Development of Hierarchical Optimization-based Models for Multiscale Damage Detection

Hao Sun

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2014

ABSTRACT

Development of Hierarchical Optimization-based Models for Multiscale Damage Detection

Hao Sun

In recent years, health monitoring of structure and infrastructure systems has become a valuable source of information for evaluating structural integrity, durability and reliability throughout the lifecycle of structures as well as ensuring optimal maintenance planning and operation. Important advances in sensor and computer technologies made possible to process a large amount of data, to extract the characteristic features of the signals, and to link those to the current structural conditions. In general, the process of data feature extraction relates to solving an inverse problem, in either a data-driven or a model-based type setting.

This dissertation explores state-of-the-art hierarchical optimization-based computational algorithms for solving multiscale model-based inverse problems such as system identification and damage detection. The basic idea is to apply optimization tools to quantify an established model or system, characterized by a set of unknown governing parameters, via minimizing the discrepancy between the predicted system response and the measured data. We herein propose hierarchical optimization algorithms such as the improved artificial bee colony algorithms integrated with local search operators to accomplish this task.

In this dissertation, developments in multiscale damage detection are presented in two parts. In the first part, efficient hybrid bee algorithms in both serial and parallel schemes are proposed for time domain input-output and output-only identification of macro-scale linear/nonlinear systems such as buildings and bridges. Solution updating strategies of the artificial bee colony algorithm are improved for faster convergence, meanwhile, the simplex method and gradient-based optimization techniques are employed as local search operators for accurate solution tuning. In the case of output-only measurements, both system parameters and the time history of input excitations can be simultaneously identified using a modified Newmark integration scheme. The synergy between the proposed method and

Bayesian inference are proposed to quantify uncertainties of a system. Numerical and experimental applications are investigated and presented for macro-scale system identification, finite element model updating and damage detection.

In the second part, a framework combining the eXtended Finite Element Method (XFEM) and the proposed optimization algorithms is investigated, for nondestructive detection of multiple flaws/defects embedded in meso-scale systems such as critical structural components like plates. The measurements are either static strains or displacements. The number of flaws as well as their locations and sizes can be identified. XFEM with circular and/or elliptical void enrichments is employed to solve the forward problem and alleviates the costly re-meshing along with the update of flaw boundaries in the identification process. Numerical investigations are presented to validate the proposed method in application to detection of multiple flaws and damage regions.

Overall, the proposed multiscale methodologies show a great potential in assessing the structural integrity of building and bridge systems, critical structural components, etc., leading to a “smart structure and infrastructure management system”.

Contents

List of Figures

List of Tables

Acknowledgments

Most of all, I would like to express my deepest thanks to my advisor Professor Raimondo Betti for his continuous support, numerous encouragement and great guidance whenever I needed it during my doctoral study at Columbia. Especially, I have to thank him for his patience with me and trust in me all the time. I would say Professor Betti has been an excellent mentor. He gave me enough freedom to pursue my own ideas towards research and shared his extensive knowledge in the field of structural health monitoring with me. This work cannot be done without his help and support. It's my honor and fortune as well as pleasure to be his student.

I am also grateful to Professor Haim Waisman, who introduced me to a new research area in computational mechanics for inverse problems. I am extremely fortunate to work with him and learn a lot from him. His insightful suggestions and kind guidance gave me a lot of inspirations. Thanks go to Professors George Deodatis and Maria Q. Feng for writing reference letters to support my job application and for the advice they supplied to me.

I would gratefully acknowledge my dissertation defense committee (Professors Raimondo Betti, Haim Waisman, George Deodatis, Maria Q. Feng and Alan C. West) for their valuable suggestions, remarks and comments in the completion of my dissertation.

My appreciation also goes to the Department of Civil Engineering and Engineering Mechanics at Columbia University, for providing me with full financial support during my study at Columbia. I would like to thank Professors Rene Testa, Andrew Smyth and Ibrahim Odeh for their suggestions and help when I served as a Teaching Assistant for their classes.

Thanks also go to my friends at Columbia – Zifeng Yuan, Nan Hu, Dongming Feng, Jianqiang Wei, Muqing Liu, Yang Jiao, Yongxiang Wang, Weizhi Shao, Chih-Shiuan Lin,

Gan Song, Yingjie Liu, Jinwoo Jang, Suparno Mukhopadhyay, Luciana Balsamo, Matthew Sloane, and others, who made my PhD life joyful and memorable.

I am deeply thankful to my best friends, Kechuang Ni, Jingwei Zheng, Jianwei Bai and Tuanwei Shi, who are like blood brothers to me. They have consistently encouraged me in the journey of realizing my dream.

Finally, my special acknowledgement is reserved for my paraents, Maotong Sun and Diancai Hu, and my two sisters, Jing Sun and Ying Sun, who love me all along and provide me with the motivation to purse a PhD. Without their support, I cannot accomplish this work. Lastly but most importantly, I am extremely thankful to my fiancée Yang Liu, who is also pursuing her PhD at Columbia and accompanied with me in the past four years. She has been a constant source of spirit support and encouragement for me. Without her strongest support, I cannot carry out the difficult task of PhD studies.

To my parents, my two sisters and my fiancée Yang
who love, encourage and support me all the time.

Chapter 1

Introduction

In recent years, health monitoring of structure and infrastructure systems has become a valuable source of information for evaluating structural integrity, durability and reliability throughout the lifecycle of structures as well as ensuring optimal maintenance planning and operation. Important advances in sensor and computer technologies made possible to process a large amount of data, to extract the characteristic features of the signals, and to link those to the current structural conditions.

With the primary goal to establish a Structural Health Monitoring (SHM) framework for assessing serviceability of a structure, detecting damages, evaluating its remaining service life, etc., the developments of SHM technologies have drawn considerable interests and efforts of researchers.

In general, we divide SHM into three stages: (i) sensing and data acquisition, (ii) data analysis, signal feature extraction and identification, model validation and updating, and (iii) maintenance, repair and rehabilitation operation. From the scale point of view, SHM can be divided into two categories: (i) monitoring a large scale (macro-scale) structural system from a global perspective (see Figure 1.1), and (ii) Nondestructive Testing/Evaluation (NDT/E) of critical structural components at a meso-scale from a local perspective (see Figure 1.2). Instrumentation, sensing, modeling and probably identification techniques might be different considering different structural length scales. Therefore, this work aims to propose methodologies to solve the inverse problems (e.g., system identification, damage detection, etc.) in SHM/NDE across multiple structural length scales.

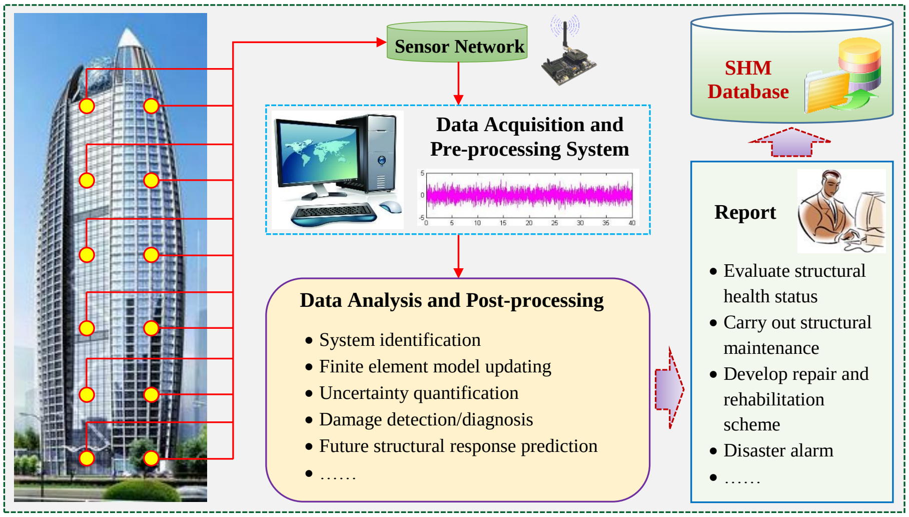

Sensor Network
Data Acquisition and Pre-processing System
SHM Database
Report
Data Analysis and Post-processing
• System identification
• Finite element model updating
• Uncertainty quantification
• Damage detection/diagnosis
• Future structural response prediction
• ……
• Evaluate structural health status
• Carry out structural maintenance
• Develop repair and rehabilitation scheme
• Disaster alarm
• ……

Figure 1.1: Schemetic representation – the process of SHM.

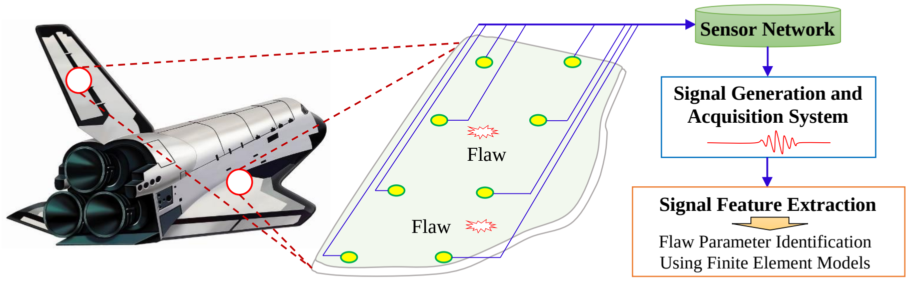

Sensor Network
Signal Generation and Acquisition System
Signal Feature Extraction
Flaw Parameter Identification Using Finite Element Models
Flaw
Flaw

Figure 1.2: Schemetic representation – NDT/E of critical structural components.

1.1 Dissertation outline

This dissertation explores state-of-the-art hierarchical optimization-based computational algorithms for solving multiscale model-based inverse problems such as system identification and damage detection. The basic idea is to apply optimization tools to quantify an established model or system, characterized by a set of unknown governing parameters, via minimizing the discrepancy between the predicted system response and the measured data. We herein propose hierarchical optimization algorithms such as the improved artificial bee colony algorithms integrated with local search operators to accomplish this task.

Developments in multiscale damage detection are presented in two parts, e.g., macro- and meso-scale analyses. The following summary provides a brief outline of this dissertation:

- **Chapter 2**: A modified artificial bee colony (MABC) algorithm is presented in the context of structural system identification. This algorithm is a population based stochastic algorithm which requires only few common control parameters. It has a simple structure, it is easy to implement, it is robust, and it provides results with high accuracy. The application of a proposed nonlinear factor for convergence control enhances the balance of global and local searches. The idea behind the nonlinear factor is to free the method, at its initial steps, to search over the entire search space and to allow for faster convergence during the local search phase which takes place towards the end of the iteration process. In this way, the search space is gradually reduced following a nonlinear trend. To avoid stagnation around local minima, a "scout bee" search will be carried out if needed. To investigate the applicability of this proposed technique to structural system identification, two linear systems and a nonlinear system were studied under different conditions, addressing issues such as the number of measurements used in identification, noise in the signals, and the knowledge of the structural mass. In all cases considered, the simulation results show that the proposed MABC algorithm can produce an excellent parameter estimation with small errors. The presented method is effective, robust and efficient even with reduced partial measurements at high noise pollution levels.

- **Chapter 3**: A hybrid heuristic optimization strategy is presented to simultaneously identify structural parameters and, when possible, dynamic input time histories from incomplete sets of output measurements. The proposed strategy combines MABC with a local search operator, Nelder-Mead Simplex Method (NMSM), integrated in a Search Space Reduction (SSR) approach, so to improve the convergence efficiency of the overall identification process. Because of the independent nature of the algorithm, a parallel scheme is implemented to improve the computational efficiency. If the time histories of the structural response and information about the structural mass are available, then the algorithm can also be used for identification of the dynamic input force time histories through a modified Newmark integration scheme, using the current estimates of the structural parameters. To investigate the applicability of the proposed technique, three numerical examples, two shear-type building models and a coupled building system model, under different conditions of data availabilities and noise corruption levels, were tested. The results show that the proposed technique is powerful in the simultaneous identification of the structural parameters and input forces even from an incomplete set of noise contaminated measurements.

- **Chapter 4**: An optimization-based Bayesian inference methodology is presented for the probabilistic finite element model updating of structural systems. The model updating process is first formulated as an inverse problem analyzed by Bayesian inference and is solved using a hybrid optimization algorithm. The proposed hybrid approach is a synergy of a new version of MABC and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. The present MABC version includes four modifications (in the solution phases of initialization, updating, selection and rebirth) compared with the standard bee algorithm, which basically improve the global search performance. BFGS is frequently inserted to improve the algorithm's finer solution search ability aiming at a higher solution accuracy. In brief, a probabilistic framework based on Bayesian inference is first derived so to get the regularized objective function for optimization. Then the proposed MABC-BFGS algorithm is applied to determine the uncertain system parameters by minimizing the objective function. System parameters as well as the prediction error covariance are updated iteratively in the optimization process. Finally, the effectiveness of the proposed approach was illustrated by the numerical data sets of the Phase I IASC-ASCE benchmark model and the experimental data sets of a three-storey frame structure (from the Los Alamos National Laboratory (LANL), NM, USA).

- **Chapter 5**: This chapter presents a novel algorithm based on the eXtended Finite Element Method (XFEM) and a hybrid artificial bee colony (HABC) algorithm to detect and quantify multiple flaws in structures. The concept is based on recent work that has shown the excellent synergy between XFEM, used to model the forward problem, and a Genetic-type Algorithm (GA) to solve an inverse identification problem and converge to the "best" flaw parameters [1–4]. In this work, an adaptive algorithm that can detect multiple flaws without any knowledge on the number of flaws beforehand is proposed. The algorithm is based on the introduction of topological variables into the search space, used to adaptively activate/deactivate flaws during run time until convergence is reached. The identification is carried out using a limited number of strain sensors assumed to be attached to the structure surface boundaries. Each flaw is approximated by a circular void with three variables: center coordinates and radius, within the XFEM framework. In addition, the proposed HABC scheme combines the guided-to-best solution updating strategy-based MABC with a local search operator of NMSM, which shows faster convergence and superior global/local search abilities compared to MABC or classic GA algorithms. Several numerical examples, with increasing level of difficulty, were studied in order to evaluate the proposed algorithm. In particular we considered identification of multiple flaws with unknown a priori information on the number of flaws (which makes the inverse problem harder), the proximity of flaws, flaws having irregular shapes (similar to artificial noise) and the effect of structured/unstructured meshes. The results show that the proposed XFEM-HABC algorithm is able to converge in all test problems and accurately identify flaws. Hence this methodology is found to be robust and efficient for nondestructive detection and quantification of multiple flaws in structures.

- **Chapter 6**: This chapter presents a novel multiscale algorithm for nondestructive detection of multiple flaws in structures, within an inverse problem type setting, which further improves the previous work on XFEM−HABC algorithm for multi-void flaws detection (see Chapter 5). The key idea is to apply a two-step optimization scheme, where first rough flaw locations are quickly determined and then fine tuning is applied in these localized subdomains to obtain global convergence to the true flaws. The two step framework combines the strengths of heuristic and gradient based optimization methods. The first phase employs a discrete type optimization in which the optimizer is limited to specific flaw locations and shapes, thus converting a continuous optimization problem in the entire domain into a coarse discrete optimization problem with a limited number of choices. To this end we develop a special algorithm called discrete artificial bee colony (DABC). The second phase employs a BFGS type approach on local well defined and bounded subdomains. A semi-analytical approach is developed to compute the stiffness derivative associated with the evaluation of objective function gradients. The XFEM with both circular and elliptical void enrichment functions, is used to solve the forward problem and alleviate the costly re-meshing of every candidate flaw, in both optimization steps. The multiscale algorithm was tested on several benchmark examples to identify various numbers and types of flaws with arbitrary shapes and sizes (e.g., cracks, voids, and their combination), without knowing the number of flaws beforehand. We studied the size effect of the pseudo grids in the first optimization step and consider the effect of modeling error and measurement noise. The results were compared with the previous work that employed a single continuous optimization scheme (e.g., XFEM-GA and XFEM-HABC methods). We illustrate that the proposed methodology is robust, yields accurate flaw detection results and in particular leads to significant improvements in convergence rates compared with the previous work.

Part I

Macro-scale (coarse-scale) analysis

Chapter 2

A Modified ABC algorithm for Input-Output System Identification

In this chapter, the ABC algorithm and its modified version are presented and validated for input-output system identification. This chapter is reproduced from the paper co-authored with Professors Hilmi Luş and Raimondo Betti and published in the Journal of Computers & Structures [5].

2.1 Introduction

Identification of the dynamic characteristics and structural parameters of models representing complex structural systems plays a key role in SHM for model updating, damage detection, active control and NDE, etc. The System Identification (SI) process, formulated as an inverse problem, aims to determine a set of parameters, either physical or non-physical, of a model that is representative of the structure in question. Physical parameters might be considered as the mass, damping and stiffness of the structural elements while the coefficients of an autoregressive model can be labeled as non-physical parameters. These estimated parameters can then be used, among other quantities, to predict the structural response to a future excitation or to assess the structural conditions. In essence, SI can be thus considered as an optimization process in which the objective is to identify a model of a system so that its predicted response to a given input is close enough to the measured response from the real system. In recent years, considerable efforts have been done in developing reliable SI models of structural systems using time histories of the structural response and/or the input, for example, as shown in References [6–12].

There are two categories of approaches for SI, depending on whether the analysis is conducted in the frequency domain, Frequency Domain Methods (FDMs), or in the time domain, Time Domain Methods (TDMs). FDMs involve the identification of modal quantities such as natural frequencies, mode shapes and damping ratios, using frequency domain tools like the transfer function or the power spectral density function [13–15], and have been widely used in structural modal analysis. However, dealing with modal characteristics is not always advantageous when one is interested in damage assessment. In fact, structural frequencies often vary more greatly because of environmental effects, such as temperature changes, than because of structural damage, according to experiments carried by Farrar and Doebling [16]. Another problem is linked to the limited sensitivity of modal characteristics to structural damage: if a change/damage occurs in the stiffnesses of just a few elements, this might lead to a very small change in the structural frequencies [17] and might not be easily detected using FDMs. Moreover, mode shapes tend to be very difficult to be accurately "measured" due to their susceptibility to noise, especially for the higher modes.

By contrast, TDMs aim to estimate the system's physical parameters using directly the time histories of the observed structural response and trying to create a mathematical (physical or non-physical) model that simulates such response as closely as possible. Some of such methodologies can even track system parameters change during the duration of the records. Among the TDMs, it is noteworthy to list the identification approaches based on Least-Square methods [10, 11, 18, 19], the Kalman filters [7–9, 20, 21], the particle filter [22], the H_∞ filter [23], and the sequential Monte Carlo method [24]. These methods have proven to be successful in application to structural parametric identification.

However, most of the above mentioned methods require a good initial guess of the parameters and a proper function gradient. In addition, difficulties arise using these methods in the identification of large systems when few measurement data is available. Thanks to the advances in computer technology over the last decades, with a tremendous increase in computational speed and memory size, methodologies based on heuristic algorithms have become more popular to solve the optimization problem in SI in the time domain: in particular, genetic algorithms (GAs) [17,25–27], particle swarm optimization (PSO) [28,29], artificial neural networks (ANN) [30], evolutionary strategy (ES) [31], and differential evolution (DE) [32] have gained great attention and recognition in the field of SI. These algorithms, even though facing some of the same issues as other TDMs (e.g., solution uniqueness in the case of an incomplete set of measurements), are conceptually simple, following laws taken from nature, and rely on performing a large number of iterations (forward analyses) to identify an "optimal" solution in a search space.

Apart from the above mentioned heuristic algorithms, a novel swarm intelligence algorithm called the artificial bee colony (ABC) algorithm was introduced by Karaboga [33] in 2005 for solving complex numerical optimization problems. This algorithm was motivated by the intelligent behavior of honey bees when seeking a high quality food source. The ABC algorithm is a population based stochastic approach with implementation simplicity since only three common control parameters are used: the colony size, the maximum number of iterations and the limit number of iterations. It has the advantage of simple structure (as simple as PSO and DE), ease of use, and high stability. Karaboga and Basturk [34] used it to optimize multivariable functions and compared their results with those produced by GA, PSO, DE, and particle swarm inspired evolutionary algorithm (PS-EA). Other studies have also compared the performance of ABC with other existing heuristic algorithms [35,36], and the results reported in the literature suggest that ABC is more effective than other methods in some engineering optimization problems [37–39]. Since this algorithm has been proven to be successful in dealing with large optimization problems, it seems natural to extend it to structural parameter identification, considering the problems associated with the limitations that occur in real life applications such as imcomplete set of noisy measurements.

In this chapter, the ABC algorithm as well as its modified version are presented in the context of input-output structural SI. The chapter is organized as follows. Section 2.2 describes the formulation of structural SI as an optimization problem. Section 2.3 presents a modified version of ABC algorithm which is suitable for structural parameter estimation. Section 2.4 discusses the numerical identification results obtained from two linear systems and a nonlinear system identified with full and partial measurements under noise free and noise polluted cases. In Section 2.5, we present the observations and the conclusions.

2.2 Optimization formulation of system identification

In the case of a structural system, the identification of its structural properties can be regarded as an optimization problem in which an objective function, e.g., the error between the actual measured structural response and the estimated response of a structural model, is defined and the parameters of such a model determined so to optimize (usually to maximize or to minimize) the objective function.

Let us consider that $\mathbf{u}(t_p)$ and $\mathbf{y}(t_p)$ are two vectors containing the input and the corresponding output, respectively, of a general system at a generic instant t_p, where $p = 1, 2, \cdots, N$ (N is the total number of sampling points). Let us now denote with $\hat{\mathbf{y}}(t_p)$ the estimated system's response obtained from a parametric system's model using the associated measured input $\mathbf{u}(t_p)$, expressed as:

$$\hat{\mathbf{y}}(t_p) = f\left(\mathbf{u}(t_p), \hat{\boldsymbol{\theta}}\right) \tag{2.1}$$

where $\hat{\boldsymbol{\theta}} = \{\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{N_\theta}\} \in \mathbb{R}^{N_\theta}$ represents a vector containing the estimated values of the model's unknown parameters $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_{N_\theta}\} \in \mathbb{R}^{N_\theta}$: in a mechanical and/or structural system, for example, the elements θ_j ($j = 1, 2, \ldots, N_\theta$) could be the mass, and/or damping and/or stiffness parameters.

