained through field testing are limited to the DOFs measured by sensors. To reflect the measured DOFs, we assume the DOFs are rearranged properly and define $\boldsymbol{\psi}_i(\boldsymbol{\alpha}) = [\boldsymbol{\psi}_i^{\mathcal{M}}(\boldsymbol{\alpha}),\ \boldsymbol{\psi}_i^{\mathcal{U}}(\boldsymbol{\alpha})]$, where $\boldsymbol{\psi}_i^{\mathcal{M}}(\boldsymbol{\alpha}) \in \mathbb{R}^{n_{\mathcal{M}}}$ represents the eigenvector entries corresponding to the measured DOFs and $\boldsymbol{\psi}_i^{\mathcal{U}} \in \mathbb{R}^{n_{\mathcal{U}}}$ corresponds to the unmeasured DOFs. Note that $n_{\mathcal{M}} + n_{\mathcal{U}} = N$, the total number of DOFs. In this paper, experimentally obtained eigenvalues and eigenvectors are denoted as λ_i^{EXP} and $\boldsymbol{\psi}_i^{\text{EXP},\mathcal{M}}$ (at measured DOFs), respectively.

2.1 Modal property difference formulation: eigenvector difference approach

The modal property difference formulation with eigenvector difference approach attempts to minimize the difference between experimental and simulated eigenvalues and eigenvectors [7]:

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize}} \quad & \sum_{i=1}^{n_{\text{modes}}} \left\{ \left\| \frac{\lambda_i^{\text{EXP}} - \lambda_i(\boldsymbol{\alpha})}{\lambda_i^{\text{EXP}}} \cdot w_{\lambda_i} \right\| + \left\| \left\{ \boldsymbol{\psi}_{-q_i,i}^{\text{EXP},\mathcal{M}} - \boldsymbol{\psi}_{-q_i,i}^{\mathcal{M}}(\boldsymbol{\alpha}) \right\} \cdot w_{\boldsymbol{\psi}_i} \right\| \right\} \\ \text{subject to} \quad & \mathbf{L}_{\boldsymbol{\alpha}} \le \boldsymbol{\alpha} \le \mathbf{U}_{\boldsymbol{\alpha}} \end{aligned} \tag{3}$$

where $\|\cdot\|$ denotes the norm of a vector; w_{λ_i} represents the weighting factor of the *i*-th eigenvalue difference; $w_{\boldsymbol{\psi}_i}$ represents the weighting factor of the *i*-th eigenvector difference; $\mathbf{L}_{\boldsymbol{\alpha}}$ and $\mathbf{U}_{\boldsymbol{\alpha}} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$ denote the lower and upper bounds for the variable $\boldsymbol{\alpha}$, respectively. The experimental eigenvector at the measured DOFs $\boldsymbol{\psi}_i^{\text{EXP},\mathcal{M}}$ is normalized so that the entry with the largest absolute value of $\boldsymbol{\psi}_i^{\text{EXP},\mathcal{M}}$, denoted as the q_i-th entry, equals to 1. Accordingly, the simulated eigenvector at the measured DOFs $\boldsymbol{\psi}_i^{\mathcal{M}}(\boldsymbol{\alpha})$ is also normalized so that the q_i-th entry equals to 1. $\boldsymbol{\psi}_{-q_i,i}^{\text{EXP},\mathcal{M}}$ and $\boldsymbol{\psi}_{-q_i,i}^{\mathcal{M}}(\boldsymbol{\alpha}) \in \mathbb{R}^{n_{\mathcal{M}}-1}$ represent the eigenvectors at the measured DOFs with the q_i-th entry removed.

At every iteration of an optimization gradient search, the generalized eigenvalue problem in Eq. (2) is solved using values of $\boldsymbol{\alpha}$ at the current iteration, which produces simulated eigenvalues $\lambda_i(\boldsymbol{\alpha})$ and eigenvectors $\boldsymbol{\psi}_i(\boldsymbol{\alpha})$. This process determines that with implicit functions $\lambda_i(\boldsymbol{\alpha})$ and $\boldsymbol{\psi}_i(\boldsymbol{\alpha})$, the objective function in Eq. (3) is implicit on $\boldsymbol{\alpha}$. For the $\boldsymbol{\alpha}$ value at current iteration, a gradient search algorithm evaluates the objective function value in Eq. (3) and

calculates Jacobian derivative to find next search gradient. The iteration continues until convergence criteria are satisfied. Besides having an implicit objective function, the eigenvector difference approach provides a non-convex optimization problem for which the global optimality of the solution cannot be guaranteed using gradient search algorithms.

2.2 Epsilon-constraint method for modal property difference formulation

To apply the B&B algorithm, the optimization problem in Eq. (3) first needs to be reformulated to an explicit form on the optimization variables. For this purpose, we apply the epsilon-constraint method widely used in multi-objective optimization problems where one of the objective functions is converted into constraints using a constant value ε [36]. For convenience, we introduce notation $\mathbf{1}$ as a vector with all entries equal to 1 and with compatible length. In this research, we use the epsilon-constraint method to convert the implicit constraints of eigenvalue equations (Eq. (2)) to be explicit:

$$\begin{aligned}
\underset{\boldsymbol{\alpha},\boldsymbol{\lambda},\boldsymbol{\psi}}{\text{minimize}} \quad & \sum_{i=1}^{n_{\text{modes}}} \left\{ \left\| \frac{\lambda_i^{\text{EXP}} - \lambda_i}{\lambda_i^{\text{EXP}}} \cdot w_{\lambda_i} \right\| + \left\| \left\{ \boldsymbol{\psi}_{-q_i,\,i}^{\text{EXP},\mathcal{M}} - \boldsymbol{\psi}_{-q_i,\,i}^{\mathcal{M}} \right\} \cdot w_{\boldsymbol{\psi}_i} \right\| \right\} \\
\text{subject to} \quad & -\varepsilon \cdot \mathbf{1} \leq [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]\{\boldsymbol{\psi}_i\} \leq \varepsilon \cdot \mathbf{1} \qquad i = 1, \dots, n_{\text{modes}} \\
& \mathbf{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \mathbf{U}_{\boldsymbol{\alpha}} \\
& L_{\lambda_i} \leq \lambda_i \leq U_{\lambda_i}, \qquad i = 1, \dots, n_{\text{modes}} \\
& \mathbf{L}_{\boldsymbol{\psi}_i} \leq \boldsymbol{\psi}_i \leq \mathbf{U}_{\boldsymbol{\psi}_i}, \qquad i = 1, \dots, n_{\text{modes}}
\end{aligned} \tag{4}$$

where $\|\cdot\|$ denotes the norm of a vector; ε is a small (pre-specified) constant value; $\lambda_i \in \mathbb{R}$ and $\boldsymbol{\psi}_i = [\boldsymbol{\psi}_i^{\mathcal{M}},\ \boldsymbol{\psi}_i^{\mathcal{U}}] \in \mathbb{R}^N$ are the optimization variables corresponding to the i-th eigenvalue and eigenvectors, respectively. Recall the $\mathbf{K}(\boldsymbol{\alpha})$ defined in Eq. (1) is explicit and affine on $\boldsymbol{\alpha}$. Contrary to the implicit function of $\lambda_i(\boldsymbol{\alpha})$ and $\boldsymbol{\psi}_i(\boldsymbol{\alpha})$ in Eq. (3), Eq. (4) adopts λ_i and $\boldsymbol{\psi}_i$ as optimization variables so that the formulation becomes explicit in terms of optimization variables $\boldsymbol{\alpha}$, $\boldsymbol{\lambda}$, and $\boldsymbol{\psi}$. The formulation is non-convex due to the bilinear terms $\mathbf{K}(\boldsymbol{\alpha})\{\boldsymbol{\psi}_i\}$ and $\lambda_i\{\boldsymbol{\psi}_i\}$ in the constraints. In Eq. (4), ε value governs the extent to which the generalized eigenvalue equation is satisfied. A small ε value ensures to nearly satisfy the eigenvalue equation in the constraints, making Eq. (4) more equivalent to Eq. (3). However, a small ε value also dictates a more restricted feasible region for which optimization algorithms may have difficulty in finding a feasible solution. On the other hand, a larger ε value may result in an optimal solution that does not satisfy the

eigenvalue equation (Eq. (2)) to a desirable accuracy. Therefore, trial-and-error is necessary in determining an appropriate value for ε.

3 Branch-and-Bound algorithm

This paper proposes the branch-and-bound (B&B) algorithm to solve the non-convex optimization problem in Eq. (4). To describe the B&B algorithm, consider a general non-convex optimization problem (*P*):

$$
\begin{aligned}
(P): \quad & \underset{\mathbf{x}}{\text{minimize}} && f(\mathbf{x}) \\
& \text{subject to} && g_i(\mathbf{x}) \le 0, \;\; i = 1, \dots, n_g \\
& && h_j(\mathbf{x}) = 0, \;\; j = 1, \dots, n_h \\
& && \mathbf{L_x} \le \mathbf{x} \le \mathbf{U_x}
\end{aligned}
\tag{5}
$$

where $f: \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$, $g_i: \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$, and $h_j: \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$ are the functions for the optimization vector variable $\mathbf{x} \in \mathbb{R}^{n_\mathbf{x}}$. $\mathbf{L_x}$ and $\mathbf{U_x}$ denote the lower and upper bounds of the optimization vector variable $\mathbf{x}$, respectively. The feasible region of the problem (*P*) is defined as $\mathcal{X} = \{\mathbf{x} | \mathbf{L_x} \le \mathbf{x} \le \mathbf{U_x},\; g_i(\mathbf{x}) \le 0\,, h_j(\mathbf{x}) = 0,\;\; i = 1, \dots, n_g,\; j = 1, \dots, n_h\}$.