The objective of any identification procedure is to find the best estimates $\hat{\boldsymbol{\theta}}$ of the structural parameters $\boldsymbol{\theta}$ so to minimize the error between the measured response $\mathbf{y}(t_p)$ and the predicted (or estimated) response $\hat{\mathbf{y}}(t_p)$ over the entire time-history. This process can be visualized in Figure 2.1. In identification problems that rely on the dynamic measurements of the response, the objective function to be minimized through the optimization process can be formulated as a sum of the mean square error (MSE) between the measured and predicted responses, such as:

$$g(\hat{\boldsymbol{\theta}}) = \frac{1}{N_o N} \sum_{i=1}^{N_o} \sum_{p=1}^{N} \left[\hat{y}_i(t_p) - y_i(t_p)\right]^2 \tag{2.2}$$

where N_o is the number of observed/recorded time histories of the response which depends on the number of sensors available in the test (it can range from 1 (only one sensor available) to the total number of degrees-of-freedom (DOF) of the system (full set of sensors)).

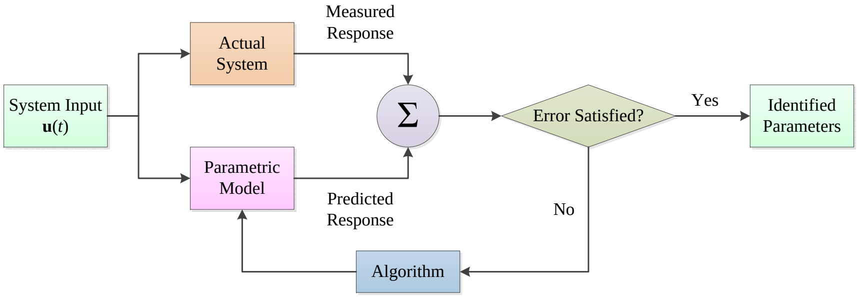


Figure 2.1: The process of system identification as an optimization problem.

Nevertheless, Stavroulakis and Antes [40] recommended to use the logarithmic transformation of $g(\boldsymbol{\theta})$ as the objective function instead of directly optimizing Equation (2.2), and this has been proven to be beneficial in solving inverse optimization problems:

$$G(\boldsymbol{\theta}) = \ln\left[g(\boldsymbol{\theta}) + \eta\right] \tag{2.3}$$

where η is a small positive constant, which prevents the appearance of a negative infinity value in $G(\boldsymbol{\theta})$ (e.g., $\eta = 1 \times 10^{-6}$). Therefore, the overall identification problem can then be summarized as:

$$\text{Find } \hat{\boldsymbol{\theta}} = \{\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{N_\theta}\} \in \Gamma \text{ such that } G(\hat{\boldsymbol{\theta}}) \to \min \tag{2.4}$$

where Γ is the feasible n-dimensional parameter search space:

$$\Gamma = \{\hat{\boldsymbol{\theta}} \in \mathbb{R}^n | \theta_j^{\min} \leq \hat{\theta}_j \leq \theta_j^{\max},\ \forall j = 1, 2, \ldots, N_\theta\} \tag{2.5}$$

where N_θ is the number of parameters to be identified, $\theta_j^{\min}$ is the lower bound of the j-th parameter and $\theta_j^{\max}$ is the corresponding upper bound.

Equations (2.3)–(2.5) show that the problem of identification can be treated as a linearly constrained nonlinear optimization problem. Hence, because of the irregularity of the multi-dimensional surface $G(\hat{\boldsymbol{\theta}})$, an efficient identification algorithm must be capable of handling problems related to multiple solutions (more than one minimum) and to local vs. global minima (many local optima with high complexities may exist in the search space). While some of these problems (e.g., multiple solutions) can be resolved independently of the algorithm used (e.g., by choosing proper sensor locations that guarantees uniqueness of the solution), others, like local *vs.* global minima, must be handled directly by the chosen algorithm. Hence heuristic algorithms with both powerful global and local search capabilities are recommended.

2.3 Artificial bee colony (ABC) algorithms

The basic architecture of the ABC algorithm was inspired by the idea of simulating honey bees' foraging behavior (exploration and exploitation of "food sources") [33–36]. Three search phases were proposed, named after the type of bees and their corresponding foraging duties, *viz.*, the Employed Phase, the Onlooker Phase and the Scout Phase. In applying ABC to system identification problems, the starting point is to accept the equivalence between "food source location" and "structural model". This can be explained by considering that a food source location is similar to a structural model in which it is characterized by a set of parameters, such as the distance from the hive, the orientation with respect to the sun, etc.; in an optimization scheme, a location represents a possible solution. Finding the "best" possible food source is equivalent to finding the set of parameters that corresponds to the optimized solution.

The detailed procedure of the ABC algorithm and its modified version to solve the inverse SI optimization problems is presented in the following subsections.

2.3.1 Solution Initialization

The initial step of the ABC algorithm is to select an initial population of possible food source locations or, in the context of system identification, of possible structural parametric models. This initial set, denoted with $\mathbf{\Theta}$ of the order $N_{pop} \times N_\theta$, can be randomly generated keeping the parameters within some specified ranges, i.e., the generic i-th possible solution (either intended as food source location or structural model) indicated by a vector $\mathbf{\Theta}_i = \{\Theta_{i1}, \Theta_{i2}, \ldots, \Theta_{iN_\theta}\}$ can be generated by:

$$\Theta_{ij} = \theta_j^{\min} + \text{rand} \cdot (\theta_j^{\max} - \theta_j^{\min}) \tag{2.6}$$

where Θ_{ij} indicates the j-th component of the i-th candidate solution; $i = 1, 2, \ldots, N_{pop}$ and $j = 1, 2, \cdots, N_\theta$; N_{pop} is the size of the population (bee colony) and N_θ is the number of parameter to be optimized; $\theta_j^{\max}$ and $\theta_j^{\min}$ are the upper and lower bound, respectively, of the j-th parameter; 'rand' represents a random number uniformly sampled from [0, 1]. Then, the initial set is put into the Employed Phase search for solution upgrading in the following iterations. The maximum number of iterations for the algorithm termination is denoted as N_{max}.

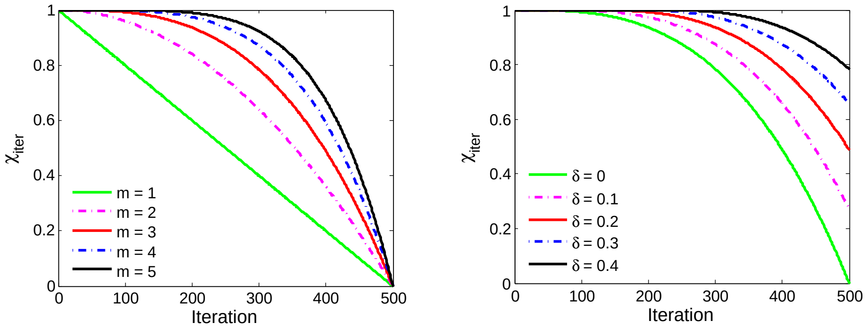
2.3.2 Employed Phase search

Starting from the initial set of solutions, a search for "improved solutions" or "better fit solutions" in the neighborhood of each existing one is conducted. This is equivalent to the task, carried out by "employed bees" or "employed foragers", of trying to improve their food sources by a neighborhood search. Hence, for each original solution, a new candidate solution is randomly generated and the objective function value of such a new solution is computed by Equation 2.3. If the fitness value of the candidate is better than that of the previous solution, then the previous solution will be updated to the new candidate one; otherwise, the previous solution is kept. At the generic iterative step, the new candidate solution represented by a vector $\mathbf{\Theta}_i^{up}$ is generated using the following updating procedure:

$$\Theta_{ij}^{up} = \Theta_{ij} + 2\lambda(\text{rand} - 0.5) \cdot (\Theta_{ij} - \Theta_{kj}) \tag{2.7}$$

where $i = 1, 2, \ldots, N_{pop}$; k $(k \neq i)$ is a random index selected from the integer set $\{1, 2, \ldots, N_{pop}\}$; j represents the j-th parameter to be updated; Θ_{ij}^{up} indicates the j-th component of the i-th updated candidate solution. And λ is a range control parameter (usually $\lambda = 1$ in the standard ABC algorithm) and is responsible for the nature of the search: if λ is a small number, then the algorithm will adapt more to a local search; otherwise, the algorithm will function for more global searches. It would then be preferable to have a parameter that changes value during the search process, from large values at the beginning of the search (where it is important to explore the entire horizon (space)) to small values towards the end of the search (where it is important to refine the convergence to the final optimal solution).

In order to balance the search preference and to control the convergence rate as the



(a) The effect of m on χ_{iter} with $\delta = 0$ (b) The effect of δ on χ_{iter} with $m = 3$

Figure 2.2: The relationship between iterations and the nonlinear factor.

iteration progresses, a nonlinear factor χ_{iter}, defined as:

$$\chi_{iter} = \begin{cases} 1, & \text{if} \quad N_{iter} < \delta N_{max} \\ 1 - \left| \dfrac{N_{iter} - \delta N_{max}}{N_{max}} \right|^{m}, & \text{if} \quad N_{iter} \geq \delta N_{max} \end{cases} \tag{2.8}$$

is proposed to modify the search range, where $iter$ denotes the current iteration number, δ is a parameter controlling the termination point of the nonlinear decreasing behavior which lies in the range of [0, 1], and m is the power exponent representing the nonlinear decreasing rate. By changing the values of the parameters δ and m, it is possible to control the nature of the search process.

Figure 2.2 depicts the behavior of the nonlinear factor χ_{iter} as a function of the iteration number, $iter$, for different values of m and δ ($N_{max} = 500$). Both plots show a nonlinear, decreasing behavior of χ_{iter} as the iteration number increases: at the beginning (small iteration number), the value of χ_{iter} approaches 1, favoring the search over the entire search space. However, as the number of iterations increases, χ_{iter} approaches, more or less rapidly, to zero, inducing small variation to the current solution. The idea behind χ_{iter} is to free the method, at its initial steps, to search over the entire search space and to allow for faster convergence during the local search phase which take place towards the end of the iteration process. To avoid stagnation around local minima, a concept equivalent to the figure of a "scout bee" in a hive will later be described.

This newly defined χ_{iter} factor is now used to adapt the variability of the new candidate solution that can be expressed as:

$$\Theta_{ij}^{up} = \Theta_{ij} + 2\chi_{iter}(\text{rand} - 0.5) \cdot (\Theta_{ij} - \Theta_{kj}) \tag{2.9}$$

In order to select which i-th food source to retain (either the current one, $\boldsymbol{\Theta}_i$, or the new candidate, $\boldsymbol{\Theta}_i^{up}$), a “greedy” selection based on roulette wheel selection strategy is carried out by comparing the fitness values corresponding to $\boldsymbol{\Theta}_i$ and $\boldsymbol{\Theta}_i^{up}$. The fitness function can be expressed as:

$$fit(\boldsymbol{\Theta}_i) = \begin{cases} [1 + G(\boldsymbol{\Theta}_i)]^{-1}, & \text{if} \quad G(\boldsymbol{\Theta}_i) > 0 \\ 1 + |G(\boldsymbol{\Theta}_i)|, & \text{if} \quad G(\boldsymbol{\Theta}_i) \leq 0 \end{cases} \tag{2.10}$$

where $G(\boldsymbol{\Theta}_i)$ and $fit(\boldsymbol{\Theta}_i)$ are the objective function and the fitness values, respectively, for the solution $\boldsymbol{\Theta}_i$.

2.3.3 Roulette Wheel selection

After the Employed Phase search, a Roulette Wheel strategy is applied to select solutions for the Onlooker Phase search. Once all the possible candidate solutions have been identified and the values of the corresponding fitness functions have been determined, then it is possible to associate a probability of selection to each candidate. Here, a fitness-proportional selection process is used so that the probability of selection associated with the i-th candidate is:

$$p_i = \frac{fit(\boldsymbol{\Theta}_i)}{\sum_{i=1}^{N_{pop}} fit(\boldsymbol{\Theta}_i)} \tag{2.11}$$

In this way, a food source with high value of the fitness function will have a higher probability of being selected.

At this point, to maintain a certain randomness in the process, the new set of N_{pop} solutions $\boldsymbol{\Theta}_i = \{\Theta_{i1}, \Theta_{i2}, \ldots, \Theta_{iN_\theta}\}$ updated by the Employed Phase search is selected at random by generating a uniformly distributed random number r_i $(i = 1, 2, \ldots, N_{pop})$, between 0 and 1, and comparing it with the corresponding probability p_i. If the probability associated with the i-th solution, p_i, is greater than the associated random number r_i, then the i-th solution is selected and passed to the Onlooker Phase search; otherwise, the corresponding solution is kept of its original values.

2.3.4 Onlooker Phase search

For each selected solution set, a new candidate solution is generated following Equation (2.9) in the Onlooker Phase Search and its fitness compared to the value of the original one. The solution with the largest fitness value will be chosen as representative of the i-th solution for the next iteration.

To this end, the new set of N_{pop} possible solution $\mathbf{\Theta}_i$ will represent the starting point of the next iteration. The solution (e.g., food source location or system's model) with the largest fitness value among all the N_{pop} locations is recorded and compared with that of the previous iteration.

2.3.5 Scout Phase search

It might happen that one of the possible solutions, let's say the l-th solution, is not improved after a given number, N_{lim}, of iterations while the other solutions keep improving their fitness: this can happen if the l-th solution represents a local minimum of the fitness function. At this point, it will be "abandoned" and a new solution, $\mathbf{\Theta}_i^{sc}$, will be obtained, with the j-th element determined by Equation (2.6). The superscript sc is chosen to indicate the candidates "scout bee" nature, which is a bee with the task of searching for a new food source. The use of such a "scout bee" solution is used to enhance the global search capability of the algorithm, providing a way to exit stagnation points (e.g., local minima). If the newly found i-th solution is "better" than the abandoned one, the search will continue using the new location as the new i-th one. Otherwise, a new solution will be generated and tested again until a new better solution is found.

2.3.6 Pseudocode of the ABC algorithm

To summarize, the basic steps of this modified ABC (MABC) algorithm are shown in Algorithm 1. It is noteworthy that the difference between the standard and the modified versions is dependent of the use of χ_{iter} in ABC. The MABC algorithm can be reduced to the standard version by setting $\chi_{iter} \equiv 1$.

This MABC algorithm promises to be a robust optimization algorithm for both global and local searches. In the identification of a structural system, the unknown model's pa-

Algorithm 1 – The basic steps of the MABC algorithm

```
***Initialization***: initialize the set of possible solutions using Equation (2.6) (following a uniformly random distribution in a specified search space), and compute the objective function using Equation (2.3) and the fitness value using Equation (2.10) of the initial population set;
$N_{iter} \leftarrow 0$;
**while** ($N_{iter} \leq N_{max}$) **do**
    ***Employed Phase search***: for each possible solution, create a new candidate solution by Equation (2.9), evaluate its objective function by Equation (2.3) and its fitness value by Equation (2.10), and select the one with the best fitness as the new possible solution;
    ***Roulette Wheel selection***: calculate the probability of each solution using Equation (2.11) and select possible solutions from those obtained in the Employed Phase by using the Roulette Wheel selection strategy;
    ***Onlooker Phase search***: create new candidates by Equation (2.9), evaluate each of them by Equations (2.3) and (2.10), and select the better ones as new possible solutions;
    **if** $N_{lim}$ is met **then**
        ***Scout Phase search***: investigate a new solution by Equation (2.6), replace the original one by it, and evaluate it by Equations (2.3) and (2.10);
    **end if**
    $N_{iter} \leftarrow N_{iter} + 1$;
**end while**
```

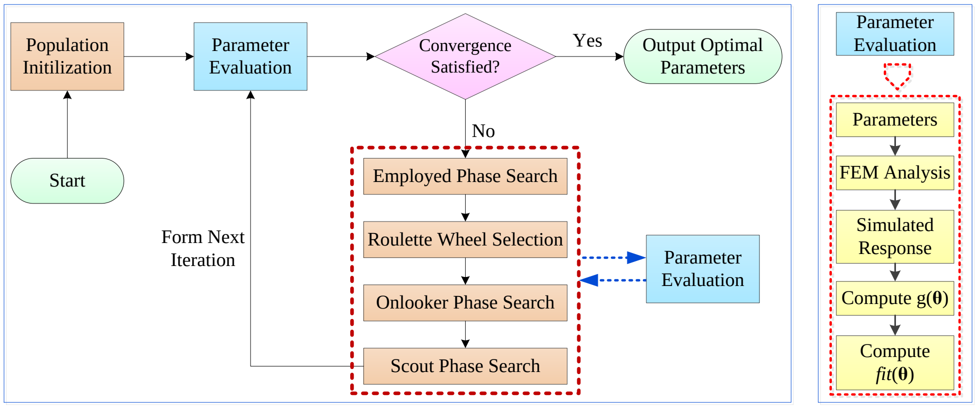


Figure 2.3: The flow chart of system identification using MABC.

rameters, for example the mass, stiffness and damping parameters, and/or coefficients describing nonlinear behavior, play the role of parameters Θ_{ij} that need to be estimated and the parametric space is the solution search space in the optimization process.

A schematic representation of the proposed algorithm in the context of system identification is presented in Figure 2.3.

2.4 Numerical applications

In order to investigate the effective applicability of the proposed approach to structural SI, two linear systems (5-DOF and 20-DOF) and a nonlinear system (2-DOF) are discussed in this work. The three systems are 2-D shear frame type structures with lumped masses at each floor. Comparisons of the proposed technique for parametric identification with other methodologies suggested in literature [28, 29, 41, 42] are carried out in this part.

To verify the effectiveness of MABC for SI with limited output data measurements and to learn the influence of data availability on the performance of this algorithm, different sets of time histories of the structural response have been considered: both the case when the time histories are measured at every DOF (full instrumentation setup) and the case when the time histories are available at some DOFs (partial instrumentation setup) have been analyzed. Since accelerometers are commonly used in dynamic field testing, the structural response is presented here in the form of time histories of the structural accelerations. To provide a fairly reasonable search space for the parameters, the lower bounds of the search space are taken to be half and the upper bounds twice the actual values of the parameters so that the search space can be defined as $\boldsymbol{\theta}^*/2 \leq \Gamma \leq 2\boldsymbol{\theta}^*$, where $\boldsymbol{\theta}^*$ is the parameter vector representing the true values. To analyze the effect of measurement noise on the identification of the parameters and so on the effectiveness of the MABC algorithm, various noise pollution levels in both input and output data have been taken into consideration in the identification analysis: both input and output (I/O) signals have been polluted with Gaussian zero-mean white noise sequences. The root mean square (RMS) values of the noise sequences are defined as some certain percentages (i.e. 0% (no noise), 5%, and 10%) of the original time history signals. To wit, the noisy input-output measurement takes the form as follows:

$$\mathbf{y}^{measure} = \mathbf{y}^{correct} + \mathbf{r} \times \beta_y \times q\% \tag{2.12}$$

where $\mathbf{y}^{correct}$ is the correct signal, $\mathbf{y}^{measure}$ the measured noisy signal, $\mathbf{r}$ the standard zero mean Gaussian white noise sequence, β_y the root-mean-square (RMS) of the correct signal, and q the level of noise corruption. It is noteworthy that, in addition to the white noise cases, the cases of colored input and of colored measurement noise were also considered. However, the identification results for the colored input and colored noise are almost identical or even better than those obtained for the corresponding white noise cases and so have not been included in the chapter for space limitation. In some previous studies on system identification, the assumption of an a-priori known mass matrix has been used quite frequently. This assumption is justified by the assertion that a sufficiently accurate estimation of the mass can be obtained from the structural drawing. Here, to test the identification capabilities of the proposed algorithm, both the case of a priori known mass distribution and the case of the unknown mass distribution have been considered.

In applying MABC to SI, the most time-consuming operation is represented by solving the 2nd order ODEs during the fitness evaluations, whereas the time taken by the parameter updates is comparatively negligible. In the first two examples, the systems' responses are simulated using the implicit Newmark-β integration method, while in the third example, obtained by solving (integrating) the ODEs using the 4th to 5th order embedded Runge-Kutta integration method with adaptive step-size after transforming the ODEs into state space form. Moreover, since the MABC is a stochastic algorithm, 20 independent runs have been conducted so as to obtain statistical insights into the process. The numerical analyses are programmed in MATLAB® (The MathWorks, Inc., MA, USA) on a standard Intel (R) Core (TM) 2 Q9550 2.83GHz personal PC with 2G RAM.

2.4.1 A 5-DOF linear system

To study the performance of the MABC algorithm, first let us consider a 5-DOF linear structural system, modeled as a shear type frame structure (Figure 2.4). The equation of motion for such a system can be written as:

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{L}\mathbf{f}(t) \tag{2.13}$$

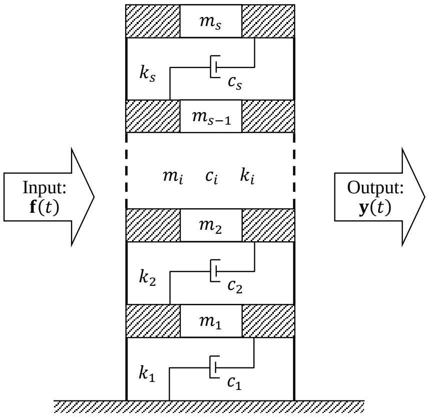


Figure 2.4: A s-DOF shear type linear structural system.

where $\mathbf{x}(t)$, $\dot{\mathbf{x}}(t)$ and $\ddot{\mathbf{x}}(t)$ are the system's response components representing displacement, velocity, and acceleration vectors, respectively; $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are the mass, damping, and stiffness matrices, of the order $s \times s$, of the system's model. For shear type models, the mass matrix is diagonal and the stiffness matrix is tri-diagonal; here we assume the damping matrix also has a tri-diagonal structure analogous to the stiffness matrix. $\mathbf{f}(t)$ is the external (force) input vector and $\mathbf{L}$ is the input location matrix. The properties of the system in this example are shown in Table 2.1. For the known mass identification case, the parameters to be identified in this example can be described by a set of ten variables:

$$\boldsymbol{\theta} = \{c_1, \ldots, c_5, k_1, \ldots, k_5\} \tag{2.14}$$

while for the unknown mass case, the system can be fully described by a set of fifteen variables:

$$\boldsymbol{\theta} = \{m_1, \ldots, m_5, c_1, \ldots, c_5, k_1, \ldots, k_5\} \tag{2.15}$$

In this numerical example, the structure is assumed to be excited by an input force sequence $f_3(t)$, which is supposed to be a known white noise sequence of 5 seconds (following Gaussian distribution with zero mean and a RMS scaled to 1 N) acting on the 3rd DOF. The sampling time is $\Delta t = 0.005$ sec. In the full output scenario, all the floor accelerations are measured, whereas in the partial output scenario only the accelerations at the 2nd, 3rd and 4th DOFs are available. The choice of such floors was dictated by the requirements that guarantee uniqueness of the identified solution [31]. The values of the algorithm parameters

Table 2.1: The structural properties of a 5-DOF linear system.