Figure 1 illustrates the concept behind the B&B algorithm. The illustration shows a univariate non-convex function $f(x): \mathbb{R} \to \mathbb{R}$ with two local minima within the feasible region $\mathcal{X}$. We denote the global minimum of the objective function value as f^* and the corresponding optimal solution as x^*, i.e. $x^* \in \mathcal{X}$ and $f^* = f(x^*)$. At the initialization iteration step $k = 0$, the B&B algorithm performs the convex relaxation of the original problem. This step is called "bounding", shown as Step 1 in Figure 1. The relaxed problem is convex and thus can be easily solved. The minimum value of the relaxed problem, denoted as $LB^{(0)}$, provides the lower bound for f^*, the global minimum of the original problem. Any feasible solution of the original optimization problem apparently provides an upper bound of f^*, denoted as $UB^{(0)}$ at iteration $k = 0$. Hence, in Step 2 of Figure 1, $UB^{(0)}$ can be obtained using a local optimization algorithm (e.g. gradient search) or other heuristics approaches. Notice that $UB^{(0)}$ is one local minimum and is not guaranteed to be the global minimum at this stage. Now, the global optimal value f^* is guaranteed to be between $LB^{(0)}$ and $UB^{(0)}$. If $\left|UB^{(0)} - LB^{(0)}\right|$ is within a specified tolerance ξ, the algorithm terminates. Otherwise, the iteration proceeds to $k = 1$; the feasible region is subdivided into subregions, $\mathcal{R}_1$, and $\mathcal{R}_2$ such that $\mathcal{X} = \mathcal{R}_1 \cup \mathcal{R}_2$, following a "branching rule". Note that the branching at the local minimum in Figure 1 is simply for illustration.

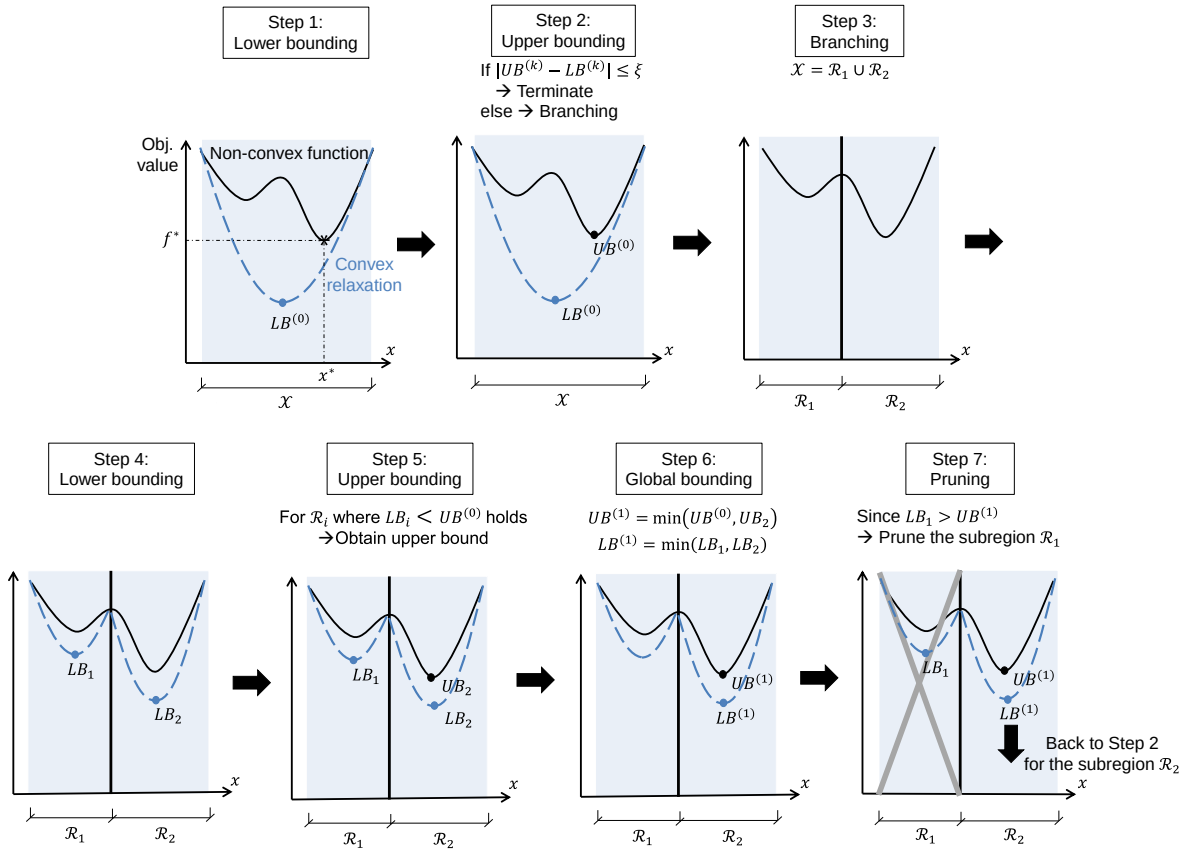


Figure 1. Illustration of the branch-and-bound algorithm

Detailed branching techniques can be found in references such as [27] (Step 3). For each subregion, the corresponding convex relaxation problem is again constructed and solved. The lower bound is denoted as LB_1 for the subregion $\mathcal{R}_1$, and as LB_2 for the subregion $\mathcal{R}_2$ (Step 4). Next, for the subregions $\mathcal{R}_i$ where $LB_i < UB^{(0)}$ holds, a feasible upper bound solution is found as UB_i. In this illustration, since $LB_2 < UB^{(0)}$, the upper bound UB_2 for the subregion $\mathcal{R}_2$ is obtained; there is no need to search for the upper bound in the subregion $\mathcal{R}_1$ since $LB_1 > UB^{(0)}$, i.e. the upper bound in the subregion $\mathcal{R}_1$ is guaranteed to be worse (larger) than $UB^{(0)}$ (Step 5). At this point, the global optimal value f^* is guaranteed to be between $LB^{(1)}$ and $UB^{(1)}$, where $UB^{(1)} = \min(UB^{(0)}, UB_2)$ and $LB^{(1)} = \min(LB_1, LB_2)$ (Step 6). Notice that taking advantage of the reducing subregions, the new relaxation problems provide tighter lower bounds than $LB^{(0)}$ of the previous relaxation problem, i.e. $LB^{(1)}$ is closer to f^* than $LB^{(0)}$. In addition, since $LB_1 > UB^{(1)}$ in this example, the subregion $\mathcal{R}_1$ cannot contain the global optimal solution x^*. Hence, the B&B algorithm will no longer search the subregion $\mathcal{R}_1$ and instead only search the subregion $\mathcal{R}_2$ in the next iterations. This step is called "pruning" (Step 7). The process repeats until the lower and upper bounds, $LB^{(k)}$ and $UB^{(k)}$, are within a specified

tolerance ξ. After the algorithm converges, the upper bound solution found by the B&B algorithm is guaranteed to be within the tolerance from f^*, the globally optimal value.

Figure 2 shows the pseudo code of a typical B&B algorithm [37]. The set for the active feasible subregions is denoted as $\mathcal{S}$, which refers to the set of subregions whose union belongs to $\mathcal{X}$. At every iteration, the B&B algorithm renumbers all the subregions in $\mathcal{S}$ and selects a subregion from $\mathcal{S}$ for branching. Specifically, when the first while-loop starts, $k \coloneqq 1$, $\mathcal{S} = \{\mathcal{X}\}$, and the algorithm selects $\mathcal{X}$ renamed as $\mathcal{R}_1$ for branching. At the k-th iteration, after selecting the subregion for branching, all the subregions in $\mathcal{S}$ are renumbered such that $\mathcal{S} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{n_s}\}$ where the last subregion $\mathcal{R}_{n_s}$ is the selection for branching. The algorithm then partitions $\mathcal{R}_{n_s}$ into n_b number of branched subregions. (While the illustration in Figure 1 shows only two subregions, the "branching" in general can produce $n_b \geq 2$ number of subregions.) Accordingly, $\mathcal{R}_{n_s}$ in the set $\mathcal{S}$ is replaced with the new set of subregions such that

Pseudo code for the branch-and-bound algorithm

$k = 0$;
Let $\mathcal{S} = \{\mathcal{X}\}$ the set of active feasible region(s);
Lower bound: obtain $LB^{(0)}$ by solving the convex relaxation of the original problem;
Upper bound: find $UB^{(0)}$, any feasible solution of the original problem;

while $\left|UB^{(k)} - LB^{(k)}\right| > \xi$ {
    $k \coloneqq k + 1$;
    Node selection: according to a node selection rule, renumber all the subregions in $\mathcal{S}$ such that $\mathcal{S} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{n_s}\}$ where the last subregion $\mathcal{R}_{n_s}$ is the selection for branching;
    Branching: partition $\mathcal{R}_{n_s}$ into n_b number of branched subregions; replace $\mathcal{R}_{n_s}$ in $\mathcal{S}$ with the new set of subregions such that $\mathcal{S} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{n_s+n_b-1}\}$;
    for $i = n_s : n_s + n_b - 1$ {
        Lower bound: Obtain LB_i by solving the convex relaxation of the subproblem in $\mathcal{R}_i$;
        if $LB_i < UB^{(k-1)}$ {
            Upper bound: find UB_i, any feasible solution of the subproblem in $\mathcal{R}_i$;
        }
    }
    Global bounding: $UB^{(k)} \coloneqq$ smallest upper bound among $UB^{(k-1)}$ and UB_i, $i = n_s : n_s + n_b - 1$;
    $LB^{(k)} \coloneqq$ smallest lower bound among LB_i, $i = 1 : n_s + n_b - 1$;
    Pruning: remove $\mathcal{R}_i$ from $\mathcal{S}$ for all i that $LB_i \geq UB^{(k)}$ holds;
}

Figure 2. Pseudo code for the branch-and-bound algorithm

$\mathcal{S} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{n_s+n_b-1}\}$. The B&B algorithm next obtains the lower bound LB_i and upper bound UB_i of the each subproblem corresponding to subregion $\mathcal{R}_i$, $i = n_s, \dots, n_s + n_b - 1$ ("bounding"). The lower bound LB_i is obtained by solving a convex relaxation of the subproblem in $\mathcal{R}_i$ while the upper bound UB_i is obtained by finding a feasible solution of the subproblem in $\mathcal{R}_i$. Hence, at the k-th iteration, the global optimal value f^* is guaranteed to be between $LB^{(k)}$ and $UB^{(k)}$ where $UB^{(k)}$ is the smallest upper bound among $UB^{(k-1)}$ and UB_i, $i = n_s, \dots, n_s + n_b - 1$, and $LB^{(k)}$ is the smallest lower bound among LB_i, $i = 1, \dots, n_s + n_b - 1$. Finally, for all i that $LB_i \geq UB^{(k)}$ holds, "pruning" takes place by removing the corresponding subregion $\mathcal{R}_i$ from the active set $\mathcal{S}$, since such $\mathcal{R}_i$ cannot contain the optimal solution x^*. Over the iterations, as the gap between the upper and lower bounds reduces and approaches ξ, the solution corresponding to $LB^{(k)}$ continues to converge toward the global optimum f^*; final difference between $LB^{(k)}$ and f^* is guaranteed to be no larger than the final gap $\left|UB^{(k)} - LB^{(k)}\right|$.