Floor (Mode) number	Mass (kg)	Damping (kNs/m)	Stiffness (kN/m)	Frequency (Hz)	Damping ratio (%)
1	0.1	0.5	130	1.6333	1.9736
2	0.1	0.5	130	4.7676	5.7608
3	0.1	0.5	130	7.5157	9.0813
4	0.1	0.5	130	9.6549	11.6661
5	0.1	0.5	130	11.0119	13.3058

used in the MABC are: $N_{pop} = 40$ (for the known mass case), $N_{pop} = 80$ (for the unknown mass case), $N_{max} = 1000$, $N_{lim} = 200$, $m = 3$ and $\delta = 0.2$. The settings of m and δ follow the assumption that χ_{iter} starts to nonlinearly decrease not early than 1/3 of N_{max} iterations and reaches the final value not smaller than 0.5.

For comparison purposes, the standard ABC algorithm was tested, in which the algorithmic parameters are the same as those used in the modified ABC algorithm (apart from the nonlinear factor χ_{iter}). A classic rank-based genetic algorithm (GA) was also implemented here by using the MATLAB® GA Toolbox^{TM}. In this GA, the population size is chosen to be equal to N_{pop}, the maximum generation equals to N_{iter}. The rate of crossover was selected equal to 0.8 while a Gaussian function with zero mean value was adopted for mutation strategy with a scale factor of 0.5 and shrink factor of 0.75 [43]. For the unknown mass case with complete instrumentation, an algorithm presented in [44, 45], based on the identification of a first-order model and on its convertion to second order, is used for comparison purpose.

Simulation results for the known mass case are summarized in Table 2.2 and Table 2.3 corresponding to full output scenario and partial output scenario, respectively. It can be observed that the MABC algorithm is able to estimate the correct values of the structural parameters in the noise free case even with partial measurements, whereas the results obtained by the standard ABC algorithm and GA have slight errors. Furthermore, it is seen from Table 2.2 that the MABC algorithm yields "finer" results (with errors from 0% to 0.6%) than those obtained via the standard ABC algorithm (with errors from 0.1% to 1.5%) for

Table 2.2: Results for 5-DOF known mass system in the full output scenario.

		0% noise				10% noise			
Params.	True	ABC		MABC		ABC		MABC	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
c_1	0.5	0.4996 (0.1)	0.002	0.5000 (0.0)	0	0.4927 (1.5)	0.026	0.4989 (0.2)	0.022
c_1	0.5	0.5000 (0.0)	0.001	0.5000 (0.0)	0	0.4996 (0.1)	0.017	0.4986 (0.3)	0.016
c_1	0.5	0.5003 (0.1)	0.001	0.5000 (0.0)	0	0.5020 (0.4)	0.010	0.4994 (0.1)	0.010
c_1	0.5	0.5000 (0.0)	0.001	0.5000 (0.0)	0	0.4994 (0.1)	0.006	0.4970 (0.6)	0.009
c_1	0.5	0.5000 (0.0)	0.001	0.5000 (0.0)	0	0.4985 (0.3)	0.014	0.5011 (0.2)	0.009
k_1	130	130.03 (0.02)	0.091	130.00 (0.0)	0.002	130.19 (0.1)	0.645	129.81 (0.1)	0.993
k_2	130	129.96 (0.03)	0.091	130.00 (0.0)	0.002	129.72 (0.2)	1.064	129.95 (0.04)	0.732
k_3	130	130.00 (0.0)	0.034	130.00 (0.0)	0.001	129.95 (0.04)	0.637	130.03 (0.02)	0.748
k_4	130	130.00 (0.0)	0.062	130.00 (0.0)	0.001	130.22 (0.2)	0.537	130.25 (0.2)	0.475
k_5	130	130.00 (0.0)	0.070	130.00 (0.0)	0.001	129.95 (0.04)	0.612	129.92 (0.1)	0.701
Max error (%)		0.1		0		1.5		0.6	

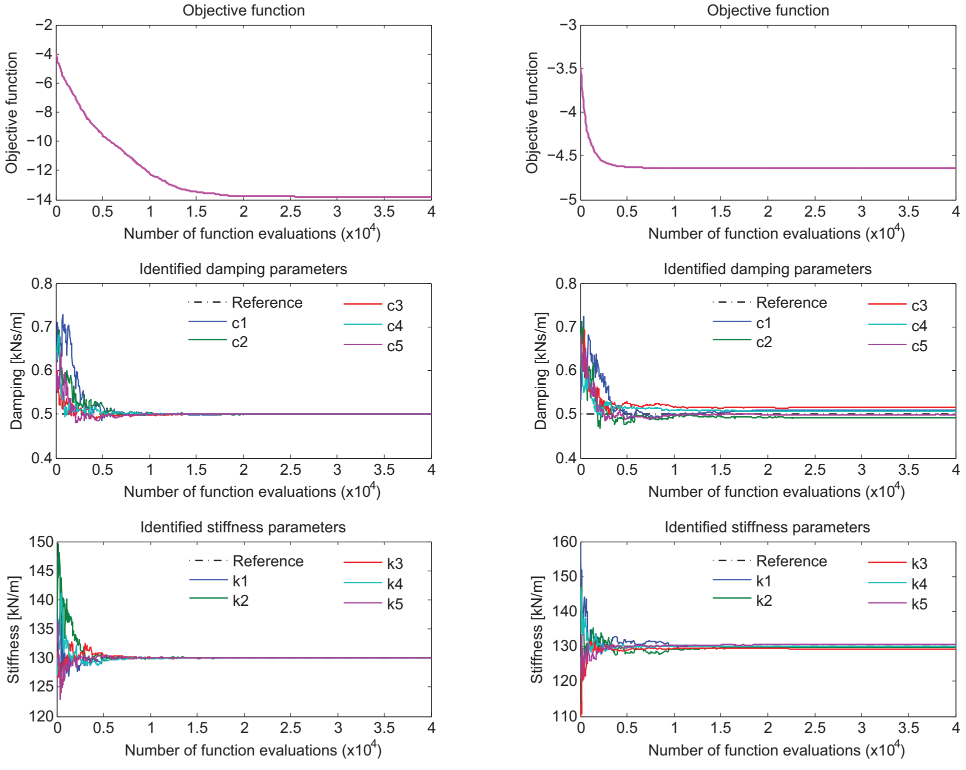
Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

Table 2.3: Results for 5-DOF known mass system in the partial output scenario.

		0% noise				10% noise			
Params.	True	GA		MABC		GA		MABC	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
c_1	0.5	0.5149 (2.9)	0.054	0.4999 (0.02)	0	0.5238 (4.8)	0.0746	0.5089 (1.8)	0.049
c_2	0.5	0.4885 (2.3)	0.049	0.5000 (0.0)	0	0.4821 (3.6)	0.0397	0.4913 (1.7)	0.034
c_3	0.5	0.5076 (1.5)	0.020	0.5001 (0.02)	0	0.5261 (5.2)	0.0575	0.5158 (3.2)	0.029
c_4	0.5	0.4988 (0.2)	0.007	0.5000 (0.0)	0	0.4690 (6.2)	0.0801	0.5075 (1.5)	0.022
c_5	0.5	0.4907 (1.9)	0.020	0.5000 (0.0)	0	0.5281 (5.6)	0.0749	0.4993 (0.1)	0.022
k_1	130	132.41 (1.9)	1.975	130.01 (0.01)	0.020	132.59 (2.0)	2.4832	129.83 (0.1)	1.906
k_2	130	133.44 (2.6)	2.002	129.99 (0.01)	0.018	132.03 (1.6)	2.0798	129.70 (0.2)	2.195
k_3	130	129.73 (0.2)	0.636	130.00 (0.0)	0.011	133.22 (2.5)	2.9945	129.31 (0.5)	1.479
k_4	130	131.14 (0.9)	1.024	130.00 (0.0)	0.007	130.84 (0.6)	0.9761	130.05 (0.1)	1.182
k_5	130	132.03 (1.6)	1.863	130.00 (0.0)	0.008	132.57 (2.0)	2.6185	130.48 (0.4)	0.908
Max error (%)		2.9		0.02		6.2		3.2	

Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

both noise free and 10% RMS noise cases. However, both versions of ABC algorithm seem relatively insensitive to noise corruption. Similarly, in the partial output scenario, both the MABC algorithm and GA perform well and yield good estimation results with small errors, ranging from 0.02% to 3.2% for MABC, and from 2.9% to 6.2% for GA, yet with the modified ABC algorithm performing slightly better than GA. Even for the 10% RMS noise polluted case with partial measurements, the results obtained by the MABC algorithm show that estimation errors pertaining to damping and stiffness parameters remain very small with maximum values of 3.2% and 0.5%, respectively. Figure 2.5 presents a noise free and noise polluted comparison of identified damping and stiffness parameters for the known mass case with partial measurements: the parameter estimates are observed to converge to the true values after 1×10^4 forward analyses, which demonstrates the accuracy and the high efficiency of the proposed approach.



(a) Noise free

(b) 10% RMS noise in I/O signals

Figure 2.5: MABC convergence for the known mass case with partial measurements.

Table 2.4: Results for 5-DOF unknown mass system in the full output scenario.

		0% noise			10% noise		
Params.	True	Ref. [42,44,45]	MABC		Ref. [42,44,45]	MABC	
		Mean	Mean	Std.	Mean	Mean	Std.
m_1	0.1	0.1000 (0.0)	0.1001 (0.1)	0.003	0.0988 (1.2)	0.1030 (3.0)	0.003
m_2	0.1	0.1000 (0.0)	0.1003 (0.3)	0.001	0.0974 (2.6)	0.1020 (2.0)	0.002
m_3	0.1	0.1000 (0.0)	0.1001 (0.1)	0.000	0.0999 (0.1)	0.1011 (1.1)	0.001
m_4	0.1	0.1000 (0.0)	0.1008 (0.8)	0.002	0.0973 (2.7)	0.1035 (3.5)	0.002
m_5	0.1	0.1000 (0.0)	0.1001 (0.1)	0.003	0.0975 (2.5)	0.1029 (2.9)	0.004
c_1	0.5	0.5000 (0.0)	0.4886 (2.3)	0.038	0.6126 (22.5)	0.4906 (1.9)	0.057
c_2	0.5	0.5000 (0.0)	0.5069 (1.4)	0.022	0.7481 (49.6)	0.5084 (1.7)	0.030
c_3	0.5	0.5000 (0.0)	0.4967 (0.7)	0.015	0.5014 (0.3)	0.5127 (2.5)	0.036
c_4	0.5	0.5000 (0.0)	0.508 (1.6)	0.014	0.4904 (1.9)	0.5060 (1.2)	0.030
c_5	0.5	0.5000 (0.0)	0.4989 (0.2)	0.010	0.5745 (14.9)	0.5127 (2.5)	0.029
k_1	130	130.00 (0.0)	129.99 (0.1)	5.448	129.94 (0.1)	134.38 (3.4)	5.522
k_2	130	130.00 (0.0)	130.11 (0.1)	2.367	120.43 (7.4)	132.49 (1.9)	2.929
k_3	130	130.00 (0.0)	130.20 (0.2)	0.961	129.25 (0.6)	131.70 (1.3)	2.129
k_4	130	130.00 (0.0)	130.72 (0.6)	2.059	130.48 (0.4)	133.37 (2.6)	2.087
k_5	130	130.00 (0.0)	130.28 (0.2)	2.911	123.86 (4.7)	134.04 (3.1)	3.381
Max error (%)		0	2.3		49.6	3.5	

Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

A more difficult case for the 5-DOF system is when the mass matrix of the system is also unknown and needs to be identified. Tables 2.4 and 2.5 present the results of the identification for both the full output scenario and the partial output scenarios, respectively. In the full output scenario, both the MABC algorithm and the algorithm proposed in [42] provide good identification results, with a maximum error of 2.3%, for the noise free case. However, in the 10% RMS noise corruption in the I/O signals, the MABC algorithm estimated errors of mass, stiffness, and damping parameters range from 1.1% to 3.5%, from 1.2% to 2.5%, and 1.3% to 3.4%, respectively, still providing excellent identification even for large noise levels. On the contrary, while the algorithm in [42,44,45] is able to estimate mass and stiffness well, it yields very large errors in damping coefficients identification (up to 49.6%). In the partial output scenario, the methodology in [42,44,45] cannot be applied and so the results of the proposed ABC algorithm are compared with those obtained through the GA previously considered. Again, it was found that the results by the MABC algorithm

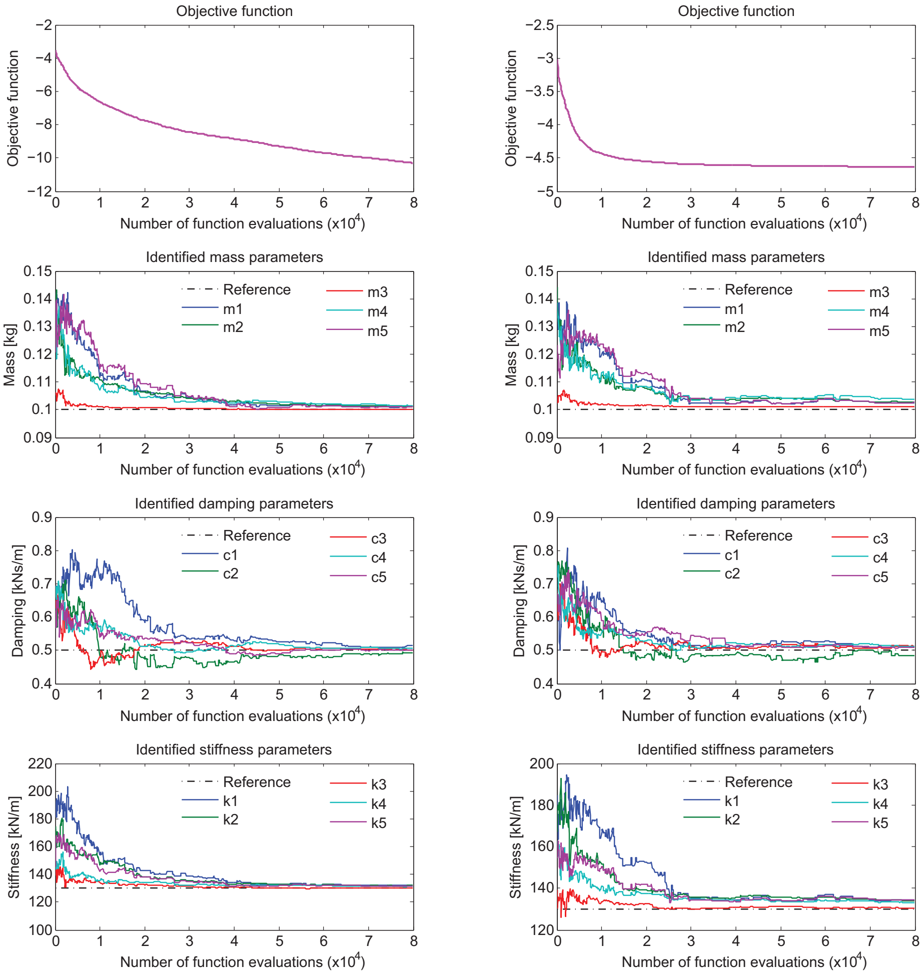
Table 2.5: Results for 5-DOF unknown mass system in the partial output scenario.

Params.	True	0% noise				10% noise			
		GA		MABC		GA		MABC	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
m_1	0.1	0.1034 (3.4)	0.008	0.1012 (1.2)	0.003	0.0965 (3.5)	0.007	0.1025 (2.5)	0.006
m_2	0.1	0.1032 (3.2)	0.007	0.1013 (1.3)	0.002	0.1010 (1.0)	0.004	0.1028 (2.8)	0.003
m_3	0.1	0.1000 (0.0)	0.000	0.1001 (0.1)	0.001	0.1023 (2.3)	0.002	0.1010 (1.0)	0.001
m_4	0.1	0.1026 (2.6)	0.007	0.1016 (1.6)	0.003	0.1008 (0.8)	0.003	0.1039 (3.9)	0.004
m_5	0.1	0.1026 (2.6)	0.010	0.1012 (1.2)	0.004	0.0951 (4.9)	0.016	0.1026 (2.6)	0.006
c_1	0.5	0.5309 (6.2)	0.091	0.5144 (2.9)	0.045	0.4207 (15.9)	0.122	0.5100 (2.0)	0.072
c_2	0.5	0.4685 (6.3)	0.102	0.4926 (1.5)	0.050	0.5616 (12.3)	0.103	0.4845 (3.1)	0.098
c_3	0.5	0.5288 (5.8)	0.078	0.5021 (0.4)	0.023	0.4694 (6.1)	0.088	0.5132 (2.6)	0.049
c_4	0.5	0.4988 (0.2)	0.009	0.5076 (1.5)	0.017	0.5072 (1.4)	0.059	0.5108 (2.2)	0.035
c_5	0.5	0.4907 (1.9)	0.055	0.4995 (0.1)	0.019	0.5035 (0.7)	0.019	0.5097 (1.9)	0.037
k_1	130	137.41 (5.7)	7.824	132.31 (1.8)	5.215	121.47 (6.6)	12.542	134.41 (3.4)	7.271
k_2	130	133.44 (2.6)	3.085	131.95 (1.5)	3.643	130.46 (0.4)	2.284	133.90 (3.0)	4.585
k_3	130	129.73 (0.2)	0.712	130.134 (0.1)	0.815	135.09 (3.9)	3.481	130.51 (0.4)	2.187
k_4	130	131.14 (0.9)	2.397	131.29 (1.0)	2.171	132.53 (1.9)	2.309	133.13 (2.4)	2.907
k_5	130	134.02 (3.1)	7.085	131.66 (1.3)	4.511	125.72 (3.3)	6.156	134.53 (3.5)	5.286
Max error (%)		6.3		2.9		15.4		3.9	

Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

are much better than those obtained by GA for both the noise free and 10% RMS noise contamination cases despite noise corruption and lack of measurements. The maximum errors obtained by GA jumps from 6.3% to 15.4% as noise level increases from noise free to 10% RMS noise, whereas those obtained by MABC remain basically the same (from 2.9% to 3.9%). It is obvious that the MABC algorithm also gives much smaller deviation values than those by GA (Table 2.5), showing a good level of accuracy despite measurement noise and lack of data.

Similar to Figure 2.5 for the above known mass case, Figure 2.6 shows a noise free and noise polluted comparison of the identified parameters for the unknown mass case with partial measurements. In both cases, convergence is quite fast (after about 4×10^4 forward analyses). It can be seen that the maximum estimate error often exists in mass and damping parameters and signal noise corruption has usually larger impact on those parameters' estimation.



(a) Noise free

(b) 10% RMS noise in I/O signals

Figure 2.6: MABC convergence for the unknown mass case with partial measurements.

To test the robustness of the proposed technique in each independent run, the probability density functions (PDFs) of the normalized parameters have been obtained and are presented in Figure 2.7. The normalized parameter is defined as the ratio of the identified parameter value to the true value. In Figure 2.7, the PDFs of the normalized parameters are obtained from 20 independent runs in the unknown mass case with partial output and 10% RMS noise corruption. It is seen that the normalized parameters approximately follow normal distributions with the mean values very close to 1 and small standard deviations, which verifies that the identified parameters are very close to the true values.

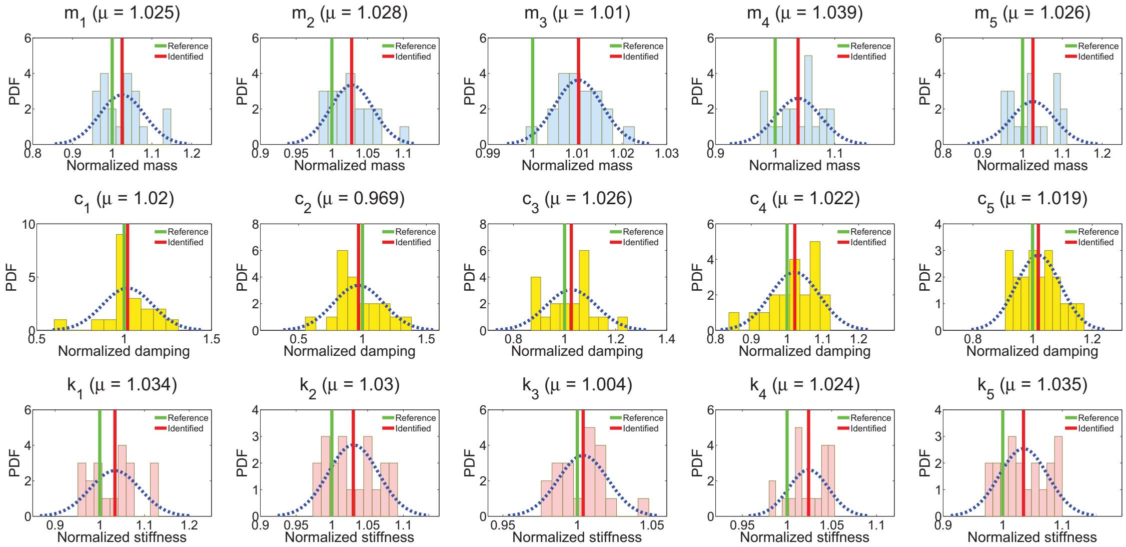


Figure 2.7: The PDFs of the normalized parameters obtained from 20 independent runs in the unknown mass case with partial output and 10% RMS noise corruption.

2.4.2 A 20-DOF linear system

To further compare the performance of the modified version of ABC with other heuristic methodologies (e.g., GAs, PSO, and DE) and to verify its applicability to SI as the number of parameters to be identified increases, a two-dimensional 20-DOF shear type linear system is analyzed, with properties given in Table 2.6. This system was used by Perry *et al.* [17] and Tang *et al.* [32] to test GAs and DE, respectively. The equations of motion are those shown in Equation (2.13) with the damping matrix expressed as a linear combination of the mass matrix **M** and of the stiffness matrix **K** written as:

$$\mathbf{C} = a\mathbf{M} + b\mathbf{K} \tag{2.16}$$

where a and b are two constant coefficients that can be derived from

$$\xi_r = \frac{a}{2\omega_r} + \frac{b\omega_r}{2} \tag{2.17}$$

where ξ_r is the modal damping ratio associated to the r-th vibrational mode (e.g., $r = 1, 2$), and ω_r the corresponding modal natural frequency. In this example, ξ_r is set as 5% in the first and second vibration modes.