The key and challenge in the B&B algorithm are to obtain gradually tighter lower bounds of f^* by formulating and solving the relaxed and convex subproblems. Remember that the non-convexity of Eq. (4) arises from the bilinear terms $\mathbf{K}(\boldsymbol{\alpha})\{\boldsymbol{\psi}_i\}$ and $\lambda_i\{\boldsymbol{\psi}_i\}$. For the construction of the convex relaxation of bilinear terms in the B&B algorithm, McCormick envelopes [38] can be utilized. To illustrate the McCormick envelope, consider one example of a scalar bilinear term $x_1 x_2$ of two optimization variables $x_1 \in [L_{x_1}, U_{x_1}]$ and $x_2 \in [L_{x_2}, U_{x_2}]$. Using McCormick envelopes, the bilinear term $x_1 x_2$ is bounded by the four affine functions as:

$$\begin{aligned} x_1 x_2 &\geq U_{x_1} x_2 + x_1 U_{x_2} - U_{x_1} U_{x_2} \\ x_1 x_2 &\geq L_{x_1} x_2 + x_1 L_{x_2} - L_{x_1} L_{x_2} \\ x_1 x_2 &\leq U_{x_1} x_2 + x_1 L_{x_2} - U_{x_1} L_{x_2} \\ x_1 x_2 &\leq x_1 U_{x_2} + L_{x_1} x_2 - L_{x_1} U_{x_2} \end{aligned} \tag{6}$$

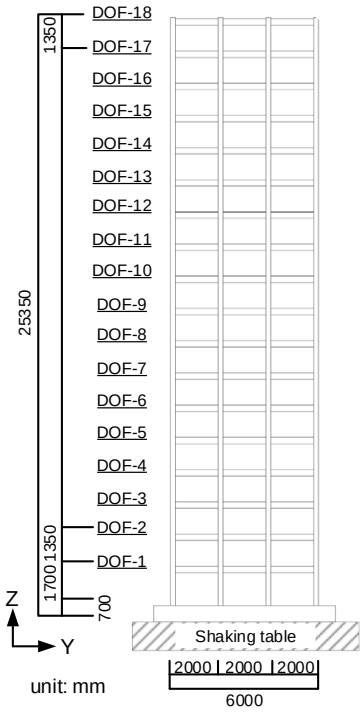
Applying the McCormick envelopes on each bilinear term in Eq. (4), the convex relaxation of the bilinear optimization problem in Eq. (4) can be constructed and the B&B algorithm can be applied.

4 Validation example

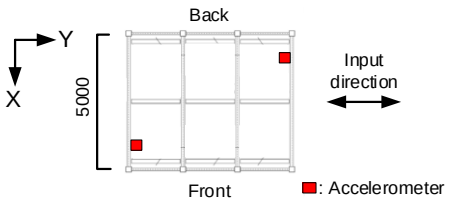
This section presents the model updating results validating the proposed B&B method, in comparison with a few conventionally adopted methods. The validation starts with data generated from the simulation model of an 18-story steel frame structure. Finally, experimental data from shaking table tests is used toward updating the model.

4.1 **18-story steel frame structure**

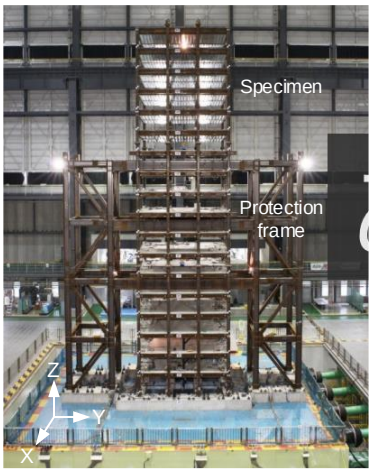
To validate the proposed approach, the model updating of an 18-story steel frame structure is conducted using simulated and experimental data. A one-third scale specimen of the 18-story steel frame was constructed and tested on the shaking table at the E-Defense in Japan [39]. Figure 3 shows the elevation view, floor plan with the location of accelerometers, and front view of the specimen. On every floor, two accelerometers were placed diagonally as shown in Figure 3(b), each measuring horizontal acceleration in both x and y directions. With a steel moment-resisting-frame, the specimen represents the behavior of typical steel high-rise buildings constructed in 1980s to 1990s in Japan. The total height of the specimen is 25.35 m. The dimension of the plan is 6 m by 5 m.



(a) Elevation view



(b) Floor plan and accelerometer location



(c) Front view

Figure 3. 18-story steel frame specimen (unit: mm)

Considering the symmetric rectangular shape of the plan, the effect of eccentricity is neglected in this preliminary study. Therefore, this research adopts an 18-DOF shear model to represent the structural behavior of the specimen. Figure 4 shows the 18-DOF shear model with its properties presented in Table 1. In this paper, the mass information of the specimen is assumed to be accurate. Hence, the inter-story stiffness is to be updated for the

proportionally damped model. The initial inter-story stiffness was obtained from the pushover analysis of a 3D finite element model. While the acceleration responses of all modeled DOFs were measured during testing, instrumentation on every floor is not commonly practical in realistic high-rise applications. Therefore, considering the limitation of sensors in practice, we only consider measurements at the DOF #3, #6, #9, #12, #15, and #18 for model updating, i.e. $n_{\mathcal{M}} = 6$. The downsized measurement scheme makes the situation more realistic and the model updating problem more challenging.

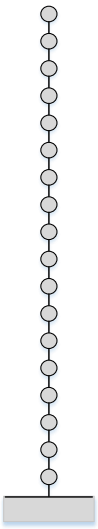


Figure 4. 18-DOF shear model

Table 1. Properties of the 18-DOF shear model

DOF	Weight (kN)	Height (m)	Initial inter-story stiffness (kN/m)	"Actual" stiffness updating variable α_j^{act}	"Actual" stiffness (kN/m)
18*	202	1.35	36,300	0.20	43,560
17	206	1.35	49,100	0.20	58,920
16	206	1.35	56,200	0.10	61,820
15*	206	1.35	61,900	-0.15	52,615
14	206	1.35	66,000	0.05	69,300
13	206	1.35	71,200	-0.15	60,520
12*	206	1.35	78,800	0.25	98,500
11	208	1.35	82,400	0.30	107,120
10	208	1.35	84,000	0.20	100,800
9*	208	1.35	87,600	-0.10	78,840
8	208	1.35	93,800	0.25	117,250
7	208	1.35	96,300	0.15	110,745
6*	208	1.35	99,000	-0.15	84,150
5	208	1.35	102,800	0.10	113,080
4	208	1.35	102,800	-0.10	92,520
3*	208	1.35	107,300	-0.05	101,930
2	208	1.35	109,200	0.05	114,660
1	208	1.70	115,500	0.05	121,275
Note: (* measured/instrumented DOFs)					

4.2 **Numerical simulation**

4.2.1 Problem definition and optimization procedures

To investigate the proposed epsilon-constraint formulation with the B&B algorithm, a simulation study is first conducted. The "actual" structure in this simulation corresponds to the assigned stiffness updating variable α_j^{act} for each modeled DOF in Table 1, which is to be identified through model updating. The "experimental" eigenvalues λ_i^{EXP} and eigenvectors $\boldsymbol{\psi}_i^{\text{EXP}}$ are calculated using the weights and the "actual" stiffness values in Table 1. The number

of "experimental" modes available in this simulation is set as four ($n_{\text{modes}} = 4$). The effect of different norms for the objective function in Eq. (4) is considered using $\mathcal{L}_1$-norm and $\mathcal{L}_2$-norm. In general, $\mathcal{L}_1$-norm is better at avoiding overfit caused by noisy/outlier data, while $\mathcal{L}_2$-norm is more prone to overfit due to the minimization of the difference squares. On the other hand, $\mathcal{L}_1$-norm provides a non-smooth objective function while $\mathcal{L}_2$-norm provides a smooth objective function. Since analytical Jacobians can be derived for the latter, $\mathcal{L}_2$-norm is usually more suitable for gradient-based algorithms. To consider the effect of ε value in Eq. (4), we first define the maximum magnitude in initial stiffness matrix $\mathbf{K}_0$ as $k_{\max} = \max_{i,j} \left| (\mathbf{K}_0)_{i,j} \right|$, which equals 224,700kN/m. The several different ε values considered are $10^{-4}k_{\max}$, $10^{-6}k_{\max}$, and $10^{-8}k_{\max}$. The upper and lower bounds for stiffness updating variable are set as $\boldsymbol{\alpha} \in [-0.3 \cdot \mathbf{1}, 0.3 \cdot \mathbf{1}]$. Therefore, each entry $\alpha_j \in [-0.3, 0.3]$ effectively restricts every stiffness updating parameter to change within ±30% of the initial/nominal value. Bounds of the eigenvalues and eigenvectors are set as $\lambda_i \in \left[-0.8\lambda_i^{\text{EXP}}, 1.2\lambda_i^{\text{EXP}}\right]$ and $\boldsymbol{\psi}_i \in [-2 \cdot \mathbf{1}, 2 \cdot \mathbf{1}]$, respectively. Recall that the largest absolute value of the experimental eigenvector $\boldsymbol{\psi}_i^{\text{EXP},\mathcal{M}}$ is normalized to 1. The weightings are set as $w_{\lambda_i} = 1$ and $w_{\boldsymbol{\psi}_i} = 1$.