In this example, the masses, modal damping ratios (used for determining the damping coefficient parameters a and b), and stiffness values are the unknown parameters to be identified. For the known mass identification case, the parameters to be identified can be described by a set of 22 variables:

$$\boldsymbol{\theta} = \{k_1, \ldots, k_{20}, \xi_1, \xi_2\} \tag{2.18}$$

while for the unknown mass case, the system can be described by a set of 42 variables:

$$\boldsymbol{\theta} = \{m_1, \ldots, m_{20}, k_1, \ldots, k_{20}, \xi_1, \xi_2\} \tag{2.19}$$

Input forces are assumed to be Gaussian noise sequences with the RMS scaled to 1 kN, a duration of 10 seconds, and sampling time $\Delta t = 0.01$ sec; they are applied at every 5 floors of the structure, in exactly the same way described in [17, 32]. Accelerations are measured at 40% of the floors for the known mass case and at 60% of the floors for the unknown mass case as detailed in Table 2.7. The MABC parameters used in this example are: $N_{pop} = 50$ (for the known mass case), $N_{pop} = 100$ (for the unknown mass case), $N_{max} = 1000$, $N_{lim} = 200$, $m = 3$ and $\delta = 0.2$.

Table 2.6: The structural properties of a 20-DOF linear system.

Mass (kg)	Levels 1–10	4000
	Levels 11–20	3000
Stiffness (kN/m)	Levels 1-10	5000
	Levels 11–15	4000
	Levels 16–20	3500
Natural period of vibration (sec)	First mode	2.123
	Second mode	0.797

Table 2.7: Location of acceleration measurements for a 20-DOF system.

Mass cases	Floor levels
Known mass	2, 4, 7, 10, 12, 14, 17, 20
Unknown mass	1, 2, 3, 4, 6, 8, 10, 12, 14, 16, 18, 20

The identified results for the 20-DOF system are presented in Tables 2.8 and 2.9 in comparison with those obtained by GA, PSO, DE and SSRM [17, 32]. The SSRM is a search space reduction method integrated with a modified GA. Figures 2.8 and 2.9 show convergences of the mean results obtained via the MABC algorithm for the known and the unknown mass cases, respectively.

For the known mass case without noise corruption, the MABC provides much better parameter estimation than the GA and PSO, yielding much smaller errors (a maximum value of 0.76% in damping parameter estimation), while its performance is comparable to those of DE and SSRM. From the convergence graphs shown in Figure 2.8, it is observed that all the parameter estimates stabilize after 2×10^4 forward analyses. Moreover, the estimated values of damping ratios are quite accurate.

For the unknown mass case with noise corruption (5% and 10% RMS) in I/O signals, the MABC algorithm performs better than the PSO and as well as the DE and the SSRM,

Table 2.8: Results for 20-DOF known mass system without noise corruption.

Error (%)	GA [17]	SSRM [17]	PSO [32]	DE [32]	MABC
Mean–k	8.33	0.52	0.71	0.41	0.23
Max.–k	31.28	1.6	3.37	1.29	0.78
Mean–c	15.81	0.64	2.24	0.53	0.22
Max.–c	28.97	1.21	8.31	1.45	0.76

Table 2.9: Results for 20-DOF unknown mass system with noise corruption.

Error (%)	5% RMS noise corruption				10% RMS noise corruption			
	SSRM [17]	PSO [32]	DE [32]	MABC	SSRM [17]	PSO [32]	DE [32]	MABC
Mean–m	1.51	3.61	1.42	1.95	3.00	7.06	3.29	2.84
Max.–m	4.02	10.81	3.56	4.02	10.4	16.27	11.21	5.82
Mean–k	1.38	3.65	1.27	2.12	2.78	5.31	2.63	3.11
Max.–k	3.83	8.13	4.11	4.69	8.64	14.36	9.02	4.96
Mean–c	6.70	10.34	7.23	1.11	14.69	17.31	13.54	1.37
Max.–c	12.9	16.57	10.68	3.00	20.36	29.06	21.04	3.01

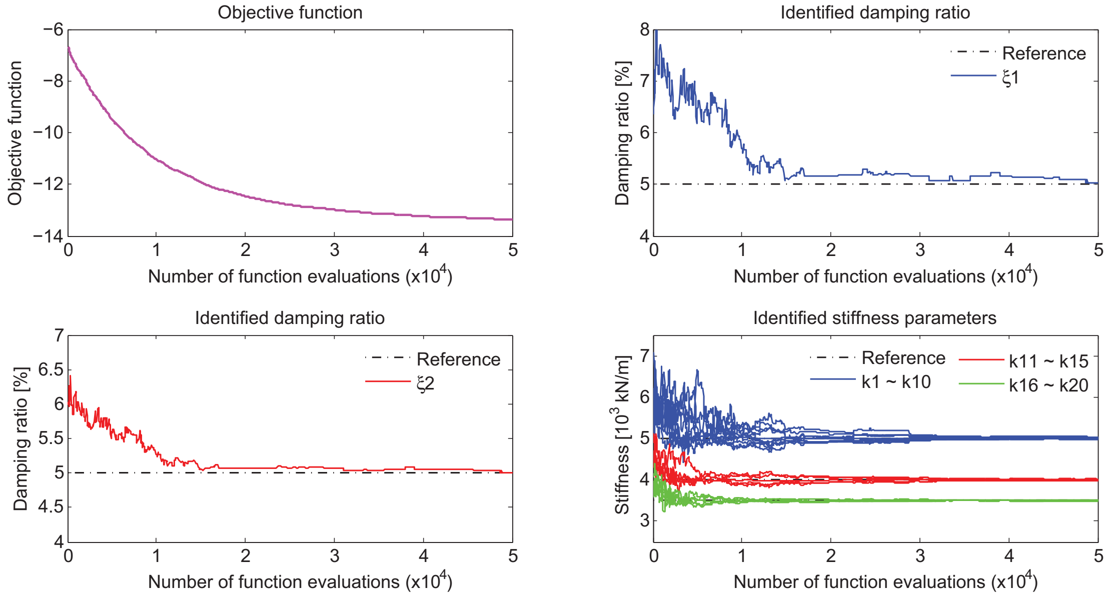


Figure 2.8: Convergence for the free noise known mass case and 40% measurements.

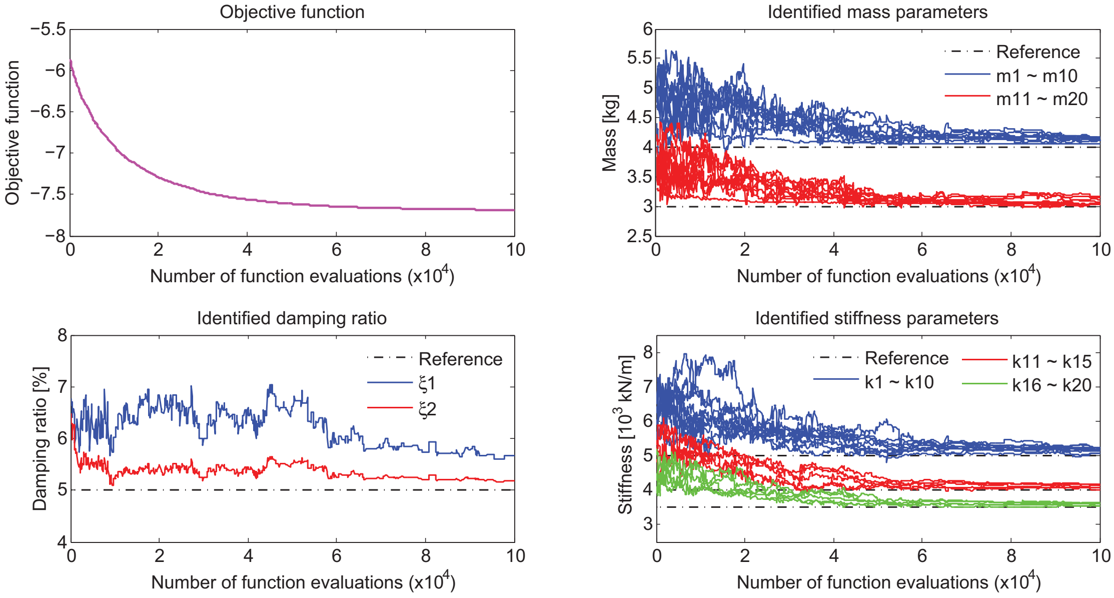


Figure 2.9: Convergence for the unknown mass case with 60% measurements and 10% noise.

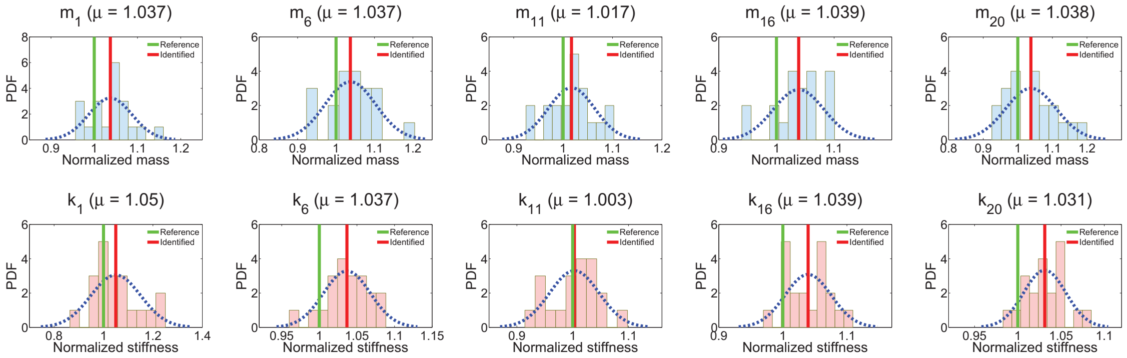


Figure 2.10: Typical PDFs of the normalized parameters obtained from 20 independent runs in the unknown mass case with 60% measurements and 10% noise corruption.

with the maximum errors of mass and stiffness obtained by MABC being slightly bigger than those obtained via the DE and the SSRM. On the other hand, the MABC algorithm seems to less sensitive to the presence of noise in the damping ratio estimation. In general, the maximum errors among all the identified parameters obtained by MABC (4.69% for 5% noise corruption and 4.96% for 10% noise corruption) are quite good when compared with those obtained by other methods.

Compared with the results discussed by Tang *et al.* [32], the proposed MABC algorithm seems to be more efficient than the DE algorithm: when applied to this 20-DOFs case, the DE algorithm uses an initial population size of 200 (equivalent $N_{max} = 1000$) while the proposed MABC algorithm runs with a much smaller initial population size (50 for the known mass case and 100 for the unknown mass case). The population size has a strong impact on the computational effort of the algorithm since it is linked to the number of fitness function evaluations which involve the time-consuming operation of solving the 2nd order ordinary differential equations of motion. For an initial population of 200, the DE algorithm will require about 2×10^6 fitness evaluations while the MABC algorithm, for initial populations of 50 (known mass case) and 100 (unknown mass case), will require about 4.5×10^4 and 9.5×10^4 function evaluations, respectively, with a substantial reduction in computational time. In terms of the CPU time used for each run by the proposed algorithm coded in MATLAB®, it took on average 18.5 min and 41.6 min, for each single simulation for the known and unknown mass case, respectively. The computational cost can be significantly decreased if the scheme is set to run in parallel on a multi-processor unit coded in more efficient computer languages (e.g., C, C++, etc.). Figure 2.10 shows several typical PDFs of the normalized mass and stiffness parameters obtained from 20 independent runs in the unknown mass case with 10% RMS noise corruption. It is concluded from this figure that the identified structural parameters are quite accurate.

2.4.3 A 2-DOF nonlinear system

To test the versatility of the modified ABC algorithm in handling nonlinear problems, a known mass 2-DOF shear type hysteretic nonlinear system, considered by Xue *et al.* [29], is employed as the third numerical example here, as shown in Figure 2.11. The equation of motion of the system can be written as:

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{r}(t) = -\mathbf{M}\mathbf{l}\ddot{u}_g(t) \tag{2.20}$$

where $\mathbf{M}$ and $\mathbf{C}$ are the 2×2 mass and damping matrices; $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ are relative displacement, velocity, and acceleration vectors to the ground, respectively; $\mathbf{l}$ is the ground motion influence coefficient vector with element 1; $\ddot{u}_g(t)$ is the earthquake acceleration; and $\mathbf{r} = (r_1 - r_2, r_2)^T$ is the vector containing the restoring forces at each floor expressed by the two following expressions.

$$\begin{aligned} \dot{r}_1 &= \left[k_1 - \alpha_1 \operatorname{sgn}(\dot{x}_1) r_1 |r_1|^{n_1 - 1} - \beta_1 |r_1|^{n_1}\right] \dot{x}_1 \\ \dot{r}_2 &= \left[k_2 - \alpha_2 \operatorname{sgn}(\dot{x}_2 - \dot{x}_1) r_2 |r_2|^{n_2 - 1} - \beta_2 |r_2|^{n_2}\right] (\dot{x}_2 - \dot{x}_1) \end{aligned} \tag{2.21}$$

This nonlinear model was clarified as the Bouc-Wen model [46]. In above two equations, $\dot{x}_1$ and $\dot{x}_2$ are the velocities of the first and second masses, respectively, k_1 and k_2 are stiffness parameters of the vertical elements, α_1, α_2, β_1, β_2, $n_1$and n_2 are nonlinear coefficient parameters, while $\operatorname{sgn}(\cdot)$ is signum function (e.g., $\operatorname{sgn}(\vartheta)$ gives -1, 0, or 1 depending on whether ϑ is negative, zero, or positive, respectively). Hence, the system parameters to be identified can be grouped in a set of 10 variables as:

$$\boldsymbol{\theta} = \{c_1, c_2, k_1, k_2, \alpha_2, \alpha_2, \beta_1, \beta_2, n_1, n_2\} \tag{2.22}$$

The values of such parameters used in the numerical analyses are shown in Table 2.10.

The ElCentro ARR4S40E (1940) earthquake acceleration data is used as input to the system with a sampling frequency of 100 Hz ($\Delta t = 0.01$ sec) for the duration of 10 seconds. The maximum peak value of the acceleration is scaled to 3g and the signal is filtered with a high frequency cutoff of 30 Hz and a low frequency cutoff of 0.1 Hz. In order to study the effects of incomplete sets of measurements, two cases are considered as shown in Figure 2.11: (1) the full output case in which a full set of measurements was available, and (2) the partial output case with only one output on the 1st DOF. Similar to the cases in [29], two noise levels 0% and 5% were considered to emphasize the effect of measurement noise on the identification of the parameters. The MABC parameters are set as follows: $N_{pop} = 30$, $N_{max} = 500$, $N_{lim} = 200$, $m = 3$ and $\delta = 0.2$. To compare the performance of the MABC algorithm for structural SI with other heuristic methodologies, the enhanced particle swarm optimization (EPSO) method proposed by Charalampakis and Dimou [28] was used: the inertia weight factor was set as 0.6 and the two acceleration factors as 1.7 suggested in [28]; the population size, maximum iteration number and number of independent runs were the same as those used in MABC.

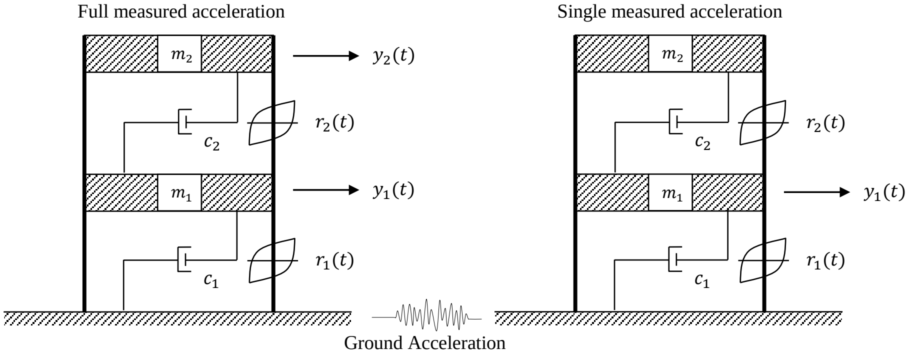


Figure 2.11: A 2-DOF nonlinear system with ground motion excitation.

Table 2.10: The structural properties of a 2-DOF nonlinear system.

Params.	Mass		Damping		Stiffness		Nonlinear parameters					
	m_1	m_2	c_1	c_2	k_1	k_2	α_1	α_2	β_1	β_2	n_1	n_2
Value	100	80	0.55	0.5	30	24	1	2	2	1	3	2

Note: the units for mass, damping and stiffness are kg, kNs/m and kN/m.

The mean values and standard deviations are summarized in Table 2.11, for the full output scenario, and in Table 2.12, for the partial output scenario. Looking at both instrumentation setups, it can be seen that the errors obtained using MABC range from 0.1% to 4.0%, in the noise free case, and from 0.1% to 6.9% in the noise polluted case, while, using EPSO, the corresponding errors range from 0.1% to 5.2%, and from 0.4% to 19.9%. The results show that the MABC approach provides almost perfect identification in the absence of measurement noise, as the EPSO. However, in the noise polluted case, the MABC algorithm performs significantly better than EPSO which appears to be more sensitive to noise corruption. From further analysis, it appears that the MABC algorithm is able to yield good identification results with small errors, even at higher noise levels. For all cases, the standard deviations obtained by the MABC are smaller than those by EPSO.

It is noteworthy that the maximum errors usually exist in the estimation of the nonlinear parameters α_i, β_i and n_i $(i = 1, 2)$, and the signal noise corruption has larger impact on the estimation of such parameters than on the estimation of damping and stiffness parameters. Figure 2.12 presents the hysteresis loops of the first floor element for the noise free and noise polluted cases in the partial output scenario. It can be seen that there is a nearly perfect

Table 2.11: Results for 2-DOF known mass system in the full output scenario.

Params.	True	0% RMS noise				5% RMS noise			
		EPSO [28]		MABC		EPSO [28]		MABC	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
c_1	0.55	0.5529 (0.5)	0.006	0.5540 (0.7)	0.006	0.5606 (1.9)	0.004	0.5522 (0.4)	0.011
c_2	0.5	0.4983 (0.3)	0.004	0.4990 (0.2)	0.004	0.4771 (4.6)	0.003	0.4969 (0.6)	0.013
k_1	30	29.969 (0.1)	0.150	30.042 (0.1)	0.175	30.111 (0.4)	0.1437	29.929 (0.2)	0.339
k_2	24	24.026 (0.1)	0.158	23.993 (0.1)	0.216	23.744 (1.1)	0.0856	24.115 (0.5)	0.431
α_1	1	0.9533 (4.7)	0.133	0.9841 (1.6)	0.093	0.9710 (2.9)	0.1409	0.9775 (2.3)	0.194
α_2	2	2.0576 (2.9)	0.142	2.0098 (0.5)	0.146	2.1955 (9.8)	0.0698	1.9672 (1.6)	0.315
β_1	2	1.9451 (2.7)	0.226	2.0407 (2.0)	0.190	2.0607 (3.1)	0.2853	1.9754 (1.2)	0.390
β_2	1	1.0375 (3.7)	0.143	0.9782 (2.2)	0.104	0.8470 (15.3)	0.1062	1.0611 (6.1)	0.382
n_1	3	3.0609 (2.0)	0.174	3.0019 (0.1)	0.124	3.0248 (0.8)	0.1925	3.0433 (1.4)	0.295
n_2	2	2.0260 (1.3)	0.134	2.0455 (2.3)	0.081	2.3338 (13.3)	0.0869	2.0184 (0.9)	0.281
Max error (%)		4.7		2.3		15.3		6.1	

Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

Table 2.12: Results for 2-DOF known mass system in the partial output scenario.

Params.	TRUE	0% RMS noise				5% RMS noise			
		EPSO [28]		MABC		EPSO [28]		MABC	
		Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
c_1	0.55	0.5546 (0.8)	0.004	0.5529 (0.5)	0.004	0.5383 (2.1)	0.0078	0.5534 (0.6)	0.013
c_2	0.5	0.4973 (0.5)	0.003	0.5023 (0.5)	0.002	0.4971 (0.6)	0.0071	0.5002 (0.1)	0.019
k_1	30	30.063 (0.2)	0.105	29.992 (0.1)	0.111	29.437 (1.9)	0.0879	30.054 (0.2)	0.343
k_2	24	23.926 (0.3)	0.121	24.035 (0.1)	0.072	24.624 (2.6)	0.0277	24.032 (0.1)	0.352
α_1	1	1.0024 (0.2)	0.085	0.9664 (3.4)	0.086	0.9543 (4.6)	0.1026	1.0245 (2.4)	0.187
α_2	2	2.0963 (4.8)	0.110	1.9189 (4.0)	0.131	1.6938 (15.3)	0.2295	2.0512 (2.6)	0.657
β_1	2	2.0749 (3.7)	0.347	1.9732 (1.3)	0.187	1.7163 (14.2)	0.1802	2.0716 (3.6)	0.404
β_2	1	0.9481 (5.2)	0.140	1.0345 (3.4)	0.074	0.9679 (3.3)	0.1393	0.9332 (6.7)	0.362
n_1	3	2.9710 (1.0)	0.101	3.038 (1.3)	0.124	3.1661 (5.5)	0.1392	2.9838 (0.5)	0.243
n_2	2	2.1009 (5.0)	0.183	2.0180 (0.9)	0.112	2.3982 (19.9)	0.226	2.0468 (2.3)	0.394
Max error (%)		5.2		4.0		19.9		6.7	

Note: The values in parentheses are absolute relative errors in %. Std. represents standard deviation.

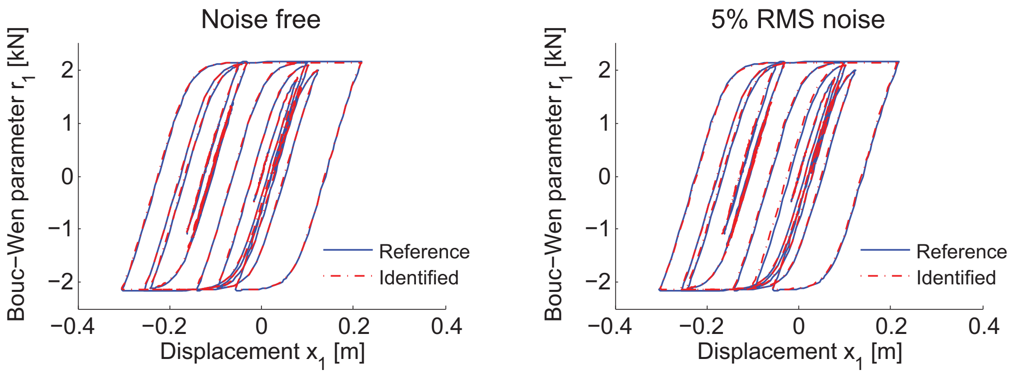
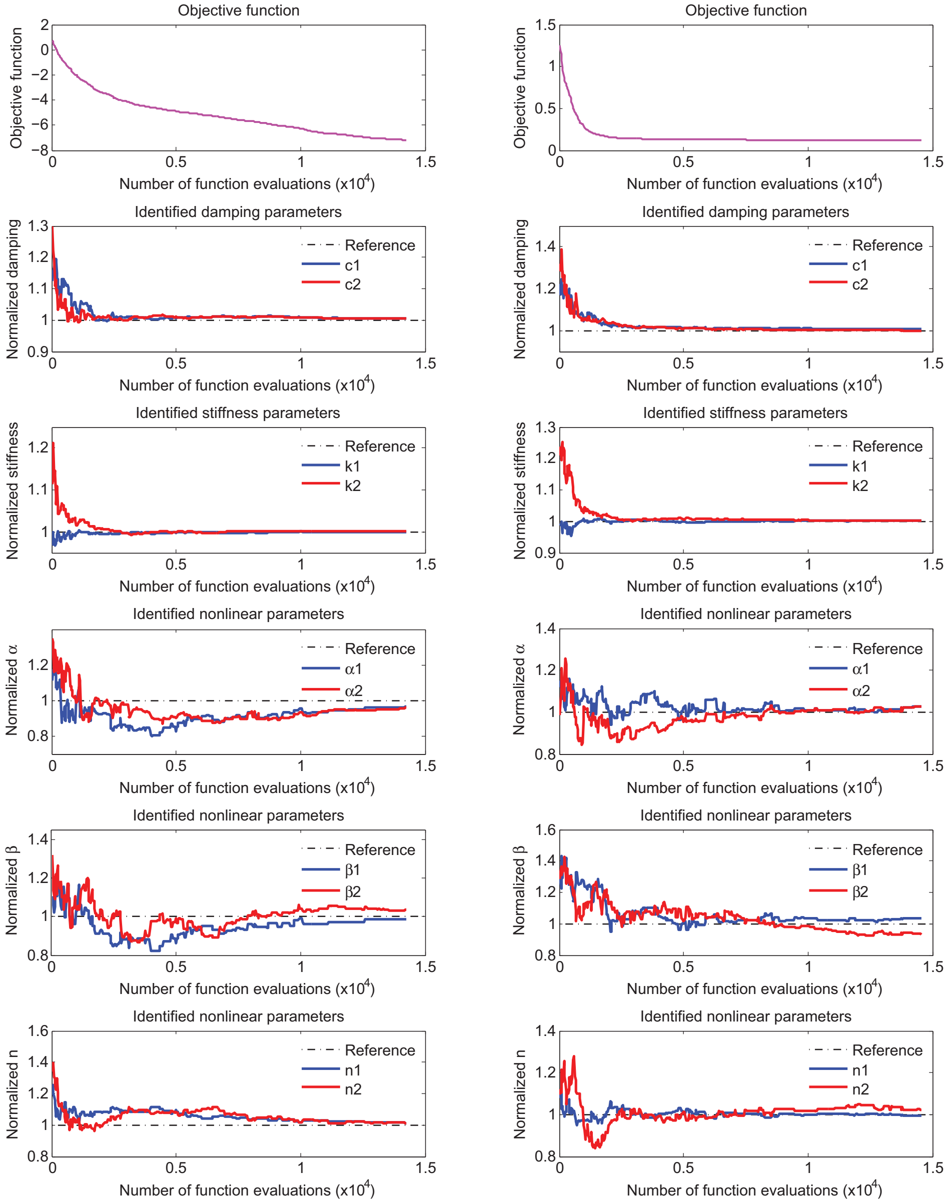


Figure 2.12: Hysteresis loops of the first DOF with partial measurements.

agreement between the simulated and MABC estimated results in the phase-plane plots of the Bouc-Wen parameter r_1 *vs.* the displacement x_1, even with partial outputs and at a high noise corruption level. Figure 2.13 shows a comparison of the identified normalized parameters for both the noise free and the noise corrupted measurements in the partial output case. It is observed that the estimation of parameters rapidly converges and is able to reach good results after about 8×10^3 forward function evaluations.

Figure 2.14 illustrates the PDFs of the normalized parameters obtained from 20 independent runs in the partial output scenario with 5% RMS noise corruption. Similar to the first two examples, the PDFs approximately follow normal distribution. The mean values of these normalized parameters are quite close to 1 accounting for good parameter estimates.



(a) Noise free

(b) 5% RMS noise in I/O signals

Figure 2.13: MABC convergence for the case of partial measurements.

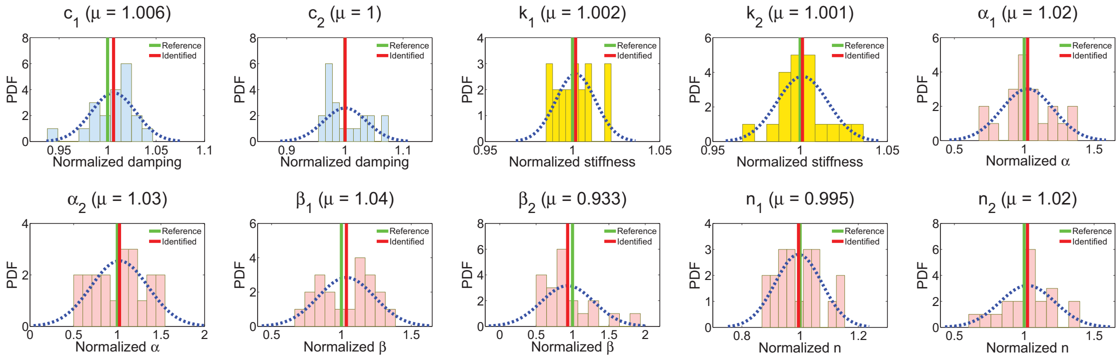


Figure 2.14: Typical PDFs of the normalized parameters obtained from 20 independent runs with 10% RMS noise corruption.

2.4.4 Observations of the numerical examples

The simulation results previously discussed lead to the following observations:

(i) From the simulation results, it can be concluded that the identification of unknown mass system is much more challenging than that of known mass system, because the unknown mass system identification is always a high multimodal problem with complex nonlinearities and convexities while the known mass system has a reduction of the search space.

(ii) The MABC algorithm has proven to be very efficient for structural parameter identification: it converges relatively quickly, it is robust, and yields small deviations of parameter estimation in each independent run.

(iii) All the algorithms considered in this study (MABC, GA, EPSO, etc.) yield good estimation results with small errors for noise free cases in full output scenarios. The MABC algorithm, however, performs more satisfactorily than others when dealing with noisy I/O signals, unknown masses and partial output information.

(iv) The parameters obtained by the MABC algorithm converge quite rapidly the correct solutions for known mass cases. For complex unknown mass systems, larger population sizes or more iteration cycles can be employed in order to find finer solutions.

(v) The MABC algorithm seems to be relatively insensitive to noise. It could provide accurate parameter estimations even for complex systems with partial measurements at high noise corruption levels (i.e. 10% RMS noise in I/O signals).

(vi) The maximum error mostly occurs in the identification of mass and damping parameters and/or nonlinear parameters. This phenomenon is enhanced if the measured signals are corrupted by noise.

2.5 Conclusions

In this chapter, a modified artificial bee colony (MABC) algorithm has been presented in the context of structural system identification. This algorithm is a population based stochastic algorithm which requires only few common control parameters. It has a simple structure, it is easy to implement, it is robust, and it provides results with high accuracy. The application of proposed nonlinear factor for convergence control enhances the balance of global and local searches. The idea behind the nonlinear factor is to free the method, at its initial steps, to search over the entire search space and to allow for faster convergence during the local search phase which takes place towards the end of the iteration process. In this way, the search space is gradually reduced following a nonlinear trend. To avoid stagnation around local minima, a "scout bee" search will be carried out if needed. Thus, the MABC algorithm is supposed to be able to locate finer solutions.

To investigate the applicability of this proposed technique to structural system identification, two linear systems (5-DOF and 20-DOF) and a nonlinear system (2-DOF) have been studied under different conditions, addressing issues such as the number of measurements used in identification, noise in the signals, and the knowledge of the structural mass. In all cases considered, the simulation results show that the proposed MABC algorithm can produce excellent parameter estimation with small errors. The presented method is effective, robust and efficient even with reduced partial measurements at high noise pollution levels.

Chapter 3

A Hybrid Parallel ABC algorithm with Search Space Reductions for Output-Only Identification with Unmeasured Input Excitations

In this chapter, a hybrid heuristic optimization strategy in a parallel scheme is presented to simultaneously identify structural parameters and dynamic input forces from incomplete sets of output-only measurements. A modified Newmark integration method is proposed to predict the unmeasured force time-history. This chapter is reproduced from the paper co-authored with Professor Raimondo Betti, which was published in the Journal of Structural Control and Health Monitoring [47].

3.1 Introduction

The majority of the study on identification of structural systems in the time domain mainly focuses on estimating physical parameters, such as mass, stiffness and damping coefficients, using both the measurement of the input excitation as well as the structural response. However, in practice, the time-history of dynamic input forces, e.g. wind, traffic, etc., are often difficult to be directly measured: while the points of application of such forces could be, more or less, accurately predicted, e.g. the lower-chord nodes in a truss bridge for traffic loads or the upwind facade of a building for wind loads, the magnitudes of such forces cannot be easily measured and is usually inferred from the measurements of system response. For this reason, dealing with unmeasured input excitations has made the identification of the structural parameters a much more challenging task, with the analysis relying only on the information contained in the structural response (output-only) measurements and on assumptions linked to the experience-based knowledge of the model (e.g., structural mass is known) and the excitations (e.g., input locations are known).

It then becomes natural (and attractive) to look at the possibility of identifying the input force along with the physical parameters just by handling the measurements of structural response. In the past, input force identification has been carried out under the assumption that the system is "completely known", e.g. all the system's parameters are known. Turco [48] proposed a strategy to identify/reconstruct the excitation forces starting from the measurement of the stress or strain history in a suitable number of points of a known structure. Lu and Law [49] presented a sensitivity-based method for identifying input excitations approximated by Fourier series using the structural dynamic response. Yet, most of system's parameters cannot be assumed known a priori (e.g., degradation and deterioration might exist in a structure) but need to be identified.

In recent years, there have been attempts to simultaneously identify the system's parameters and the input force within a time-domain framework using techniques such as the Iterative Least-Squares Estimation (ILSE) and Kalman filter based techniques. Wang and Haldar [50] proposed a time domain Recursive Least-Squares (RLS) technique to estimate the stiffness and damping parameters, at the element level, of a shear type structure excited by unknown input forces. In a later work, they further extended this approach by combining it with the EKF and with the Weighted Global Iteration (WGI) method for the case of a limited number of time-history measurements [51]. Similar studies on local level structural health assessment with unknown input excitation using ILSE and/or RLS based techniques can be also found in [52, 53]. Moreover, Chen and Li [54] presented a modified ILSE procedure for simultaneous identification of system parameters and input excitation from output-only measurements imposing that the input force vector could be modified and updated iteratively together with the system's parameters. Yang *et al.* [20]

developed an adaptive tracking technique based on the EKF, called the AEKF-UI, to identify the structural parameters of linear and nonlinear systems and to track their changes without any information about the input excitation. Similarly, Xu *et al.* [55] showed that a weighted adaptive version of the ILSE could provide a successful identification of the structural parameters and of the dynamic input loading.

In general, most of the above methods can yield good estimates of both system's parameters and/or input forces when the structural response (accelerations, velocities and displacements) time histories are available at every single discrete point (degree-of-freedom (DOF)) of the structural model. However, this is a constraint that is not usually satisfied in real-life applications (see [51]).

Only recently, the issue related to the availability of a limited number of structural response components, e.g. accelerations, has been considered within the identification of the input excitation. In this framework, the value of the unknown force at a given time step is computed from the measured acceleration at such a given time step and from the other components of the response (e.g. velocity and/or displacement) computed at previous time steps. Perry and Koh [17], for example, used this procedure with a modified GA algorithm called the Search Space Reduction Method (SSRM) [41] for output-only structural identification, while Law and Yong [56] and Trinh and Koh [57] used such a combination of measured and computed components of the response in sub-structural identification methods for large structural models.

In this chapter, a hybrid version of the ABC algorithm (denoted with HABC) has been proposed for the identification of the parameters of a structural model and, when possible, of the input excitation considering the limitations that can occur in real life applications, such as only one type of measurements (either accelerations, or velocities or displacements), incomplete sets of measurements due to a limited number of sensors and noisy data. This hybrid algorithm combines MABC, proposed in Chapter 2, with a local search operator like the Nelder-Mead simplex method (NMSM) and a Search Space Reduction (SSR) strategy to improve the rate of convergence and accuracy of the algorithm.

The chapter is organized as follows. In Section 3.2, the procedure for the input force identification is discussed. Section 3.3 presents the proposed HABC-SSR based parallel computational identification methodology. In section 3.4, numerical simulations and identification results obtained from three examples are presented and discussed in details. Section 3.5 contains the final conclusions of this chapter.

3.2 Unknown input force identification based on a modified Newmark integration

In the case of output-only system identification, a parametric system's model, in the time domain, requires the estimation of input (denoted with $\hat{\mathbf{u}}(t_p)$, $p = 1, 2, \ldots, N$) so as to simulate the estimated/predicted system response (denoted with $\hat{\mathbf{y}}(t_p)$), namely,

$$\hat{\mathbf{y}}(t_p) = f\left(\hat{\mathbf{u}}(t_p), \hat{\boldsymbol{\theta}}\right) \tag{3.1}$$

where $\hat{\boldsymbol{\theta}}$ represents the parametric vector as discussed in Chapter 2. In the case when the input time-history is known, then the estimated input is directly set equal to the measured input, namely, $\hat{\mathbf{u}}(t_p) = \mathbf{u}(t_p)$. If, instead, the input is unknown (only measurements of the structural response (output-only) are available), then the input time-history $\hat{\mathbf{u}}(t_p)$ becomes an additional "parameter" to be identified along with the system parameters.

In some cases, for example when some of the structural parameters, e.g. the masses, are known, it is possible to identify the time-histories of the external excitation at the same time as the remaining structural parameters by using only the measurements of the structural response, e.g. the acceleration time-histories. Here it is assumed that there is only one force or one set of forces that need to be identified. This force or set of forces will be labelled in this chapter as "primary force". Although, in real life applications, a structure is subjected to a variety of environmental and man-made actions that occur simultaneously (e.g. traffic and wind on a bridge, or wind and dynamic shaker on a building), it is reasonable to assume that the effects induced by the action of interest (primary force) are predominant over those induced by the other forces (that will then be called "secondary forces"): it is then possible to consider that the recorded structural response be induced only by one force or by a specific set of forces, neglecting the contributions of the secondary forces. However, in one of the numerical examples in Section 3.4, we will show the effects of including the secondary effects generated by low amplitude additional excitations on the identification of the primary input force and of the structural parameters.

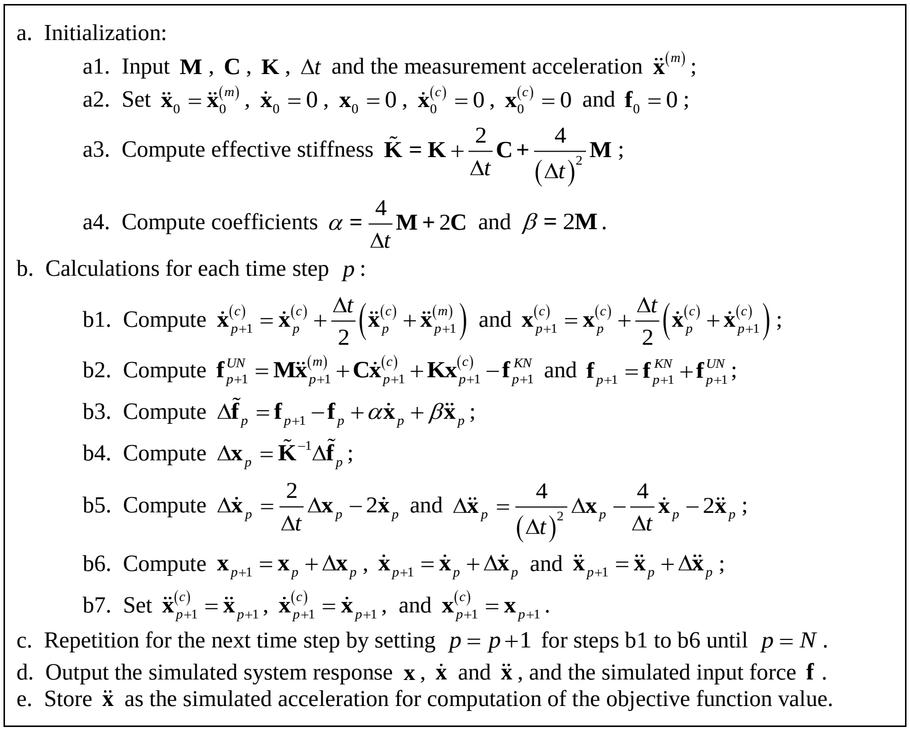
Within an iterative optimization process, at each iteration the simulated structural response and the corresponding input forces can be computed simultaneously using the current estimated parameters. In this study, a modified Newmark integration scheme is proposed for the identification of the input force by using the time-histories of the structural accelerations, following a numerical procedure first proposed by Perry and Koh [17] and further employed in the sub-structural identification of large structural systems [57]. Before introducing such a procedure, the following assumptions are made:

(i) In order to avoid non-uniqueness of the solution in the simultaneous identification of structural parameters and dynamic input, the mass matrix or the mass distribution of the structure are assumed known a priori. This is an acceptable assumption since, in real life applications, sufficiently accurate estimates of the mass matrix can be obtained from the structural drawings looking at the structural geometry, type of materials, etc. [10].

(ii) The initial conditions of the structure are assumed to be known: for instance, it is reasonable to assume that, before the application of the input force that need to be identified (primary input force), the structure be at rest. This assumption is commonly used in all applications involving seismic analysis of structures, where the effects of secondary excitations like wind and traffic are usually neglected.

(iii) The points of applications of the unknown dynamic input forces are assumed to be known. For example, the locations of the dynamic shakers on a building or a bridge, in the case of a forced excitation test, are known. In this case, the locations of the primary input forces are known but the magnitude can be estimated through the identification process. At the other locations, the input forces are assumed either to be known or zero.

(iv) Sensors are assumed available at the DOFs where the unknown input forces are applied and at the adjacent DOFs. Sensor placements should be such to prevent non-unique solutions for the identification process [58].



```
a. Initialization:
    a1. Input $\mathbf{M}$, $\mathbf{C}$, $\mathbf{K}$, $\Delta t$ and the measurement acceleration $\ddot{\mathbf{x}}^{(m)}$;
    a2. Set $\ddot{\mathbf{x}}_0 = \ddot{\mathbf{x}}_0^{(m)}$, $\dot{\mathbf{x}}_0 = 0$, $\mathbf{x}_0 = 0$, $\dot{\mathbf{x}}_0^{(c)} = 0$, $\mathbf{x}_0^{(c)} = 0$ and $\mathbf{f}_0 = 0$;
    a3. Compute effective stiffness $\tilde{\mathbf{K}} = \mathbf{K} + \frac{2}{\Delta t}\mathbf{C} + \frac{4}{(\Delta t)^2}\mathbf{M}$;
    a4. Compute coefficients $\alpha = \frac{4}{\Delta t}\mathbf{M} + 2\mathbf{C}$ and $\beta = 2\mathbf{M}$.
b. Calculations for each time step $p$:
    b1. Compute $\dot{\mathbf{x}}_{p+1}^{(c)} = \dot{\mathbf{x}}_p^{(c)} + \frac{\Delta t}{2}\left(\ddot{\mathbf{x}}_p^{(c)} + \ddot{\mathbf{x}}_{p+1}^{(m)}\right)$ and $\mathbf{x}_{p+1}^{(c)} = \mathbf{x}_p^{(c)} + \frac{\Delta t}{2}\left(\dot{\mathbf{x}}_p^{(c)} + \dot{\mathbf{x}}_{p+1}^{(c)}\right)$;
    b2. Compute $\mathbf{f}_{p+1}^{UN} = \mathbf{M}\ddot{\mathbf{x}}_{p+1}^{(m)} + \mathbf{C}\dot{\mathbf{x}}_{p+1}^{(c)} + \mathbf{K}\mathbf{x}_{p+1}^{(c)} - \mathbf{f}_{p+1}^{KN}$ and $\mathbf{f}_{p+1} = \mathbf{f}_{p+1}^{KN} + \mathbf{f}_{p+1}^{UN}$;
    b3. Compute $\Delta\tilde{\mathbf{f}}_p = \mathbf{f}_{p+1} - \mathbf{f}_p + \alpha\dot{\mathbf{x}}_p + \beta\ddot{\mathbf{x}}_p$;
    b4. Compute $\Delta\mathbf{x}_p = \tilde{\mathbf{K}}^{-1}\Delta\tilde{\mathbf{f}}_p$;
    b5. Compute $\Delta\dot{\mathbf{x}}_p = \frac{2}{\Delta t}\Delta\mathbf{x}_p - 2\dot{\mathbf{x}}_p$ and $\Delta\ddot{\mathbf{x}}_p = \frac{4}{(\Delta t)^2}\Delta\mathbf{x}_p - \frac{4}{\Delta t}\dot{\mathbf{x}}_p - 2\ddot{\mathbf{x}}_p$;
    b6. Compute $\mathbf{x}_{p+1} = \mathbf{x}_p + \Delta\mathbf{x}_p$, $\dot{\mathbf{x}}_{p+1} = \dot{\mathbf{x}}_p + \Delta\dot{\mathbf{x}}_p$ and $\ddot{\mathbf{x}}_{p+1} = \ddot{\mathbf{x}}_p + \Delta\ddot{\mathbf{x}}_p$;
    b7. Set $\ddot{\mathbf{x}}_{p+1}^{(c)} = \ddot{\mathbf{x}}_{p+1}$, $\dot{\mathbf{x}}_{p+1}^{(c)} = \dot{\mathbf{x}}_{p+1}$, and $\mathbf{x}_{p+1}^{(c)} = \mathbf{x}_{p+1}$.
c. Repetition for the next time step by setting $p = p+1$ for steps b1 to b6 until $p = N$.
d. Output the simulated system response $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$, and the simulated input force $\mathbf{f}$.
e. Store $\ddot{\mathbf{x}}$ as the simulated acceleration for computation of the objective function value.
```

Figure 3.1: Structural system response simulation and input force identification procedure.