To solve the optimization problem in Eq. (4) with the B&B algorithm, a commercial optimization solver BARON [40] is adopted. Other than the basic steps of the B&B algorithm explained in Figure 1, BARON utilizes a range reduction technique that can efficiently narrow the feasible region. The detail of the B&B algorithm in BARON including the range reduction technique, node selection rule, branching rule, and convex relaxation methods can be found in [27, 41]. The B&B algorithm is set to converge when the absolute difference between the upper bound and lower bound of the global optimum (i.e., $UB^{(k)}$ and $LB^{(k)}$ at the k-th iteration) is within 10^{-6}, i.e. $\xi = 10^{-6}$ in Figure 1 . When converged, the solution from BARON is guaranteed to be within the tolerance of the global optimum. As shown below, the $\mathcal{L}_1$-norm formulation of Eq. (4) is rewritten using slack variables $\delta_{\lambda_i} \in \mathbb{R}$ for eigenvalues and $\boldsymbol{\delta}_{\boldsymbol{\psi}_i} \in \mathbb{R}^{n_{\mathcal{M}}-1}$ for eigenvectors. The rewritten problem definition provides an equivalent format that can be readily solved in BARON. Due to their complete mathematical equivalency, for consistency and simplicity in the results presentation, we refer to Eq. (7) as Eq. (4) hereinafter.

$$
\begin{aligned}
&\underset{\boldsymbol{\alpha},\boldsymbol{\lambda},\boldsymbol{\psi},\boldsymbol{\delta}_{\boldsymbol{\lambda}},\boldsymbol{\delta}_{\boldsymbol{\psi}}}{\text{minimize}} \quad \sum_{i=1}^{n_{\text{modes}}} \left(\delta_{\lambda_i} + \sum_{j=1}^{n_{\mathcal{M}}-1} \boldsymbol{\delta}_{\boldsymbol{\psi}_i,j} \right) \\
&\text{subject to} \quad -\delta_{\lambda_i} \le \frac{\lambda_i^{\text{EXP}} - \lambda_i}{\lambda_i^{\text{EXP}}} \cdot w_{\lambda_i} \le \delta_{\lambda_i}, \quad i = 1 \dots n_{\text{modes}} \\
&\qquad -\boldsymbol{\delta}_{\boldsymbol{\psi}_i} \le \left\{ \boldsymbol{\psi}_{-q_i,\,i}^{\text{EXP},\mathcal{M}} - \boldsymbol{\psi}_{-q_i,\,i}^{\mathcal{M}} \right\} \cdot w_{\boldsymbol{\psi}_i} \le \boldsymbol{\delta}_{\boldsymbol{\psi}_i}, \quad i = 1 \dots n_{\text{modes}} \\
&\qquad -\varepsilon \cdot \mathbf{1} \le [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]\{\boldsymbol{\psi}_i\} \le \varepsilon \cdot \mathbf{1}, \quad i = 1 \dots n_{\text{modes}} \\
&\qquad \mathbf{L}_{\boldsymbol{\alpha}} \le \boldsymbol{\alpha} \le \mathbf{U}_{\boldsymbol{\alpha}} \\
&\qquad L_{\lambda_i} \le \lambda_i \le U_{\lambda_i}, \quad i = 1, \dots, n_{\text{modes}} \\
&\qquad \mathbf{L}_{\boldsymbol{\psi}_i} \le \boldsymbol{\psi}_i \le \mathbf{U}_{\boldsymbol{\psi}_i}, \quad i = 1, \dots, n_{\text{modes}}
\end{aligned}
\tag{7}
$$

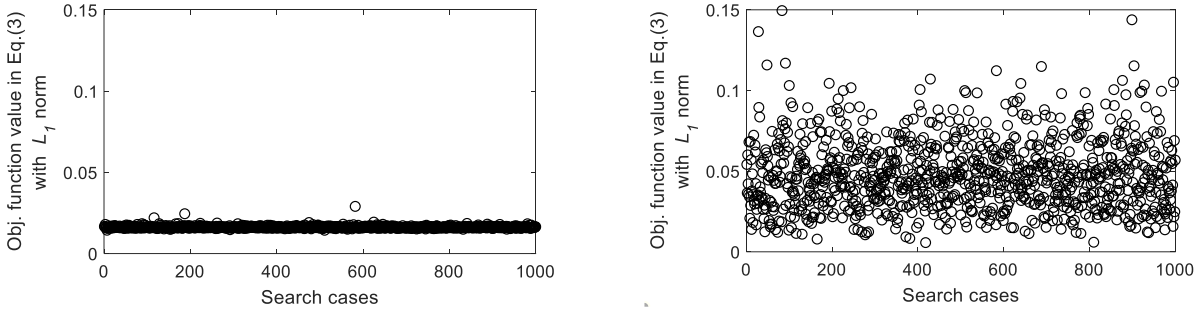
Recall that Eq. (4) is an approximation of Eq. (3). Eq. (3) is the more ideal/accurate modal property difference formulation for model updating, and therefore should be used when possible. Toward performance comparison with the proposed B&B algorithm, we use the ideal Eq. (3) with conventional local optimization algorithms. A number of common conventional algorithms have been implemented in SMU, an-open source MATLAB package for structural model updating developed by authors [42]. SMU supports Eq. (3) with two optimization solvers, `lsqnonlin` and `fmincon`, in MATLAB optimization toolbox [43]. In this research, the interior-point method from `fmincon` is utilized. The interior-point method approximates the original problem using a logarithmic barrier function. The algorithm seeks a local optimum through searching the interior feasible region [44]. In order to increase the chance of finding a better local minimum, randomly generated 1,000 starting points are used to solve the problem in Eq. (3). The analytical Jacobian of the $\mathcal{L}_2$-norm objective function in Eq. (3) has been implemented in SMU and hence used for improving computational efficiency and accuracy. For $\mathcal{L}_1$-norm, the default numerical gradient is utilized.

To compare the B&B algorithm with a stochastic global search algorithm, Eq. (3) is also solved by the genetic algorithm available in MATLAB global optimization toolbox [45]. Genetic algorithm is a derivative-free optimization algorithm. At each step, the algorithm randomly selects individuals from the current population and use them as parents to produce the children for the next generation. Over successive generations, the population may evolve and yield a better solution [46]. Since genetic algorithm also cannot provide any guarantee of global optimality, 1000 random searches are conducted, for consistency with the randomized interior-point optimization in SMU. For both

the interior-point method and the genetic algorithm, variable bounds and weightings in Eq. (3) are set as $\boldsymbol{\alpha} \in [-0.3 \cdot \mathbf{1}, 0.3 \cdot \mathbf{1}]$, $w_{\lambda_i} = 1$, and $w_{\boldsymbol{\psi}_i} = 1$. For all results reported in this paper, we use a Microsoft Windows PC with Intel® Core™ i7-9700 (3.00 GHz) and 8GB RAM memory.

4.2.2 Model updating results in simulation

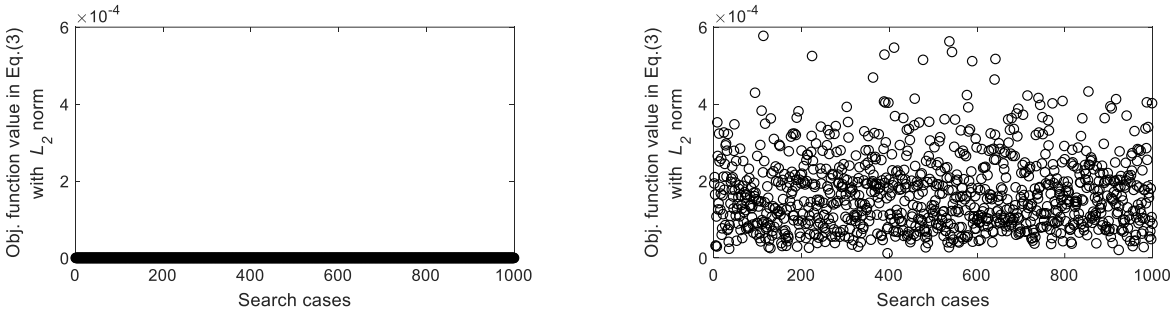
The interior-point method and the genetic algorithm are first applied to solve Eq. (3) using both $\mathcal{L}_1$-norm and $\mathcal{L}_2$-norm, respectively. One thousand uniformly distributed $\boldsymbol{\alpha}$ vectors within the specified bounds are generated as the starting points for the interior-point method. Recalling the model updating problem described in Section 4.2.1, these $\boldsymbol{\alpha}$ vectors correspond to one thousand sets of inter-story stiffness values, each within the ±30% range of the corresponding initial value listed in Table 1. Figure 5(a) shows the objective function values in $\mathcal{L}_1$-norm from the 1,000 random searches of the interior-point method, and Figure 5(b) shows the same results from the genetic algorithm. Overall, searches by the genetic algorithm ended in a wide range of scattered performance, while the interior-point method achieved more uniformly smaller objective function values.



(a) interior-point method (b) genetic algorithm

Figure 5. Objective function values from the 1,000 random searches using Eq. (3) with $\mathcal{L}_1$-norm

Figure 6(a) shows the objective function values in $\mathcal{L}_2$-norm from the 1,000 random searches of the interior-point method, and Figure 6(b) shows the same results from the genetic algorithm. Similar to Figure 5, with $\mathcal{L}_2$-norm the interior-point method generally achieved better objective function values than the genetic algorithm. In addition, the interior-point method converged to solutions in a close cluster, but the genetic algorithm converged at a large variation of solutions with scattered performance.



(a) interior-point method (b) genetic algorithm

Figure 6. Objective function values from the 1,000 random searches using Eq. (3) with $\mathcal{L}_2$-norm

Among 1,000 sets of the optimum parameter set $\boldsymbol{\alpha}^*$ for each norm and each algorithm, the best parameter set is chosen as the one corresponding to the minimum objective function value. To measure the accuracy of the updating results, relative error of stiffness updating variables is calculated as:

$$e_i = \frac{|\alpha_i^* - \alpha_i^{\text{act}}|}{1 + \alpha_i^{\text{act}}} \times 100\ (\%), \qquad i = 1 \cdots n_{\boldsymbol{\alpha}} \tag{8}$$

where α_i^* is the optimal solution of the *i*-th parameter and α_i^{act} is the actual value as listed in Table 1. The average relative error among all stiffness variables is calculated as:

$$e_{\text{avg}} = \frac{1}{n_{\boldsymbol{\alpha}}} \sum_{i=1}^{n_{\boldsymbol{\alpha}}} e_i \tag{9}$$

Table 2 summarizes the model updating results for the three algorithms using simulated data. Compared with the genetic algorithm, the random search with interior-point method achieves better performance both in computational cost and accuracy. The interior-point method especially works better using $\mathcal{L}_2$-norm, owing to the implementation of analytical Jacobian in SMU.

Similar to the interior-point method or the genetic algorithm, the performance using both $\mathcal{L}_1$ and $\mathcal{L}_2$-norm is studied for the B&B algorithm. Using either norm for the B&B algorithm, the effect of different ε values – ranging from $10^{-4}k_{\max}$to $10^{-8}k_{\max}$ – is also investigated. For both norms, the table shows when a smaller ε value is chosen, the algorithm achieves more accurate updating results than when a larger ε value is chosen; this is expected since a larger ε value implies the eigenvalue equations in Eq. (2) are satisfied with larger error. Among all cases considered by the B&B algorithm, ε value equal to $10^{-8}k_{\max}$ using $\mathcal{L}_1$-norm achieves the best accuracy ($e_{\text{avg}} = 0.004\%$). It was also observed that ε values smaller than $10^{-8}k_{\max}$ do not further improve accuracy. Between using $\mathcal{L}_1$ and $\mathcal{L}_2$-

Table 2. Optimization results in simulation with different algorithms, formulations, norms, and ε constraint values

	Problem setup					Optimization results	
Algorithm	Formulation	Norm and ε values			# of searches	Computational time	e_{avg}
Interior-point	Eq. (3)	$\mathcal{L}_1$-norm			1,000	2,040 sec	3.51%
		$\mathcal{L}_2$-norm			1,000	1,634 sec	0.03%
Genetic algorithm	Eq. (3)	$\mathcal{L}_1$-norm			1,000	36,488 sec	3.70%
		$\mathcal{L}_2$-norm			1,000	11,438 sec	5.14%
B&B	Eq. (4)	$\mathcal{L}_1$-norm	ε	$10^{-4}k_{\max}$	1	0.06 sec	3.00%
				$10^{-6}k_{\max}$	1	0.09 sec	0.17%
				$10^{-8}k_{\max}$	1	0.30 sec	0.004%
		$\mathcal{L}_2$-norm	ε	$10^{-4}k_{\max}$	1	5.08 sec	4.30%
				$10^{-6}k_{\max}$	1	2.41sec	3.52%
				$10^{-8}k_{\max}$	1	4.16 sec	2.45%

norm with the B&B algorithm, the objective function in $\mathcal{L}_1$-norm achieves better computational efficiency than $\mathcal{L}_2$-norm by the order of 10 times; $\mathcal{L}_1$-norm also generally provides accurate stiffness updating variables (smaller e_{avg}) compared to $\mathcal{L}_2$-norm.

Overall, compared with the interior-point method and the B&B algorithm, the genetic algorithm requires considerable computational time but provides relatively low accuracy. The interior-point method works better when $\mathcal{L}_2$-norm is used instead of $\mathcal{L}_1$-norm, providing a solution with $e_{\text{avg}} = 0.03\%$. Among all the computational cases in Table 2, the single search by the B&B algorithm using $\mathcal{L}_1$-norm and $\varepsilon = 10^{-8}k_{\max}$ finds the best solution with $e_{\text{avg}} = 0.004\%$. Furthermore, note that the B&B solution is guaranteed to be within a tolerance ($\xi = 10^{-6}$) from the global optimum, while this tolerance certificate is not available by either interior-point method, or genetic algorithm, or other randomized/stochastic search algorithms in general.

Finally, this simulation study shows that using a sufficiently small ε value in the proposed formulation in Eq. (4) leads to accurate updating results that can be as good as the conventional formulation in Eq. (3). To reiterate, when applicable, formulation in Eq. (3) is more ideal/accurate for model updating. However, Eq. (4) – a rewritten explicit form attempting to approximate Eq. (3) – is necessary for adopting the B&B algorithm. Therefore, being less ideal and an approximation of Eq. (3), Eq. (4) poses a less advantageous position for the B&B algorithm. Nevertheless,

based on the simulation results and our general experience that the B&B algorithm implemented in BARON does not work as well with $\mathcal{L}_2$-norm, we will choose $\mathcal{L}_1$-norm and $\varepsilon = 10^{-8}k_{\max}$ for the B&B algorithm in the remainder of this study. In addition, both $\mathcal{L}_1$-norm and $\mathcal{L}_2$-norm are adopted for the interior-point method so that the best between the two norms by interior-point method is readily available for comparison with the B&B algorithm.

4.3 **Experimental validation**

The proposed approach is next verified using the experimental data of the 18-story structure. A uniaxial ground movement was applied along the Y-direction in Figure 3. A long-period long-duration artificial ground motion was used. Figure 7 shows the acceleration time history of the input ground motion, which was scaled so that the maximum pseudo velocity response spectra is equal to 0.4 m/s. Under this excitation, the specimen remained in elastic range that was confirmed by strain gauges installed on the columns and beam-ends [39].

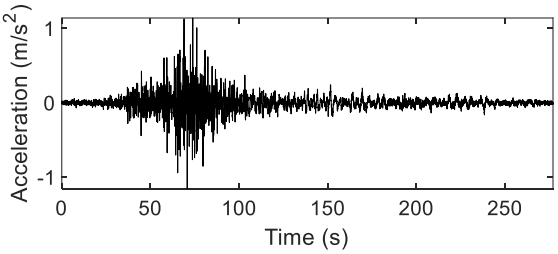


Figure 7. Long-period long-duration ground motion

As same with the numerical study, the number of measured DOF is six ($n_{\mathcal{M}} = 6$) at the DOF #3, #6, #9, #12, #15, and #18. The average of the two acceleration responses along the Y-direction at each floor in Figure 3(b) is taken as the acceleration response for each DOF. A system identification algorithm, Numerical Algorithms for Subspace State Space System Identification (N4SID) [47], was applied on the acceleration responses to extract the modal properties of the specimen. Under the long-period ground excitation, the first two modes are extracted with high confidence from the structural response. Table 3 and Figure 8 summarize the corresponding identified natural frequencies, modal damping ratio, and mode shapes.

Table 3. Modal frequencies and damping ratios obtained by system identification

Mode i	Natural frequency f_i^{EXP}	Damping ratio
1st	0.865 Hz	0.0116
2nd	2.687 Hz	0.0043

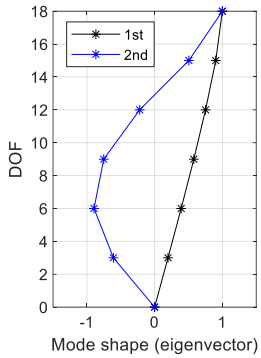


Figure 8. Eigenvectors/mode shapes ($\boldsymbol{\psi}_i^{\text{EXP},\mathcal{M}}$ at measured DOFs only) obtained by system identification

4.3.1 Optimization procedures and conditions

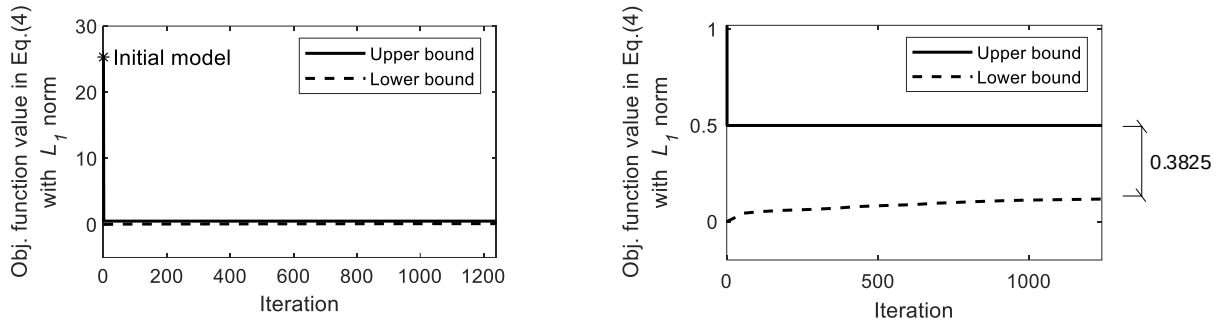
For model updating of the 18-DOF shear model, the B&B algorithm is first applied to solve the epsilon-constraint formulation in Eq. (4) using $\mathcal{L}_1$-norm and $\varepsilon = 10^{-8}k_{\max}$ based on the findings from the simulation study. To improve the computational efficiency of the B&B algorithm, we change the setting of the range reduction technique implemented in BARON. The range reduction technique efficiently reduces the feasible region, which helps obtain tighter lower bounds of the global optimum. The number of variables that BARON applies the range reduction technique is changed from the default automatic selection by BARON to all variables. A single search with the time limitation of 600 seconds is conducted. For comparison with the B&B method, the interior-point method is applied to solve Eq. (3) using $\mathcal{L}_1$-norm and $\mathcal{L}_2$-norm. Randomly generated 1,000 starting points are used for the interior-point method.