Based on the above assumption, the procedure to identify the unknown input force is presented in Figure 3.1 and can be described as follows. At a given k-th iteration of the iterative optimization algorithm, a mass matrix (assumed to be known) and an estimated damping and stiffness matrix are available, based on the estimated set of parameters at that iteration. With this set of matrices, the effective stiffness matrix $\tilde{\mathbf{K}}$ (step a.3, Figure 3.1) and the coefficients α and β (step a.4, Figure 3.1) can be obtained. At this point, for each time step p, the velocity and displacement vectors at time step $p+1$ can be computed from the measured acceleration vector at the same time step and the values of the computed displacements and velocities at a previous time step, namely:

$$\dot{\mathbf{x}}_{p+1}^{(c)} = \dot{\mathbf{x}}_p^{(c)} + \frac{\Delta t}{2}\left(\ddot{\mathbf{x}}_p^{(c)} + \ddot{\mathbf{x}}_{p+1}^{(m)}\right) \tag{3.2}$$

$$\mathbf{x}_{p+1}^{(c)} = \mathbf{x}_p^{(c)} + \frac{\Delta t}{2}\left(\dot{\mathbf{x}}_p^{(c)} + \dot{\mathbf{x}}_{p+1}^{(c)}\right) \tag{3.3}$$

where $\mathbf{x}$, $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ are the system's response components representing displacement, velocity, and acceleration vectors, respectively. Δt denotes the sampling time. The superscript '(m)' represents the measured quantities and the superscript '(c)' denotes the computed or simulated quantities (step b.3, Figure 3.1). The subscript p indicates a quantity corresponding to time t_p. Since the structure may be excited by a combination of both known and unknown input forces, the dynamic force vector can be written as:

$$\mathbf{f}_p = \mathbf{f}_p^{KN} + \mathbf{f}_p^{UK} \tag{3.4}$$

where $\mathbf{f}^{KN}$ represents the known force vector while $\mathbf{f}^{UK}$ is the unknown force vector obtained as shown in step b.2, Figure 3.1.

Once the unknown component of the force vector has been computed, it can then be used together with the known component to obtain the updated estimated values of the accelerations, velocities and displacements, according to the modified Newmark integration scheme (step b.4-b.7, Figure 3.1). These estimated accelerations are then used to compute the objective function value in Equation (2.3).

It is noteworthy that, in this chapter, we consider linear structural systems with the equation of motion given in Equation (2.13). The energy dissipation mechanism is described by the Rayleigh damping model as shown in Equations (2.16) and (2.17).

3.3 HABC-SSR-based parallel identification methodology

In this section, a hybrid strategy is proposed for the simultaneous identification of structural system's parameters and, when possible, of the dynamic input time histories. The proposed strategy combines the modified ABC (MABC) algorithm, proposed in Chapter 2, with a Local Search (LS) operator, the Nelder-Mead Simplex Method (NMSM), integrated within a Search Space Reduction (SSR) strategy so to improve the convergence as well as to increase the identification accuracy and reliability. Since each run of the proposed algorithm is independent from the other runs, a parallel computing scheme can be implemented to improve the computational efficiency.

3.3.1 The hybrid artificial bee colony (HABC) algorithm

A hybrid optimization strategy is proposed here, which is comprised of a heuristic global search optimizer and of an LS operator. The global search optimizer, namely a heuristic algorithm (e.g., the MABC in Chapter 2), is able to overcome the local optima stagnation problem while the LS operator is useful for fine solution tuning near the global optimum. In the current chapter, the Nelder-Mead simplex method (NMSM), a non-gradient-based LS operator, is adopted in addition to the global optimizer of MABC leading to a hybrid algorithm, namely the HABC for short.

3.3.1.1 Overview of the Nelder-Mead simplex method.

The Nelder-Mead simplex method (NMSM), as a local descent algorithm, was first proposed by Nelder and Mead [59] and then further studied by Lagarias *et al.* [60]. It is characterized by fast convergence in search for a local minimum and is applicable for unconstrained multi-dimensional optimization problems without using function gradient information. This method converges rapidly to a minimum point of the cost function by generating a simplex and using this simplex to search for the most promising directions. A simplex is defined as a geometrical figure which consists of $N_\theta + 1$ vertices (or points), where N_θ is the number of parameters (or variables) to be optimized. The simplex is updated according to the procedures of *reflection*, *expansion*, *contraction* and *shrinkage*, each of which is characterized by some scalar parameters, namely the coefficients of *reflection* (γ), *expansion* (η), *contraction* (χ) and *shrinkage* (δ). These parameters are selected by the user and must satisfy the conditions: $\gamma > 0$, $\eta > 1$, $\eta > \gamma$, $0 < \chi < 1$, and $0 < \delta < 1$. In this chapter, we select the scalar coefficients as $\gamma = 1$, $\eta = 2$, $\chi = 0.5$ and $\delta = 0.5$, as suggested in a standard NMSM algorithm. The detailed steps of the NMSM algorithm can be found in References [59,60].

3.3.1.2 The HABC scheme.

In developing an enhanced/hybrid strategy with an LS operator, one has to consider the following factors: (1) where to add the LS operator, (2) when to start the LS operator, (3) which individual solution among the population should be used to start the LS operator, and (4) the maximum number of iterations for the LS operator. In this proposed HABC strategy, the NMSM is introduced within the MABC right after the Onlooker Phase Search and is implemented every N_{ls} iterations, using the current best individual $\hat{\boldsymbol{\theta}}_{\text{best}}$ from MABC as the initial guess $\boldsymbol{\Theta}_1^{\text{ls}}$ for the local search. A simplex around the initial guess $\boldsymbol{\Theta}_1^{\text{ls}}$ is first generated developing N_θ variations of $\boldsymbol{\Theta}_1^{\text{ls}}$ (e.g., $\boldsymbol{\Theta}_2^{\text{ls}}$, $\boldsymbol{\Theta}_3^{\text{ls}}$, $\cdots$, $\boldsymbol{\Theta}_{N_\theta+1}^{\text{ls}}$) by slightly altering each single component Θ_{1j}^{ls} to $\boldsymbol{\Theta}_1^{\text{ls}}$, where $j = 1, 2, \cdots, N_\theta$. In this study, a 3% increase on each component of $\boldsymbol{\Theta}_1^{\text{ls}}$ was used to create the other N_θ additional parameter sets. These N_θ vectors, together with $\boldsymbol{\Theta}_1^{\text{ls}}$, are then used as vertices of the simplex. Once the simplex has been formed, the NMSM is carried out following the procedure introduced in [59, 60] until the termination criterion is reached (namely, for example, either the iteration number reaches N_{lsiter} or the objective function value achieves a user-defined tolerance ε). At the end of the NMSM, the updated parameter vector $\boldsymbol{\Theta}_{\text{best}}^{\text{ls}}$ is transferred back to the MABC replacing the initial individual $\hat{\boldsymbol{\theta}}_{\text{best}}$, and then the global search resumes.

It is noteworthy that a penalty strategy is implemented here to renew the objective function when NMSM is used for local search. If a candidate parameter set is outside the feasible search space (restricted by the parameter bounds), then an exaggerated objective function value is returned. Since this value is uncommonly large in comparison to other objective function values, the corresponding parameter values can be easily eliminated and replaced by other optimal values.

For a single independent run, the flaw chart of the proposed HABC algorithm is illustrated in Figure 3.2.

3.3.2 Search Space Reduction strategy for the HABC algorithm

The advantage of working with heuristic algorithms such as the proposed HABC algorithm is that it is possible to perform simultaneously many independent runs, each of which will lead to an optimal solution. The hope is that these solutions are quite close but it might happen that few of those might end up being stack in local minima. At this point, it is convenient to use this set of optimal solutions to redefine a new search space and re-perform the optimization procedure using this new space as the initial search space. This strategy can be labeled as *Search Space Reduction* (SSR).

The idea of SSR, which aims to increase the efficiency and possibly the accuracy of

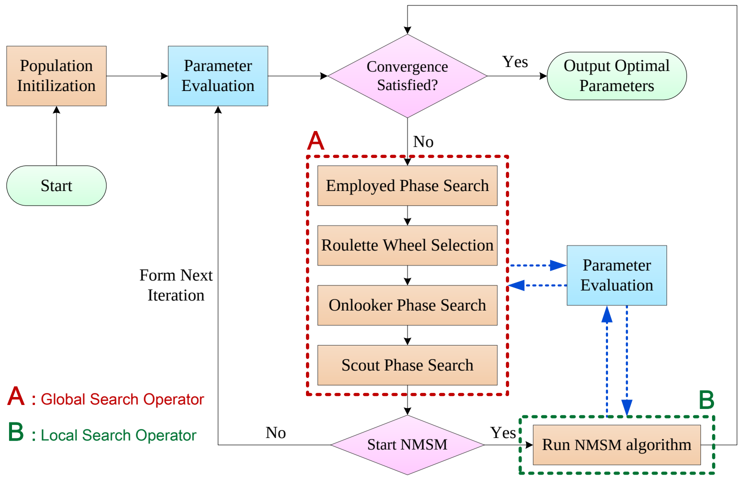


Figure 3.2: Flow chart of the proposed HABC algorithm for a single independent run.

identification algorithms by reducing (shrinking) the large search space, has been successfully used in GA and hybrid evolutionary algorithm in the previous studies by Perry *et al.* [41] and Charalampakis and Koumousis [61], showing a substantial improvement in the final results when dealing with an initial large parameter search space. Hence, it appears natural to adopt such a strategy within the proposed version of HABC. Here, this strategy is implemented by carrying out several independent runs of the HABC, following which the weighted mean and weighted standard deviation values of the identified parameters are computed so to determine the new parameter bounds.

Initially, the HABC is performed for a few (IR^{ssr}) independent runs evaluating the values of the objective function and of the corresponding fitness function of the solutions. All these independent runs will use the same initial search space and can be performed either sequentially or in parallel, if multiple processors are available. The end result will be a set of IR^{ssr} solutions which can be sorted out and the worst IR^{ssr}_w ones are eliminated. At this point, the remaining $IR^{ssr} - IR^{ssr}_w$ (denoted as $IR^{(0)}$) solutions can be grouped in a set $\mathbf{\Omega}$ as follows:

$$\mathbf{\Omega} = \{\hat{\boldsymbol{\theta}}_q \mid q = 1, 2, \ldots, IR^{(0)}\} \tag{3.5}$$

where each solution has a corresponding weighting coefficient defined as:

$$w_q = \frac{fit(\hat{\boldsymbol{\theta}}_q)}{\max\{fit(\hat{\boldsymbol{\theta}}_q)\}_{q=1,2,\ldots,IR^{(0)}}} \tag{3.6}$$

where $fit(\cdot)$ is the fitness function used in HABC to rank the solution. In Equation (3.6), it can be seen that the "best" identified solution will have a weight of unity. For this set of solutions, it is possible to calculate the weighted mean value of the j-th parameter ($j = 1, 2, \ldots, N_\theta$) as follows:

$$\bar{\theta}_j = \left(\sum_{q=1}^{IR^{(0)}} w_q \hat{\theta}_{qj} \right) \Bigg/ \left(\sum_{q=1}^{IR^{(0)}} w_q \right) \tag{3.7}$$

while the corresponding weighted standard deviation is expressed as:

$$\sigma_j = \sqrt{ \left(\sum_{q=1}^{IR^{(0)}} w_q \big(\hat{\theta}_{qj} - \bar{\theta}_j \big)^2 \right) \Bigg/ \left(\sum_{q=1}^{IR^{(0)}} w_q \right) } \tag{3.8}$$

Afterwards, the new trial upper bound and lower bound of the search space of the j-th parameter can be defined symmetrically around the weighted mean value, so that:

$$\begin{aligned} \tilde{\theta}_j^{\max} &= \bar{\theta}_j + \lambda \sigma_j \\ \tilde{\theta}_j^{\min} &= \bar{\theta}_j - \lambda \sigma_j \end{aligned} \tag{3.9}$$

where λ, positive integer, is defined as the window width coefficient of the new trial range. However, the new trial bounds might exceed the initial boundary values and this would be in contrast with the purpose of the SSR. To overcome this problem, it is convenient to define the final new search space of the j-th parameter as the intersection of the initial search space and the new trial search space, namely:

$$\left[\check{\theta}_j^{\min}, \check{\theta}_j^{\max} \right]^{\text{new}} = \left[\theta_j^{\min}, \theta_j^{\max} \right]^{\text{initial}} \cap \left[\tilde{\theta}_j^{\min}, \tilde{\theta}_j^{\max} \right]^{\text{trial}} \tag{3.10}$$

Within this newly defined search space, the HABC will resume and new optimal solutions will be reached.

In implementing this SSR strategy, it is important to make the following considerations:

- λ should not be a very small number in order to obtain a reasonably feasible search space (avoiding stagnation of the optimization process in a small region).
- $IR^{(0)}$ should be large enough so to obtain statistically meaningful results for the weighted mean and standard deviation $\bar{\theta}_j$ and σ_j.

To this end, Figure 3.3 and Figure 3.4 present a schematic representation of the proposed HABC algorithm with the associated SSR strategy for the case of a serial (Figure 3.3) or parallel (Figure 3.4) scheme.

3.3.3 The HABC-SSR based identification scheme

While the architecture of the proposed HABC-SSR strategy is relatively simple and straightforward, the implementation requires a set of parameters/indicators that need to be defined by the user. These parameters/indicators are:

- $\boldsymbol{\theta}^{\min}$ and $\boldsymbol{\theta}^{\max}$: the lower and the upper bound of the initial parameter search space;
- N_θ: the number of parameters to be identified;
- N_{pop}: the population size;
- N_{max}: the maximum number of iterations for HABC;
- N_{lim}: the number of limit iterations;
- m: the power exponent representing the nonlinear decreasing rate;
- δ: a parameter controlling the termination point of the nonlinear decreasing behavior;
- N_{ls}: the starting iteration for the beginning of the local search NMSM;
- N_{lsiter}: the maximum number of iterations in NMSM;
- IR: the total number of independent HABC runs;
- IR^{ssr}: the number of independent runs before SSR is applied;
- IR^{ssr}_w: the number of solutions to be eliminated before SSR is applied;
- λ: the window width coefficient for the new search space bound in SSR.

The values of such parameters are very much problem dependent, e.g. the number of parameters to be identified and their bounds, and are strongly related to the computational capabilities available, e.g. multi-processor machine.

3.4 Numerical applications

In order to investigate the applicability of the proposed technique to the identification of both structural parameters and dynamic input time histories, three structural systems, a 5-DOF system, a 15-DOF system and three connected frames, have been considered in this

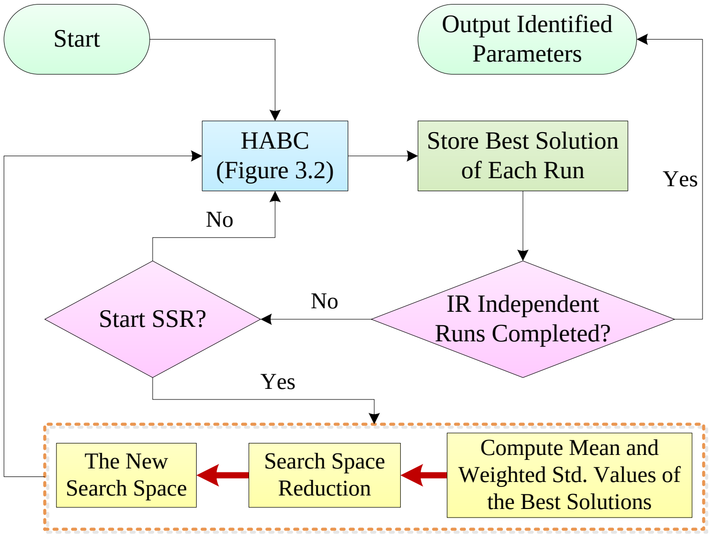

Start
Output Identified Parameters
HABC (Figure 3.2)
Store Best Solution of Each Run
Yes
No
Start SSR?
No
IR Independent Runs Completed?
Yes
The New Search Space
Search Space Reduction
Compute Mean and Weighted Std. Values of the Best Solutions

Figure 3.3: The HABC-SSR based serial computational identification scheme.

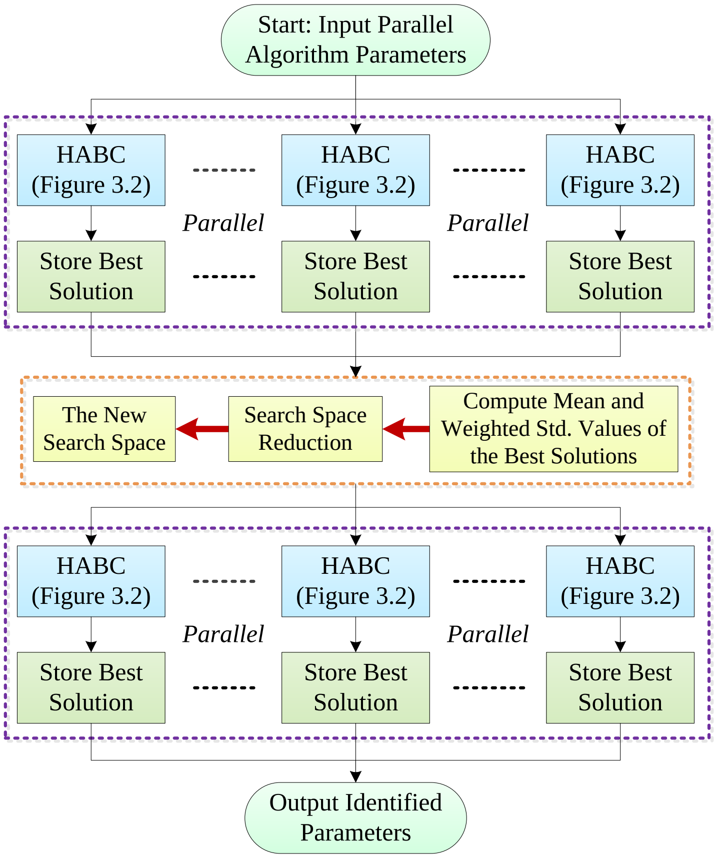


Figure 3.4: The HABC-SSR based parallel computational identification scheme.

study. These systems were taken from previous studies [17, 54, 62] so that a comparison of the proposed technique with other methodologies could be carried out.

To verify the effectiveness of the proposed technique with limited output data measurements and to learn about the influence of data availability on the performance of this algorithm, different sets of time histories of the structural response have been considered: both the case when the time histories of the structural response are measured at every DOF (full instrumentation setup) and the case when the time histories are available only at some DOFs (partial instrumentation setup) have been analyzed. Since accelerometers are commonly used in dynamic field testing, the structural response presented here is in the form of time histories of the structural accelerations.

Similar to Chapter 2, the lower bounds of the search space are taken to be half and the upper bounds twice the actual values of the parameters so that the search space can be defined as $\boldsymbol{\theta}^*/2 \leq \Gamma \leq 2\boldsymbol{\theta}^*$, where $\boldsymbol{\theta}^*$ is the parameter vector representing the true values. To analyze the effect of measurement noise on the identification process, various noise pollution levels in both input and output data have been considered: both input and output (I/O) signals have been polluted with Gaussian zero-mean white noise sequences, whose RMS values have been defined as some percentages (e.g., 0% (no noise), 5%, and 10%) of the original time-history signals. The identification analysis is repeated 10 times and the average results are reported so to obtain some statistical insights into the process. In each analysis, the mean values of $0.5(IR - IR^{ssr})$ best solutions (those with best fitness values) are considered representative of the identification results. The parameter identification error is defined as follows:

$$\varepsilon_j = \left| \frac{\hat{\theta}_j - \theta_j^*}{\theta_j^*} \right| \times 100\% \qquad (j = 1, 2, \ldots, N_\theta) \tag{3.11}$$

where $\hat{\theta}_j$ and θ_j^* are the estimated and the true values of the j-th parameter, respectively.

The numerical analyses are programmed in MATLAB$^{\circledR}$ using the Parallel Computing Toolbox$^{\text{TM}}$ with the *ClusterSize* of 12 on a standard Intel (R) Xeon (R) X5355 2.66GHz server with 32G RAM.

3.4.1 A 5-DOF system

Let's consider a 5-DOF shear type structural system as the one presented in a previous work by Perry [62]. The structural properties are: the lumped mass at each DOF $m_1 \sim m_3 = 5000$ kg and $m_4 \sim m_5 = 4000$ kg, the stiffness values $k_1 \sim k_3 = 4000$ kN/m and $k_4 \sim k_5 = 2500$ kN/m, respectively. The natural periods of the first two modes are 0.80 sec and 0.30 sec, respectively. The two Rayleigh damping coefficients a and b are 0.5733 and 3.4680×10^{-3}, respectively, corresponding to a damping ratio of 5% in each of the first two modes. In general, the mass matrix only contributes to the damping matrix with a small percentage in the Rayleigh relationship in Equation (2.16). In order to study the effects of one incomplete set of measurements, two cases are studied: (1) full sensor placement and (2) partial sensor placement, considering as output only the time histories of the accelerations measured at the 2nd, 4th and 5th DOFs. The structure is assumed to be excited by a Gaussian white noise sequence $F_5(t)$, with a RMS scaled to 1 kN, acting on the 5th DOF. This type of input excitation was chosen so to test the proposed HABC-SSR algorithm under more unfavorable conditions with respect to those used in [62] where a "smoothed" white noise sequence was used. To further penalize the approach, a very short time history of the structural accelerations (500 data points at time interval of 0.001 sec) was considered, which is identical to the one used in [62]. Since the identification is carried out using only output measurements and an attempt is made to identify the input force, the mass matrix is assumed to be known. However, in practical applications, it is possible not to know the exact value of each single mass but instead a distribution of each floor mass. Hence, in addition to the case of the mass matrix totally known a priori, the case of the mass matrix known as a normal distribution is also tested here so to quantify the effects due to the limited a priori knowledge of system mass on the identification of both structural parameters and input. In conclusion, HABC-SSR is tested in the identification of the stiffness values, of two damping coefficients and of the time histories of input force. The parameters settings of the proposed algorithm used for all cases in this example are given by: $N_\theta = 7, N_{pop} = 30, N_{max} = 60, N_{lim} = 200, m = 3, \delta = 0.2, IR = 30, IR^{ssr} = 6, IR_w^{ssr} = 1, \lambda = 6, N_{ls} = 15$ and $N_{lsiter} = 50$.