For practiced model updating using experimental data, the variable bounds in Eq. (3) and Eq. (4) need to be carefully set to ensure the physical meaning of the updated model. Considering the elastic behavior of the specimen during loading and the relatively low material variance in steel members, the lower and upper bounds of stiffness updating variable $\boldsymbol{\alpha}$ is set as $\boldsymbol{\alpha} \in [-0.3 \cdot \mathbf{1}, 0.3 \cdot \mathbf{1}]$, i.e., allowing 30% difference from the initial inter-story stiffness values in Table 1. For the optimization variables of eigenvalue λ_i and eigenvector $\boldsymbol{\psi}_i$ in Eq. (4), relatively large

variance is allowed, such as $\lambda_i \in \left[-0.8\lambda_i^{\mathrm{EXP}}, 1.2\lambda_i^{\mathrm{EXP}}\right]$ and $\boldsymbol{\psi}_i \in [-2 \cdot \mathbf{1}, 2 \cdot \mathbf{1}]$, to avoid an infeasible problem in Eq. (4). Other parameters that must be carefully chosen are the weightings w_{λ_i} and $w_{\boldsymbol{\psi}_i}$. In general, eigenvalues are identified with much better accuracy compared to eigenvectors. Hence, this research puts more emphasis on eigenvalues by setting $w_{\lambda_i} = 100$ and $w_{\boldsymbol{\psi}_i} = 1$. A more systematic method for assigning weightings can be made considering the Pareto optimal [3] of the eigenvalue and eigenvector difference in Eq. (3) and Eq. (4), but it is out of the scope of this paper. Furthermore, if multiple excitation records and system identification results are available, statistical properties of identified results, such as standard deviation, can be reflected in weightings w_{λ_i} and $w_{\boldsymbol{\psi}_i}$ to take into account uncertainties in measurements and system identification results. Such weighting changes do not affect the non-convexity of the optimization problems, nor do they change the capability of the B&B algorithm in guaranteeing the tolerance certificate when solving the problems.

4.3.2 Model updating results in experiment

The B&B algorithm is first applied to solve Eq. (4) with $\mathcal{L}_1$-norm using the first two modes. Figure 9 shows the convergence history of the upper bound and the lower bound of the global optimum for the time duration of 600 seconds. The initial model provides the objective function value of 25.32. During the preprocessing local searches, the B&B algorithm found a feasible solution with the objective function value equal to 0.4993. This feasible solution provides the upper bound of the global optimum. As the optimization iteration continues, the lower bound of the global optimum has increased from 0 to 0.1168 within the time limit. The gap between the upper and lower bounds of the global optimum ended at 0.3825. However, as mentioned in [48], the B&B algorithm often finds a possible global optimum as an upper bound solution after a small number of iterations, and the remainder of the algorithm is devoted to tightening the lower bound to ensure the global optimality. The upper bound solution found by the B&B algorithm provides a smaller objective function value compared to the initial model and is used as an updating solution in this paper.

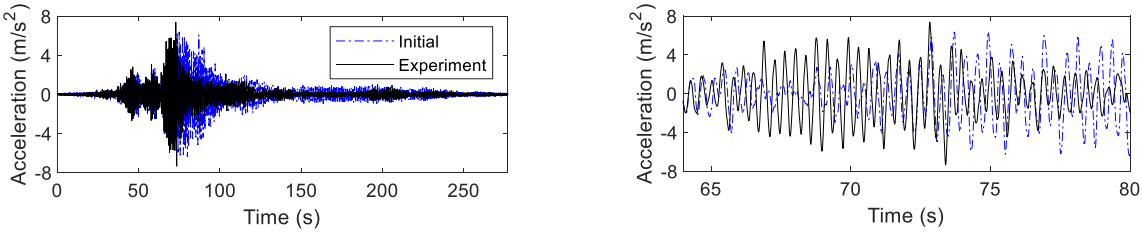


(a) Convergence history with the objective function value of the initial model

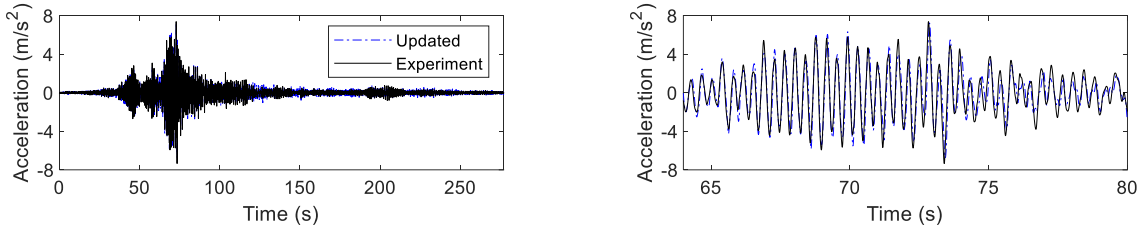
(b) Convergence history (Zoomed in)

Figure 9. Convergence history of the B&B algorithm

The updated model by the B&B algorithm is next used to simulate the acceleration responses subject to the input ground motion. The initial condition of the model is considered as at rest. For simplicity, classical Rayleigh damping is utilized and its coefficients are calculated using the first two damping ratios and frequencies listed in Table 3. Figure 10 plots the acceleration time history responses of the DOF-18 subject to the input ground motion. For consistency, the same time history record is used both for system identification and for the validation of time history responses. Compared to the initial model, the B&B solution successfully provides accurate time history responses in terms of phase and amplitude.



(a) Initial model



(b) Updated model by the B&B algorithm

Figure 10. Acceleration response of the updated model at the DOF-18 by the B&B algorithm

Figure 11 compares the maximum acceleration magnitude at each DOF from experimental measurements, together with the simulated values from the initial model and from the updated model, respectively. Overall, the updated model matches better with experimental data than the initial model. While the B&B solver ended with the tolerance of 0.3825 from the lower bound of the global optimum, the upper bound solution found by the B&B solver successfully provides an acceptable updated model.

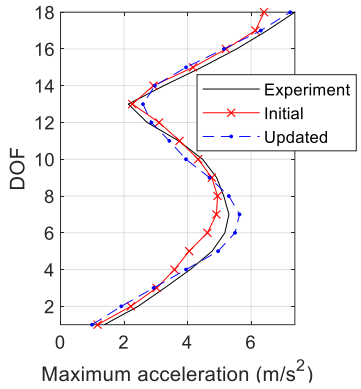


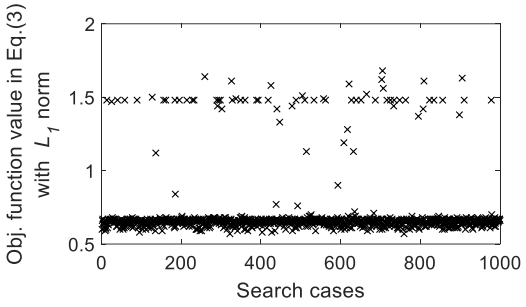
Figure 11. Maximum acceleration magnitude at all modeled DOFs

Next, the interior-point method is applied to solve Eq. (3). Figure 12(a) plots the objective function values of Eq. (3) using $\mathcal{L}_1$-norm and $\mathcal{L}_2$-norm obtained from the 1,000 random searches (because of the different formulations in Eq. (3) and Eq. (4), note that the numerical values in the vertical axis of Figure 12 is not directly comparable with Figure 9). For both norms, several optimization searches end with local optima that have large objective function values. Among the 1000 solutions, the best solution is taken as the one corresponding to the minimum objective function value (0.5676 for $\mathcal{L}_1$-norm and 0.0425 for $\mathcal{L}_2$-norm). In Figure 12(b) for $\mathcal{L}_2$-norm, the solution "A" corresponds to one of the local optima while the "Best" solution corresponds to the minimum objective function equal to 0.0425. In order to visualize the non-convexity of the optimization problem in Eq. (3), a hyperline is defined in $\mathbb{R}^{18}$ space with the objective function in Eq. (3) with $\mathcal{L}_2$-norm denoted by $f(\mathbf{x})$, updating variable $\mathbf{x} \in \mathbb{R}^{18}$, and scalar θ as:

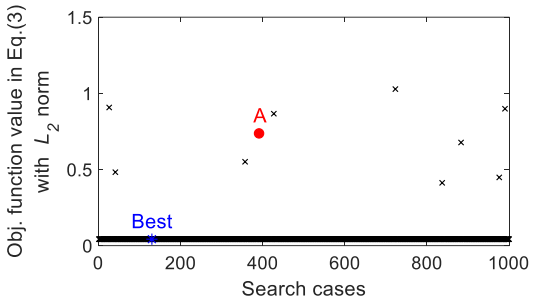
$$g(\theta) = f\big(\mathbf{x}(\theta)\big) = f\big(\mathbf{x}_{\mathrm{A}} + \theta(\mathbf{x}_{\mathrm{Best}} - \mathbf{x}_{\mathrm{A}})\big) \tag{10}$$

where $\mathbf{x}_{\mathrm{A}}$ and $\mathbf{x}_{\mathrm{Best}}$ are the optimization variables at the solution A and the best solution, respectively. If $f(\mathbf{x})\colon \mathbb{R}^{18} \rightarrow \mathbb{R}$ is convex on $\mathbf{x}$, then $g(\theta)\colon \mathbb{R} \rightarrow \mathbb{R}$ should also be convex on θ [49]. Figure 12(c) displays a walk from the solution A ($\theta = 0$) to the best solution ($\theta = 1$). Clearly, there are two valleys along the hyperline near to $\theta = 0$ and $\theta = 1$,

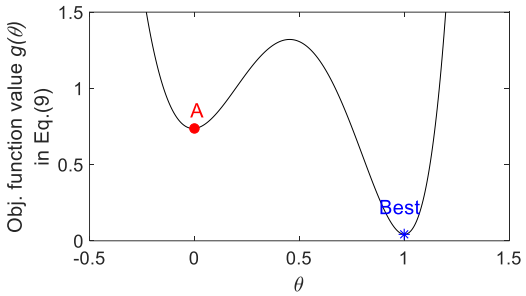
respectively. Therefore, the optimization problem is confirmed to be non-convex, and the solution obtained by local optimization algorithms cannot guarantee global optimality.