Case 1 — Mass matrix is known a priori: in this case, the proposed HABC-SSR al-

Table 3.1: Identified stiffness error for totally known mass matrix case (5-DOF system).

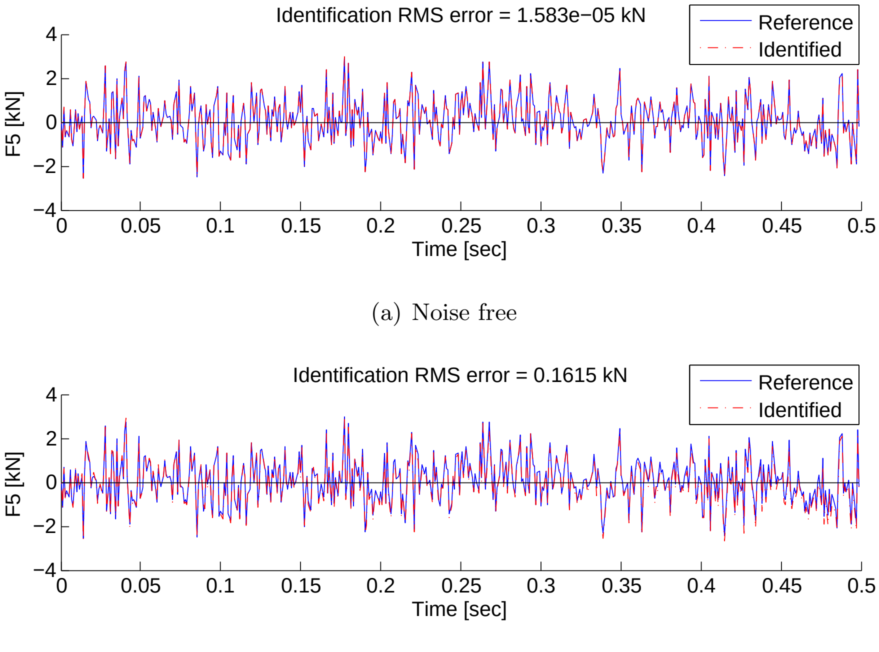
Noise level	Full sensor placement		Partial sensor placement			
			HABC-SSR		SSRM [62]	
	Mean error	Max error	Mean error	Max error	Mean error	Max error
0	0.00	0.00	0.00	0.00	0.13	0.31
5	0.67	1.09	1.08	1.88	0.93	2.19
10	0.94	1.95	1.26	2.16	2.24	6.06

Note: the numbers in the table are in percentage (%).

gorithm performs quite well in the identification of the structural parameters and of the input force time-history. Table 3.1 presents the stiffness identification errors obtained with the HABC and SSRM presented in [62]. In the noise free case, the identification of the floor stiffness is exact for both the case of a full instrumentation setup and the case of limited instrumentation. Even when dealing with noisy measurements, the performance of the HABC-SSR is excellent: for a noise level of 10% RMS, the mean error is below 1% for the full instrumentation setup while it barely passes the 1% mark (1.26%) for the case of limited instrumentation, with a maximum error around 2% (1.95% and 2.16%). These values are much lower than those obtained with the SSRM, where a maximum error up to 6% was recorded. Similar accuracy is achieved in the identification of the damping coefficients, where the stiffness proportional damping coefficient (b in Equation (2.16)) is identified with an error of 0.44% in the case of 10% RMS noise. Similar to the result in [62], the other damping coefficient a is not identified as accurately since the mass contribution to the damping is not predominant. In general, it is possible to say that the proposed HABC seems to be not sensitive to measurement noise in the identification of the unknown structural parameters.

Same accuracy can be achieved in the estimation of the unknown input force time-history. Figure 3.5 shows a comparison of the time histories of the original input force and of the identified one, considering the incomplete instrumentation setup and the cases of noise-free measurement (Figure 3.5(a)) and 10% RMS noise corrupted measurement (Figure 3.5(b)). These two figures show an almost perfect agreement between the original time-history and the identified one (with the identification RMS errors of 1.583×10^{-5} kN and 0.162 kN for the noise-free and 10% RMS noise corruption case, respectively). It is also noteworthy that, when a longer time history of noisy measurement is considered (e.g. 5000 or 10000 data points), the identification of structural parameters is still good; however, a drift will be observed in the computed velocities and displacements as well as in the identified force caused by the numerical integration of noisy data. The computed velocities, displacements and the identified force can be de-trended using either a high-pass filter or the de-trending technique introduced by Wu *et al.* [63].

Having the possibility of running the proposed HABC-SSR on a parallel scheme, it was possible to test the efficiency of running the identification process in parallel in comparison with the sequential approach. The time required to run the serial version of the algorithm was 946.7 sec, much longer than the 109.3 sec needed by the parallel strategy version, making the parallel strategy more appealing for large computational problems. In order to test the function of the SSR strategy on parameter identification refinement, Figure 3.6 presents the



(a) Noise free

(b) 10% RMS noise

Figure 3.5: Identified force with partial sensor placement (5-DOF system).

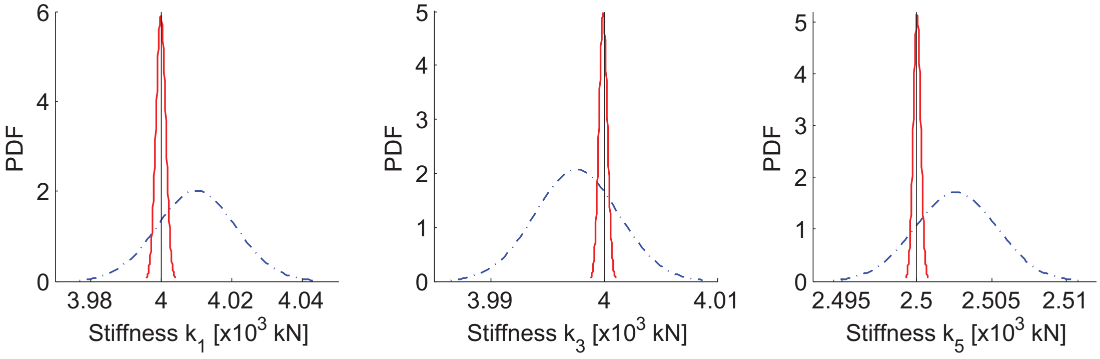


Figure 3.6: The PDFs of the stiffness parameters in the partial output scenario without noise corruption (dashed blue line: before/without SSR; solid red line: after/with SSR)

typical probability density functions (PDFs) of the identified stiffness parameters before and after the SSR in the noise-free incomplete output scenario. It can be observed that the PDFs obtained after/with the implementation of SSR have the mean values quite closer to the true values of the structural stiffnesses, with much smaller standard deviations than those obtained before/without SSR; this shows that the SSR strategy can effectively improve the accuracy of the identification process.

Case 2 — Floor masses are known as normal distributions: here, to account for uncertainties in the assessment of the floor masses, the values of the floor mass are known as normal distributions, centered on the correct value and with a standard deviation of 5% of such value. Figure 3.7 shows the distributions of the floor masses used in this example. Even in the presence of such uncertainties, the proposed HABC-SSR performs quite well, with very accurate identification of the structural parameters. As shown in Table 3.2, the mean error in the estimation of the floor stiffnesses is quite small (below 2%) in both full and partial instrumentation cases and for 10% RMS measurement noise, with a maximum error in the order of 3%. These results can be considered quite good, considering the fact that only output measurements were used in the identification. As expected, these identified stiffenesses are slightly more inaccurate than those obtained in the case of the known mass matrix because of the uncertainties in the floor masses: however, the small magnitude of such errors shows that such uncertainties do not strongly affect the identification of other structural parameters, giving a certain flexibility in determining the floor mass values.

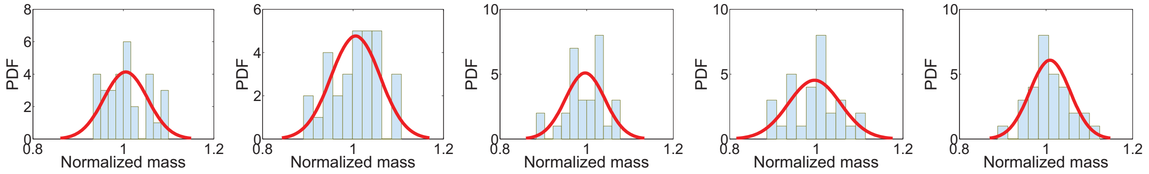


Figure 3.7: A typical normalized mass distribution for 30 independent runs.

Table 3.2: Identified stiffness error for the case of mass matrix known as a normal distribution (5-DOF system)

Noise level	Full sensor placement		Partial sensor placement	
	Mean error	Max error	Mean error	Max error
0	0.19	0.35	0.54	0.91
5	0.83	1.50	0.89	1.69
10	1.61	2.66	1.72	3.13

Note: the numbers in the table are in percentage (%).

3.4.2 A 15-DOF system

The next example is a 15-storey high rise building, previously studied by Chen and Li [54] using a modified iterative least-square based procedure (MILSP). This building is modeled as a 15-DOF shear type structure with a lumped mass and a massless spring at each floor level. The mass properties are $m_1 = 30 \times 10^3$ kg, $m_2 \sim m_{14} = 28.896 \times 10^3$ kg and $m_{15} = 27.741 \times 10^3$ kg, while the stiffness values are $k_1 = 43051$, $k_2 = 42776$, $k_3 = 42761$, $k_4 = 42536$, $k_5 = 42496$, $k_6 = 42422$, $k_7 = 42398$, $k_8 = 42372$, $k_9 = 42291$, $k_{10} = 42172$, $k_{11} = 42114$, $k_{12} = 42093$, $k_{13} = 41898$, $k_{14} = 41464$ and $k_{15} = 43051$ kN/m, respectively. The natural periods of the first two modes are 1.61 sec and 0.54 sec and the two Rayleigh damping coefficients a and b are chosen as 0.2936 and 6.406×10^{-3} respectively providing 5% damping ratio for each of the first two modes. The structure is assumed to be excited by a sinusoidal excitation induced by a dynamic shaker at the top floor. This primary input force has a magnitude of 1×10^3 kN and a frequency of 7.96 Hz. Two cases of sensor availability are considered: (1) full sensor placement and (2) partial sensor placement with sensors located at 1, 3, 5, 7, 9, 11, 13, 14 and 15 floor levels. The available time histories are

Table 3.3: Identified stiffness error (15-DOF system).

Noise level	Full sensor placement				Partial sensor placement	
	This work		MILSP [54]			
	Mean error	Max error	Mean error	Max error	Mean error	Max error
0	0.01	0.04	-	-	0.07	0.26
5	0.46	1.10	2.66	3.46	1.12	1.78
10	0.67	1.45	-	-	1.85	6.51

Note: the numbers in the table are in percentage (%).

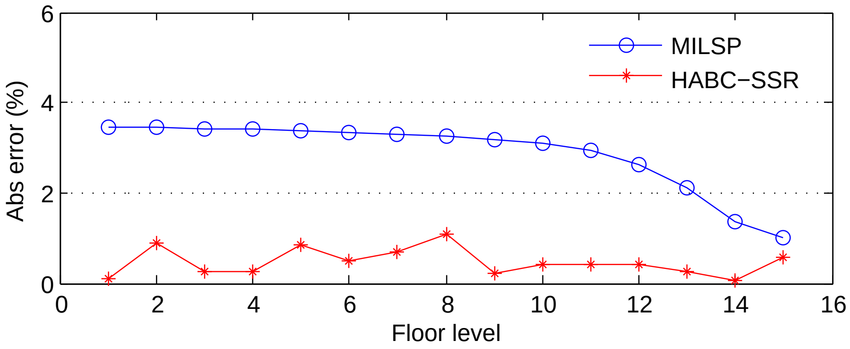
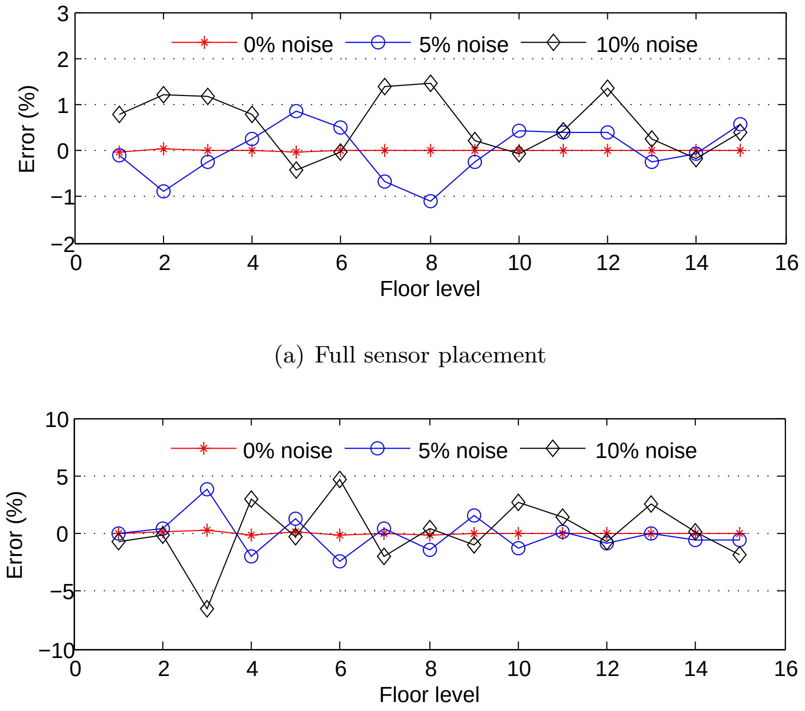


Figure 3.8: Stiffness identification error comparison between the MILSP and the proposed HABCA-SSR with full sensor placement and 5% noise corruption.

those of the structural acceleration, containing 500 data points sampled at 0.001 sec. These time histories are also contaminated with Gaussian white noise at 5% and 10% RMS levels. The objective here is to determine the fifteen stiffness values, the two damping coefficients and the unknown force acting on the top floor. The parameters HABC-SSR are given by: $N_\theta = 17, N_{pop} = 30, N_{max} = 150, N_{lim} = 200, m = 3, \delta = 0.2, IR = 30, IR^{ssr} = 6, IR^{ssr}_w = 1, \lambda = 8, N_{ls} = 20$ and $N_{lsiter} = 100$.

Even in this example of a 15-DOF system, the performance of the proposed algorithm is quite good. Looking at Table 3.3, both the mean and the max errors are quite small for both cases (full and partial instrumentation setup), even for 10% RMS measurement noise. Figure 3.8 illustrates the comparison of the floor stiffness identification error obtained by the proposed HABC-SSR and the MILSP presented in [54], clearly showing an error from the HABC-SSR that is consistently smaller than that obtained from the MILSP. Figure 3.9 presents the distribution of such identification errors for the different noise levels for the case of full instrumentation (Figure 3.9(a)) and limited instrumentation (Figure 3.9(b)) setups. For the case of a full set of sensors, errors in the estimated floor stiffnesses of the order of 1%, even in the presence of 10% RMS noise, are excellent. Same conclusion can be said when looking at the partial instrumentation case, where a mean error less than 2% is an indication of a successful identification.

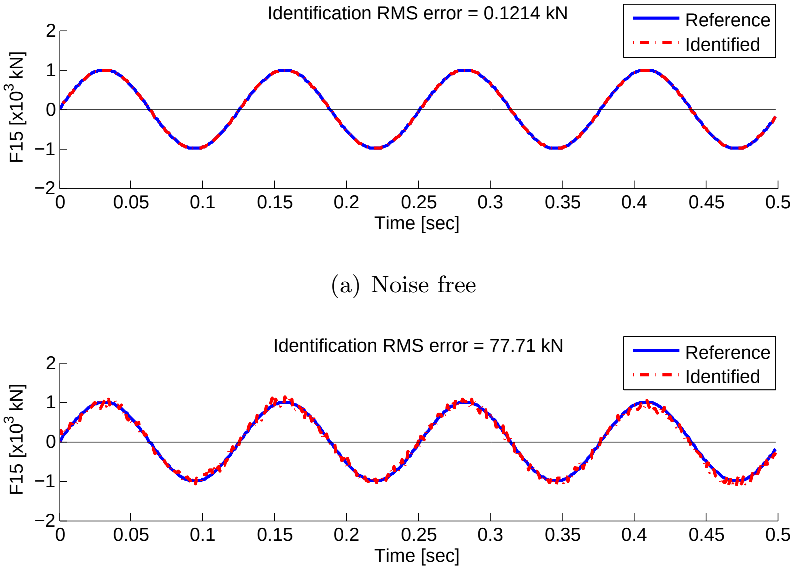
With regard to the identification of the input force, Figure 3.10 shows the comparison of the correct input force and of the identified one for the cases of noise free data (Figure 3.10(a)) and of 10% RMS noise data (Figure 3.10(b)) when only few accelerometers are available. In both cases, the identification of the input force can be considered successful, with the RMS errors of 0.121 kN and 77.710 kN, respectively.



(a) Full sensor placement

(b) Partial sensor placement

Figure 3.9: Distribution of stiffness identification errors (15-DOF system).



(a) Noise free

(b) 10% RMS noise

Figure 3.10: Identified force with partial sensor placement (15-DOF system).

3.4.2.1 Effects of "secondary forces" on the accuracy of the parameters and force identification.

Herein, we also carried out a parametric study on the effects of low-amplitude "secondary forces" distributed over the entire structure on the simultaneous identification of the structural parameters and the time-history of the "primary force" (the action of the dynamic shaker). Zero mean white noise sequences, with an identical RMS of 20 kN (2% of the amplitude of the sinusoidal force applied by the shaker) are applied at each structural floor: they can represent the effects of additional environmental low-level excitations (usually unmeasured) that are present at the time of the dynamic shaker test. Only the scenario of partial sensor placement is considered here, with three different levels of noise corruption, viz., 0%, 5% and 10% RMS white noise added to the original signals. Each time history of the structural response at different DOFs has a total of 1000 data points, with a sampling time of 0.001 sec.

Table 3.4 summarizes the identified stiffness errors while Figure 3.11 shows the detailed error distributions for the stiffness of each floor. It can be seen that the parameter identi-

Table 3.4: Identified stiffness error (under primary/secondary force excitations)

Error (%)	Noise level (%)		
	0	5	10
Mean error	0.36	1.65	2.10
Max error	0.74	3.11	8.16

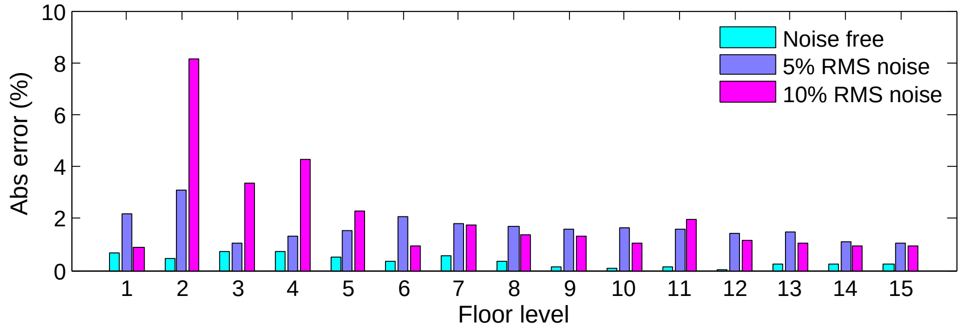


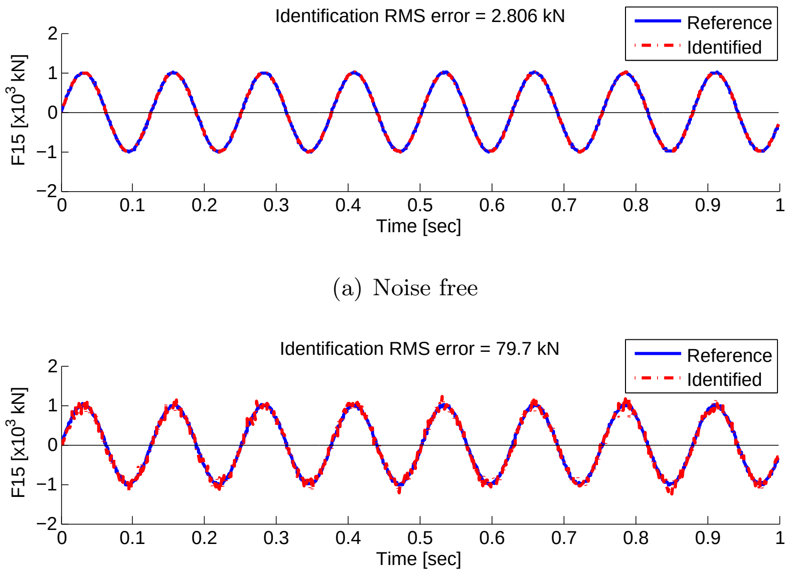
Figure 3.11: Identified stiffness errors (under primary/secondary force excitations).

fication is still quite good. Figure 3.12 shows the identified sinusoidal force induced by the shaker at the top floor for the cases of no measurement noise and 10% RMS noise. The results show an excellent agreement between the correct and the identified force, with small RMS errors (2.806 kN (noise free) and 79.700 kN (10% RMS noise)).

It is then possible to conclude that, in this example, the identification of both the structural parameters and the primary input force seems not to be affected by the presence of unknown secondary forces as long as the amplitude of such forces is small compared to that of the primary force.

3.4.3 A system of three connected buildings

Another interesting example for testing the effectiveness of the proposed HABC-SSR is represented by 3 building structures of different heights, connected through two link bridges at different levels and subjected to seismic ground excitation. In this example, already studied by Perry and Koh [17], the buildings have 5, 15 and 10 floors, respectively, and are represented by shear type models, with known floor masses, as shown in Figure 3.13. Each



(a) Noise free

(b) 10% RMS noise

Figure 3.12: Identified primary force (under primary/secondary force excitations).

of the link bridges ('link bridge 1' between the first and central building at the 3rd DOF and 'link bridge 2' between the central and second building at the seventh DOF) can be described as an axial spring whose stiffness is unknown and needs to be identified.