(a) Objective function values from the 1,000 random searches using $\mathcal{L}_1$-norm



(b) Objective function values from the 1,000 random searches using $\mathcal{L}_2$-norm



(c) Hyperline walk between the solutions A and the best solution for $\mathcal{L}_2$-norm

Figure 12. Optimization results by the interior-point method

To further investigate the optimality of each solution, the cross validation is next conducted. Table 4 summarizes the objective function values calculated using each solution obtained by the B&B algorithm and the interior-point method. Clearly, the interior-point solution using Eq. (3) with $\mathcal{L}_1$-norm is a local optimum, i.e. the solution obtained by the B&B algorithm achieves smaller objective function in Eq. (3) with $\mathcal{L}_1$-norm (0.4998 by the B&B algorithm vs 0.5676 by the interior-point method). This cross validation again highlights the drawback of local search methods which cannot guarantee the global optimality of the solution.

Table 4. Cross validation of each updated result

	Objective function		
Optimal solution	Eq. (4) with $\mathcal{L}_1$-norm	Eq. (3) with $\mathcal{L}_1$-norm	Eq. (3) with $\mathcal{L}_2$-norm
B&B, Eq. (4) with $\mathcal{L}_1$-norm	**0.4993**	**0.4998**	0.0430
Interior-point, Eq. (3) with $\mathcal{L}_1$-norm	0.5676	0.5676	0.0584
Interior-point, Eq. (3) with $\mathcal{L}_2$-norm	0.5275	0.5275	**0.0425**

Using the updated results, the eigenvalues/natural frequencies and eigenvectors/mode shapes of the updated models are calculated. The modal assurance criterion (MAC) is defined as:

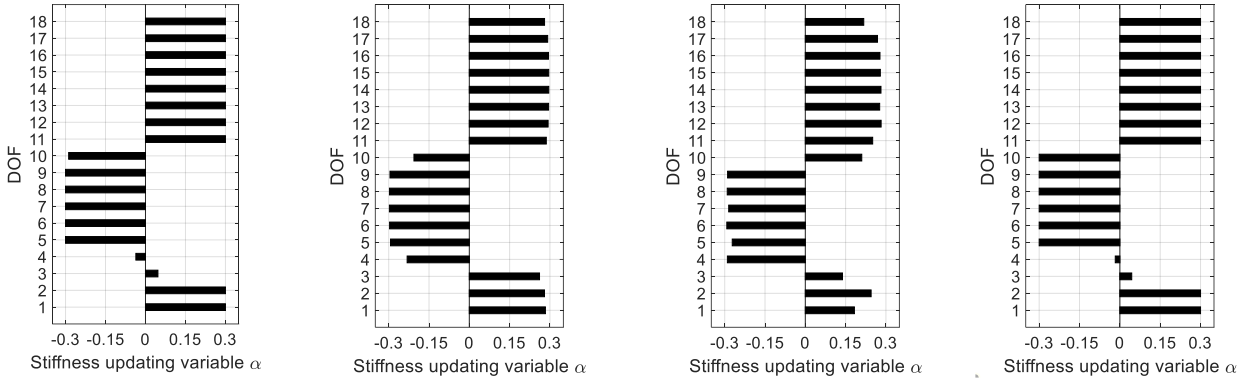
$$\mathrm{MAC}_i = \frac{\left(\left(\boldsymbol{\psi}_i^{\mathrm{EXP},\mathcal{M}}\right)^{\mathrm{T}} \boldsymbol{\psi}_i^{\mathcal{M}}\right)^2}{\left\|\boldsymbol{\psi}_i^{\mathrm{EXP},\mathcal{M}}\right\|_2^2 \left\|\boldsymbol{\psi}_i^{\mathcal{M}}\right\|_2^2}, \quad i = 1 \ldots n_{\mathrm{modes}} \tag{11}$$

The MAC value represents the similarity between two vectors, i.e., the i-th experimental eigenvector $\boldsymbol{\psi}_i^{\mathrm{EXP},\mathcal{M}}$ and i-th simulated eigenvector $\boldsymbol{\psi}_i^{\mathcal{M}}$ at the measured DOFs. The root-mean-square (RMS) error of the acceleration responses of all DOFs between the experimental measurements and updated models is also calculated. Table 5 summarizes the comparison of the updated results. Although the interior-point method shows good computational efficiency in this example, note that the method cannot provide any knowledge about how close the solution is to global optimality. On the other hand, the B&B algorithm provides a certificate that its solution is guaranteed to be no more than 0.3825 away from the global optimum of the objective function value. Compared to the initial model, the solution obtained by the B&B algorithm and the best solutions obtained by the interior-point method achieve more accurate natural frequencies and lower RMS errors. MAC values showed good consistency between the B&B solution and the $\mathcal{L}_2$-norm best solution by the interior-point method. On the other hand, the local optimum solution A by the interior-point method provides a larger RMS error, while the $\mathcal{L}_1$-norm solution and the local solution A by the interior-point method show worse MAC values than the B&B algorithm.

Table 5. Updated results obtained by different algorithms

		Initial model	B&B algorithm	Interior-point method		
Equation			Eq. (4)	Eq. (3)		
Norm			$\mathcal{L}_1$-norm	$\mathcal{L}_1$-norm	$\mathcal{L}_2$-norm	
Solution			Upper bound solution	Best solution	Solution A	Best solution
Computational time		-	600 sec for one search (time-out)	1,071 sec for 1000 searches	505 sec for 1,000 searches	
Natural frequency f_i (relative diff. from f_i^{EXP} in Table 3)	$i = 1$	0.909Hz (5.09%)	0.865Hz (0.00%)	0.865Hz (0.00%)	0.868Hz (0.35%)	0.865Hz (0.00%)
	$i = 2$	2.486Hz (-7.48 %)	2.687Hz (0.00 %)	2.687Hz (0.00 %)	2.680Hz (-0.26 %)	2.687Hz (0.00 %)
MAC value (with $\boldsymbol{\psi}_i^{\mathrm{EXP},\mathcal{M}}$ in Figure 8)	$i = 1$	0.9998	0.9962	0.9957	0.9951	0.9963
	$i = 2$	0.9898	0.9930	0.9896	0.9895	0.9931
RMS error in acc. response		12.30 $\mathrm{m/s^2}$	4.27 $\mathrm{m/s^2}$	4.31 $\mathrm{m/s^2}$	5.86 $\mathrm{m/s^2}$	4.31 $\mathrm{m/s^2}$

Figure 13 shows the stiffness updating variables for each updated result. The stiffness updating variables are nearly consistent among the B&B solution and the $\mathcal{L}_2$-norm best solution by the interior-point method. On the other hand, the $\mathcal{L}_1$-norm solution and the local solution A by the interior-point method provide different values from the other results, especially at the DOF-3, DOF-4, and DOF-10. This observation highlights the importance of finding global optimum in model updating for accurately identifying the updating parameters. In this respect, the B&B algorithm provided reliable model updating results with a tolerance certificate from the global optimum, i.e. the solution is guaranteed to be within a certain tolerance of the global optimum.



(a) Upper bound solution by the B&B algorithm with $\mathcal{L}_1$-norm

(b) Best solution by the interior-point method with $\mathcal{L}_1$-norm

(c) Solution A by the interior-point method with $\mathcal{L}_2$-norm

(d) Best solution by the interior-point method with $\mathcal{L}_2$-norm

Figure 13. Stiffness updating variables obtained by different algorithms

5 Summary and future work

This paper investigates the branch-and-bound (B&B) algorithm toward solving the non-convex optimization problem in finite element model updating. Using the epsilon-constraint technique, Eq. (4) first proposes a reformulation of the conventional modal property difference formulation in Eq. (3). The reformed formulation is explicit in terms of optimization variables, which enables the application of the B&B algorithm. A simulation study is first conducted using the 18-DOF shear model. The simulation study shows that a small ε value in the proposed epsilon-constrain formulation can provide accurate updating results.

The proposed model updating approach is next validated by the shaking table test of an 18-story steel frame structure and compared with a randomized gradient search algorithm. Using the first two modes in model updating, the upper bound solution found by the B&B algorithm provides the reliable updated model in terms of the natural

frequencies, mode shapes, and time history responses. The performance is similar to the updated model by the randomized gradient search algorithm. Since such randomized local optimization algorithms may or may not yield a good solution even with thousands of searches, there is a need for algorithms such as the B&B algorithm that can guarantee the tolerance of a solution from the global minimum.

Although this study validates the feasibility of the B&B algorithm for solving non-convex optimization problems in finite element model updating, future research is needed to further reduce the tolerance/gap between the lower and upper bounds. When using experimental data, the B&B algorithm showed slower convergence; the computational efficiency could be improved in future work. In particular, better efficiency may be achieved by implementing binary expansion techniques [50], which can transform the bilinear optimization problem in Eq. (4) into mixed integer linear programming. A novel relaxation and branching scheme could also lead to a better computational efficiency [34].

Finally, while this study focuses on deterministic model updating, future research may apply the B&B algorithm on probabilistic model updating problems that allow uncertainty quantification. Since commonly adopted optimization formulations usually use implicit functions of the optimization variables, the first challenge would be recasting them into explicit forms (by a process similar to e.g. Eq. (4) or [13]). Such explicit forms with additional variables are expected to increase computational difficulty for the B&B algorithm, which is another challenge that is yet to be overcome in the future.

Acknowledgements

This research was partially funded by the National Science Foundation (CMMI-1634483). The first author received scholarship support from the Nakajima Foundation. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the view of the sponsors.

Reference

[1] M. I. Friswell and J. E. Mottershead, *Finite element model updating in structural dynamics*. Dordrecht; Boston: Kluwer Academic Publishers, 1995.

[2] P. W. Moller and O. Friberg, "Updating large finite element models in structural dynamics," *AIAA Journal*, vol. 36, pp. 1861-1868, 1998.

[3] K. Christodoulou and C. Papadimitriou, "Structural identification based on optimally weighted modal residuals," *Mechanical Systems and Signal Processing*, vol. 21, pp. 4-23, 2007.
[4] O. S. Salawu, "Detection of structural damage through changes in frequency: A review," *Engineering Structures*, vol. 19, pp. 718-723, 1997.
[5] M. Kurata, J.-H. Kim, J. P. Lynch, K. H. Law, and L. W. Salvino, "A probabilistic model updating algorithm for fatigue damage detection in aluminum hull structures," in *Smart Materials, Adaptive Structures and Intelligent Systems*, 2010, pp. 741-750.
[6] K. V. Yuen, "Updating large models for mechanical systems using incomplete modal measurement," *Mechanical Systems and Signal Processing*, vol. 28, pp. 297-308, 2012.
[7] X. Dong and Y. Wang, "Modal property difference formulations and optimization algorithm comparison towards FE model updating," *Proceedings of SPIE 2018, Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring* Denver, CO, USA, 2018.
[8] B. Jaishi and W. X. Ren, "Damage detection by finite element model updating using modal flexibility residual," *Journal of Sound and Vibration*, vol. 290, pp. 369-387, 2006.
[9] D. Zhu, X. Dong, and Y. Wang, "Substructure stiffness and mass updating through minimization of modal dynamic residuals," *ASCE Journal of Engineering Mechanics*, vol. 142, p. 04016013, 2016.
[10] C. Farhat and F. M. Hemez, "Updating finite element dynamic models using an element-by-element sensitivity methodology," *AIAA Journal*, vol. 31, pp. 1702-1711, 1993.
[11] J. B. Kosmatka and J. M. Ricles, "Damage detection in structures by modal vibration characterization," *ASCE Journal of Structural Engineering*, vol. 125, pp. 1384-1392, 1999.
[12] Y. Reuland, P. Lestuzzi, and I. F. C. Smith, "Data-interpretation methodologies for non-linear earthquake response predictions of damaged structures," *Frontiers in Built Environment*, vol. 3, 2017.
[13] K.-V. Yuen and S.-C. Kuok, "Bayesian methods for updating dynamic models," *Applied Mechanics Reviews*, vol. 64, 2011.
[14] R. Astroza, A. Alessandri, and J. P. Conte, "A dual adaptive filtering approach for nonlinear finite element model updating accounting for modeling uncertainty," *Mechanical Systems and Signal Processing*, vol. 115, pp. 782-800, 2019.
[15] I. Behmanesh and B. Moaveni, "Accounting for environmental variability, modeling errors, and parameter estimation uncertainties in structural identification," *Journal of Sound and Vibration*, vol. 374, pp. 92-110, 2016.
[16] Y. Wang, X. Dong, and D. Li, "A non-convexity study in finite element model updating," *Proceedings of the 9th International Conference on Structural Health Monitoring of Intelligent Infrastructure (SHMII-9)*, St. Louis, MO, USA, 2019.
[17] J. Jang and A. W. Smyth, "Model updating of a full-scale FE model with nonlinear constraint equations and sensitivity-based cluster analysis for updating parameters," *Mechanical Systems and Signal Processing*, vol. 83, pp. 337-355, 2017.
[18] A. Teughels, G. De Roeck, and J. A. Suykens, "Global optimization by coupled local minimizers and its application to FE model updating," *Computers & structures*, vol. 81, pp. 2337-2351, 2003.
[19] P. G. Bakir, E. Reynders, and G. De Roeck, "An improved finite element model updating method by the global optimization technique 'Coupled Local Minimizers'," *Computers & Structures*, vol. 86, pp. 1339-1352, 2008.
[20] R. Levin and N. Lieven, "Dynamic finite element model updating using simulated annealing and genetic algorithms," *Mechanical systems and signal processing*, vol. 12, pp. 91-120, 1998.
[21] N. F. Alkayem, M. Cao, Y. Zhang, M. Bayat, and Z. Su, "Structural damage detection using finite element model updating with evolutionary algorithms: a survey," *Neural Computing and Applications*, pp. 1-23, 2017.
[22] F. Kang, J. Li, and Q. Xu, "Structural inverse analysis by hybrid simplex artificial bee colony algorithms," *Computers & Structures*, vol. 87, pp. 861-870, 2009.
[23] D. Li, X. Dong, and Y. Wang, "Model updating using sum of squares (SOS) optimization to minimize modal dynamic residuals," *Structural Control and Health Monitoring*, vol. 25, p. e2263, 2018.
[24] D. Li and Y. Wang, "Sparse sum-of-squares optimization for model updating through minimization of modal dynamic residuals," *Journal of Nondestructive Evaluation, Diagnostics and Prognostics of Engineering Systems*, vol. 2, 2019.
[25] D. Li and Y. Wang, "Modal dynamic residual-based model updating through regularized semidefinite programming with facial reduction," *Mechanical Systems and Signal Processing*, vol. 143, p. 106792, 2020.
[26] J. E. Falk and R. M. Soland, "An algorithm for separable nonconvex programming problems," *Management science*, vol. 15, pp. 550-569, 1969.

[27] M. Tawarmalani, N. V. Sahinidis, and N. Sahinidis, *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications* vol. 65. Berlin, Germany: Springer Science & Business Media, 2002.
[28] E. L. Lawler and D. E. Wood, "Branch-and-bound methods: A survey," *Operations research,* vol. 14, pp. 699-719, 1966.
[29] H. S. Ryoo and N. V. Sahinidis, "Global optimization of nonconvex NLPs and MINLPs with applications in process design," *Computers & Chemical Engineering,* vol. 19, pp. 551-566, 1995.
[30] J. Binney and G. S. Sukhatme, "Branch and bound for informative path planning," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 2147-2154.
[31] D. J. Chmielewski, T. Palmer, and V. Manousiouthakis, "On the theory of optimal sensor placement," *AIChE journal,* vol. 48, pp. 1001-1012, 2002.
[32] V. Balakrishnan, S. Boyd, and S. Balemi, "Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems," *International Journal of Robust and Nonlinear Control,* vol. 1, pp. 295-317, 1991.
[33] P. Thanedar and G. Vanderplaats, "Survey of discrete variable optimization for structural design," *Journal of Structural Engineering,* vol. 121, pp. 301-306, 1995.
[34] S. S. Dey, A. Santana, and Y. Wang, "New SOCP relaxation and branching rule for bipartite bilinear programs," *Optimization and Engineering,* vol. 20, pp. 307-336, 2019.
[35] T. Janter and P. Sas, "Uniqueness aspects of model-updating procedures," *AIAA Journal,* vol. 28, pp. 538-543, 1990.
[36] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization,* vol. 26, pp. 369-395, 2004.
[37] H. S. Ryoo and N. V. Sahinidis, "A branch-and-reduce approach to global optimization," *Journal of global optimization,* vol. 8, pp. 107-138, 1996.
[38] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems," *Mathematical programming,* vol. 10, pp. 147-175, 1976.
[39] K. Suita, Y. Suzuki, and M. Takahashi, "Collapse behavior of an 18-story steel moment frame during a shaking table test," *International Journal of High-Rise Buildings,* vol. 4, pp. 171-180, 2015.
[40] N. V. Sahinidis, "BARON: A general purpose global optimization software package," *Journal of global optimization,* vol. 8, pp. 201-205, 1996.
[41] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical Programming,* vol. 103, pp. 225-249, 2005.
[42] Y. Wang, X. Dong, D. Li, and Y. Otsuki, SMU: MATLAB Package for Structural Model Updating, version 1.1, https://github.com/ywang-structures/Structural-Model-Updating, January 2019.
[43] MathWorks Inc., *Optimization Toolbox™ User's Guide*, Version 8.4. ed. Natick, MA: MathWorks Inc., 2019.
[44] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica,* vol. 4, pp. 373-395, 1984.
[45] MathWorks Inc., *Global Optimization Toolbox™ User's Guide*, Version 4.2. ed. Natick, MA: MathWorks Inc., 2019.
[46] D. Whitley, "A genetic algorithm tutorial," *Statistics and computing,* vol. 4, pp. 65-85, 1994.
[47] P. Van Overschee and B. De Moor, "N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems," *Automatica,* vol. 30, pp. 75-93, 1994.
[48] K.-C. Goh, M. G. Safonov, and G. P. Papavassilopoulos, "Global optimization for the biaffine matrix inequality problem," *Journal of Global Optimization,* vol. 7, pp. 365-380, 1995.
[49] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK ; New York: Cambridge University Press, 2004.
[50] A. Gupte, S. Ahmed, M. S. Cheon, and S. Dey, "Solving mixed integer bilinear problems using MILP formulations," *SIAM Journal on Optimization,* vol. 23, pp. 721-744, 2013.