To generate the data used in the identification, the central building is represented by a mass distribution $m_1 \sim m_5 = 7 \times 10^5$ kg and $m_6 \sim m_{15} = 4 \times 10^5$ kg while the stiffnesses are $k_1 \sim k_5 = 6 \times 10^5$ kN/m and $k_6 \sim k_{15} = 4 \times 10^5$ kN/m. The side buildings have identical structural properties for all floors, namely, $m = 4 \times 10^5$ kg and $k = 4 \times 10^5$ kN/m. The two link bridges are modeled with linear springs whose stiffness is $k_L = 10 \times 10^5$ kN/m. These distributions of mass and stiffness corresponds to natural periods of the first two modes equal to 1.59 sec and 0.80 sec. The two Rayleigh damping coefficients a and b are chosen as 0.105 and 0.0034, respectively, resulting in a 2% damping ratio for each of the first two modes. The system is excited by the NS component of the 1940 ElCentro earthquake record and the system's responses are simulated for a duration of 5 sec using a sampling frequency of 200 Hz.

In this example, the objective is to test the performance of the HABC-SSR in identifying the fifteen stiffness values of the model of the central building, the two damping coefficients

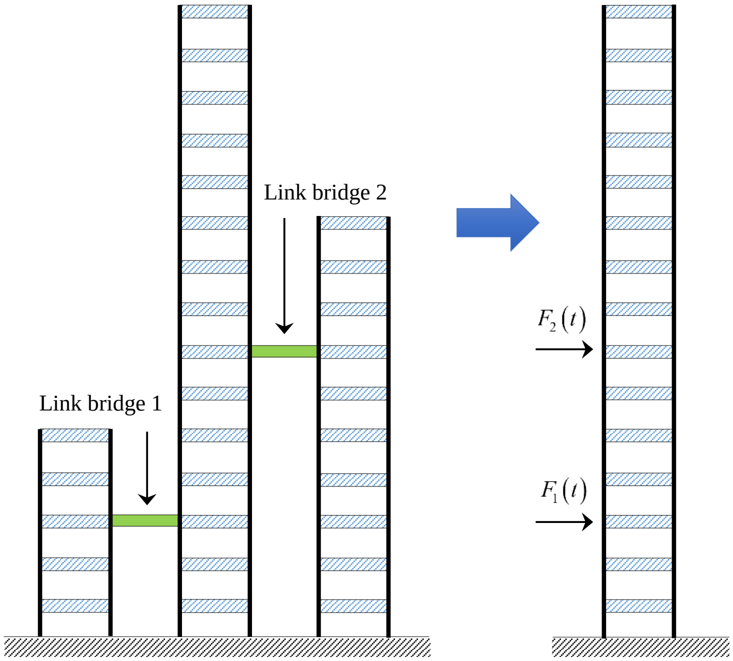


Figure 3.13: A system of three connected buildings.

a and b and the two unknown forces within the link bridges, without having any information about the two side buildings. Here, it is assumed that the floor masses of the central building are known and that acceleration time histories, with three different levels of measurement noise (no noise, 5% and 10% RMS), are only available at the 2, 3, 4, 6, 7, 8, 10, 12 and 14 DOFs. In this case, it is assumed that the ground excitation has been recorded and is known while the two forces transmitted by the link bridges to the central buildings are unknown, making it a case of known and unknown input forces. The parameter settings for the proposed HABC-SSR algorithm used in this examples are chosen as follows: $N_\theta = 17, N_{pop} = 30, N_{max} = 200, N_{ls} = 200, m = 3, \delta = 0.2, IR = 30, IR^{ssr} = 6, IR_w^{ssr} = 1, \lambda = 8, N_{ls} = 20$ and $N_{lsiter} = 100$.

Table 3.5 summarizes the mean and maximum errors in the identified stiffness values from the HABC-SSR and compares them with those obtained in [17]. In the noise free case, the HABC-SSR identification is almost perfect (mean error of about 0.02% and maximum error of 0.05%). In the presence of measurement noise, the result of a mean error less than 1% and a maximum error less than 2%, for both 5% and 10% RMS noise, is still quite

Table 3.5: Identified stiffness error (a system of three connected buildings)

Noise level	HABC-SSR		SSRM [17]	
	Mean error	Max error	Mean error	Max error
0	0.02	0.05	0.34	1.53
5	0.78	1.60	0.85	2.08
10	0.94	1.82	1.47	3.82

Note: the numbers in the table are in percentage (%).

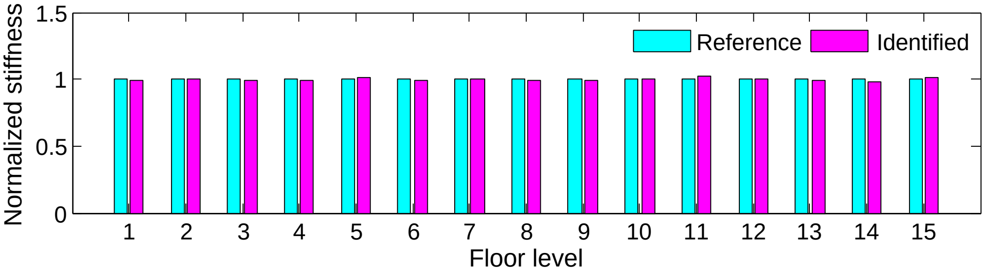


Figure 3.14: Identified normalized stiffness with 10% noise corruption (a system of three connected buildings).

remarkable: Figure 3.14 shows the comparison of the true stiffnesses and of the identified stiffnesses for the case of 10% RMS measurement noise, with almost perfect agreement between the true and the identified values.

With regard to the identification of the two unknown forces from the link bridges, the proposed HABC-SSR algorithm is capable of successfully identifying the unknown components of the input force. This can be seen in Figure 3.15, where the two identified forces are compared with the correct ones for the case of 10% RMS noise. In the presence of noise, a high-pass filter is used so to remove a linear trend caused by the numerical integration of noisy accelerations. It can be seen that the identification of the two time histories is quite accurate, even in the presence of measurement noise. It is noteworthy that the identification of the force $F_2(t)$ is more accurate than the one of $F_1(t)$: this slight discrepancy could be caused by the fact that $F_2(t)$ has a large magnitude than $F_1(t)$ and so it has a bigger impact on the system's response. This phenomenon was also reported in [17].

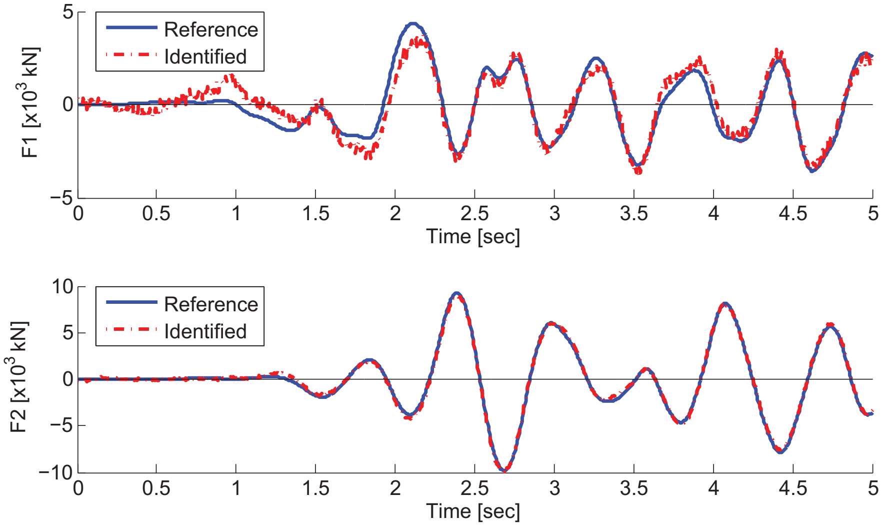


Figure 3.15: Identified coupling forces with 10% noise corruption (a system of three connected buildings).

3.5 Conclusions

This chapter presents a hybrid parallel strategy for the simultaneous identification of structural parameters and, when possible, of input time histories using an incomplete set of acceleration measurements. The proposed strategy is a parallel computational scheme that combines a hybrid algorithm, consisting of the MABC algorithm and the Nelder-Mead simplex method, with a Search Space Reduction technique. Three numerical examples have been carried out to verify the applicability and effectiveness of the proposed method. The simulation results in this study lead to the following observations and conclusions:

(1) In the process of identifying structural parameters, if the measured signals are only accelerations and if the masses are known, the time histories of the dynamic input can be simultaneously computed by using a modified Newmark integration scheme based on the measurements and on the current estimated structural parameters.

(2) In the case where the input excitation has to be identified, the proposed method requires a certain a priori knowledge of the structural mass distribution, either the entire mass matrix or the distribution of the various floor masses. The proposed identification strategy provides very good parameter and input force estimation even

with 5% standard deviation of mass estimation error.

(3) For noise free measurement data, the proposed strategy can exactly identify the unknown parameters and input forces for the full instrumentation setup case as well as for the incomplete instrumentation case. When output measurements are corrupted by noise, the identification results are still quite good with reasonably small errors.

(4) The input force time histories can be exactly identified in the noise free case. In the presence of measurement noise, the identified time histories are also quite accurate, even though the data need to be de-trended to overcome the problems linked to the numerical integration of noisy accelerations. The numerical studies show that the input force(s) can be well identified even with high noise corruption (10% RMS noise).

(5) A parametric study shows that the identification of both the structural parameters and the primary input force seems not to be affected by the presence of unknown secondary forces as long as the amplitude of such forces is small compared to that of the primary force.

(6) The parallel scheme can dramatically save computational time compared with the serial scheme if a powerful computer with multiple processors is used.

In general, the identification results show that the proposed technique is powerful, robust and efficient in the simultaneous identification of structural parameters and input time histories even from incomplete output-only measurements with high noise corruption.

Chapter 4

Extension of the Hybrid ABC Algorithm with Bayesian Inference for Probabilistic FEM Updating

In this chapter, a Bayesian inference methodology is presented for the probabilistic finite element model updating and damage detection of structural systems. This chapter is reproduced from the paper co-authored with Professor Raimondo Betti, which is currently under review [64].

4.1 Introduction

Structural finite element model (FEM) updating, based on input-output or output-only measurements, has been widely used in structural health monitoring, damage and risk evaluation, structural control, etc. Since modeling errors between the theoretical FEM and the real structure (e.g. due to uncertainties caused by dispersed material properties, variation of construction process, complicated component joint behavior, etc.) are always present, updating (correcting) the initial/prior FEM characterized by a set of system parameters is necessary. Because of the pressing need in the industry area, this topic has gained increasing interest in various disciplines [65–73].

In the process of model updating, one's objective is to find the optimal system parameters and the uncertainties associated with them. The parameters to be identified can be either physical quantities (e.g. structural mass, damping, stiffness, etc.) or modal quantities (e.g. natural frequency, mode shape, transfer function, etc.) of a structural system. The data used in the updating operation is usually the time history of the structural response (e.g. dynamic strains, displacements, accelerations, etc.) recorded at various locations and, when possible, the time history of the input excitation. There have been many attempts to study system identification for model updating in recent years (see Chapters 2–3).

Model updating belongs to the general classification of inverse problems whose well- or ill-posedness determines whether the model is 'globally identifiable', 'locally identifiable' or 'globally unidentifiable' [74]. In the globally identifiable case (well-posed problem), the optimal parameter region is centered at a point, while the locally identifiable case (ill-posed problem) has multiple sets of optimal parameter regions (multiple global optimal points). In the globally unidentifiable case (badly ill-posed problem), there exist infinite sets of global optimal solutions (e.g., points in a flat region of the parameter space). In dynamic testing, for example, the model 'identifiability' is usually determined by the number of observations and by the spatial placement of sensors [58].

There are two categories of methods, in general, for FEM updating and system identification, namely, deterministic *vs.* probabilistic methods. The difference between these two methods is that deterministic methods aim to obtain a single set of optimal parameters while probabilistic methods estimate statistical distributions of the structural parameters providing a family of possible models [74].

In most deterministic methods, the uncertain system parameters are determined by optimizing (usually minimizing) the objective function defined as the Mean Square Error (MSE) between the measured and the predicted data [5]. To date, deterministic methods such as least-squares-based methods, heuristic algorithms and filtering techniques have been developed for model updating and system identification [5, 18, 25, 31, 47, 55, 56, 65, 75]. By providing well-defined values of the structural parameters, deterministic methods can effectively determine the structural conditions or health (e.g. damage extent) when the updating/identification problem is well-posed; however, when many modeling uncertainties are involved, they fail to give a satisfactory results. On the contrary, probabilistic methods seem to be much more "robust" in dealing with uncertainties due to their ability to consider more than one model weighted by the probabilities conditional on the measured data [76].

One of the most popular probabilistic methods for FEM updating is known as the Bayesian approach; such an approach takes into account the complete information associated within the measured data for statistical inference while an appropriate likelihood function is given [73]. Recently, many studies have been done on Bayesian FEM updating for a robust prediction of future structural response and for a reliable assessment of structural condition. For example, a comprehensive Bayesian model updating framework was first proposed by Beck and Katafygiotis [66], and then further applied to structural dynamic models with different types of testing data (e.g. accelerations, modal frequencies, mode shapes, etc.), using several versions of Monte Carlo Method (MCM), such as Markov Chain MCM (MCMCM) [74], adaptive MCMCM [77], transitional MCMCM [76], hybrid MCM [70], etc. Other studies on structural damage detection, uncertainty quantification and reliability assessment using Bayesian methodologies can be found in References [73, 78–80].

Nevertheless, challenges still exist in the Bayesian FEM updating in spite of its attractive robustness in the direct quantification of uncertain parameters. To wit, they are:

- *Computational challenges*: there exist severe computational challenges when a large number of uncertain parameters need to be updated (in a high-dimensional parameter space), especially when the model is not globally identifiable [70]. Sampling issues arise when the posterior probability density function (PDF) is highly nonconvex with multiple/infinite sharply shaped optima.
- *Challenge in construction of the likelihood function*: in some applications, the likelihood function cannot be written analytically. Even though the correlation formula can be used as the likelihood function in terms of a single observed sequence, the joint PDF of these correlations is difficult to be explicitly derived. Moreover, the computation burden becomes expensive or, sometimes, impossible though an implicit form of the likelihood function exists [73].
- *Challenge of the updated PDF validation*: due to the complex topology of the updated PDF, in practice, we cannot validate the solution.

The basic idea in this work is to develop an effective and efficient optimization method for solving high-dimensional optimization problems in probabilistic FEM updating, without the need to directly compute the PDFs of the structural parameters. The objective function is a regularized error function derived from the Bayesian inference for model updating, in which the regularization term comes from the prior PDF of system parameters while the regularization parameter is defined conditional on the measured data. The optimal set of uncertain parameters can be determined by minimizing the regularized MSE function using powerful optimization algorithms (e.g., the proposed MABC-BFGS; see Section 4.3 for details), given the measured data, the prior PDFs of model parameters and an appropriate likelihood function.

The chapter is organized as follows. Section 4.2 describes the Bayesian probabilistic framework in FEM updating. Section 4.3 presents a modified ABC (MABC) algorithm, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method and the proposed hybrid MABC-BFGS algorithm for solving the high-dimensional optimization problems in Bayesian analyses. Then, a numerical example of the 3D Phase I IASC-ASCE benchmark structure is illustrated in Section 4.4. Furthermore, the proposed algorithm is validated for both model updating and damage detection using the experimental data sets of a three-storey frame structure from LANL in Section 4.5. Finally, discussions and concluding remarks are presented in Section 4.6.

4.2 Bayesian probabilistic framework

4.2.1 Bayes' Theorem

The focal concept of Bayesian inference is *Stochastic Embedding* (e.g., embedding any deterministic model into a class of probabilistic models), in which a model is described by probability distributions of both the unknown parameters and the prediction error [66]. Let's say a system can be discretized using a FEM and thus characterized by a set of uncertain model parameters $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$, where N_θ denotes the number of parameters. In order to define this model, one needs to determine the uncertain parameters. If the model is deterministic, then the goal is to determine a unique set of parameter values while, if the model is stochastic, each parameter will be characterized by a PDF whose characteristic parameters (e.g., mean and standard deviation) need to be determined. One possibility is to use the Bayes' Theorem that helps identifying the parameter uncertainties using the system's response measurements. In Bayesian inference, provided that the joint prior PDF of the structural parameters, $p(\boldsymbol{\theta})$, is given (e.g., by user's assessment of the parameter uncertainties), the joint posterior PDF of $\boldsymbol{\theta}$ conditional on the observed data $\mathcal{D}$ is defined as:

$$p(\boldsymbol{\theta}|\mathcal{D}) = c^{-1} p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \tag{4.1}$$

with c being the normalizing factor (the evidence given by data $\mathcal{D}$) which can be written as:

$$c = p(\mathcal{D}) = \int_{\Theta} p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \tag{4.2}$$

Here, $p(\mathcal{D}|\boldsymbol{\theta})$ is the likelihood function which gives a measure of the agreement between the measured and the predicted system responses.

4.2.2 Formulation of the likelihood function

The process of *Stochastic Embedding* can be realized by considering the time history of the prediction error $\boldsymbol{\epsilon}(t_j)$ representing the difference between the measured and the predicted system responses [66]. Assuming that the discrete system response $\mathbf{y}(t_j)$ is available at N_o ($N_o \leq$ the total number of DOFs) observed DOFs, the corresponding predicted system response $\mathbf{y}(t_j, \boldsymbol{\theta})$ at a generic time instant $t_j (j = 1, 2, \ldots, N)$ can be modeled in terms of the measured system response $\mathbf{y}(t_j)$ and of the predicted error $\boldsymbol{\epsilon}(t_j, \boldsymbol{\theta})$ (due to measurement noise and modeling error) for a specified set of parameters $\boldsymbol{\theta}$:

$$\mathbf{y}(t_j) = \mathbf{y}(t_j, \boldsymbol{\theta}) + \boldsymbol{\epsilon}(t_j, \boldsymbol{\theta}) \tag{4.3}$$

where N is the total number of data points. At this point, to describe the uncertainty of the prediction error, the characterization of $\boldsymbol{\epsilon}$ can be carried out through some probability models that produce the maximum uncertainty based on the Principle of Maximum (Information) Entropy [81]. To wit, $\boldsymbol{\epsilon}$ can be modeled as a discrete zero-mean Gaussian white noise process [66]; namely, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \Sigma_{\varepsilon}) = \mathcal{N}(0, \sigma^2 \mathbf{I}_{N_o \times N_o})$ with σ being a positive scalar for a specified $\boldsymbol{\theta}$. Let's assume that the prediction errors are statistically independent given a set of measured data $\mathcal{D} = \{\mathbf{y}(t_j) : j = 1, 2, \ldots, N\}$. With these assumptions, the likelihood function can be expressed following the multivariate Gaussian PDF as:

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{j=1}^{N} p\left(\mathbf{y}(t_j)|\boldsymbol{\theta}\right) = \frac{1}{(2\pi\sigma^2)^{N_oN/2}} \exp\left[-\frac{N_oN}{2\sigma^2}\mathfrak{J}_g(\boldsymbol{\theta};\mathcal{D})\right] \tag{4.4}$$

with

$$\mathfrak{J}_g(\boldsymbol{\theta};\mathcal{D}) = \frac{1}{N_oN}\sum_{i=1}^{N_o}\sum_{j=1}^{N}\left[y_i(t_j) - y_i(t_j, \boldsymbol{\theta})\right]^2 \tag{4.5}$$

where $\mathfrak{J}_g(\boldsymbol{\theta};\mathcal{D})$ is the *goodness-of-fit* function (MSE between the measured and the simulated system outputs). The stochastic independence here is information-based and should not be confused with causal independence. For example, the predictions errors at previous time steps have no influence on those at later time steps [82].

4.2.3 The prior and posterior PDFs

Before applying the Bayes's Theorem to update the system parameters, we need to know their prior distributions. Now, let's assume that the prior system parameter vector $\boldsymbol{\theta}$ follows a multivariate Gaussian distribution with mean $\bar{\boldsymbol{\theta}} \in \mathbb{R}^{N_\theta}$ and covariance matrix $\Sigma_\theta \in \mathbb{R}^{N_\theta \times N_\theta}$, so its joint PDF can be written as:

$$p(\boldsymbol{\theta}) = \frac{1}{(2\pi)^{N_\theta/2}|\Sigma_\theta|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})\Sigma_\theta^{-1}(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^T\right] \tag{4.6}$$

This can be built using the a priori information on the parameters available to the user before the analysis, and so it can be defined as the prior PDF. It is noteworthy that if the prior parameters are assumed to be independent from each other, $p(\boldsymbol{\theta})$ could be obtained from the product of the prior distribution of each parameter, namely, $p(\boldsymbol{\theta}) = \prod_{k=1}^{N_\theta} p(\theta_k)$. In this case, Σ_θ becomes a diagonal matrix whose diagonal elements consist of the parameters' standard deviations.

However, since σ^2 is an unknown variable in Equation (4.4), it also needs to be updated. In general, since σ^2 is always positive, its prior distribution can be modeled by an inverse Gamma distribution or a lognormal distribution (note that these two distributions can be equivalently transformed to each other by adjusting their distribution parameters) [73]. Let's now assume that, based on the a priori information, σ^2 follows the inverse Gamma distribution given by $p(\sigma^2) = IG(\alpha, \beta)$, where α ($\alpha > 0$) and β ($\beta > 0$) are the constant "hyperparameters",

$$p(\sigma^2) = \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \exp\left(-\frac{\beta}{\sigma^2}\right) \tag{4.7}$$

where $\Gamma(\cdot)$ is the Gamma function.

A hierarchical Bayesian inference can then be used to provide a bridge between the prior and the posterior PDFs of both $\boldsymbol{\theta}$ and σ^2 [83]. Using Equation (4.1), we can state that the augmented posterior PDF $p(\boldsymbol{\theta}, \sigma^2|\mathcal{D})$ is proportional to

$$p(\boldsymbol{\theta}, \sigma^2|\mathcal{D}) \propto p(\mathcal{D}|\boldsymbol{\theta}, \sigma^2) p(\boldsymbol{\theta}) p(\sigma^2) \tag{4.8}$$

where $p(\mathcal{D}|\boldsymbol{\theta}, \sigma^2)$ is the augmented likelihood function which is given by Equation (4.4), identical to $p(\mathcal{D}|\boldsymbol{\theta})$. The substitution of Equations (4.4), (4.6) and (4.7) into Equation (4.8) yields

$$p(\boldsymbol{\theta}, \sigma^2|\mathcal{D}) \propto \frac{1}{\sigma^{N_o N + 2(\alpha+1)}} \exp\left[-\frac{1}{\sigma^2}\left(\frac{N_o N}{2}\mathfrak{J}_g(\boldsymbol{\theta}; \mathcal{D}) + \beta\right) - \frac{1}{2}(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})\Sigma_\theta^{-1}(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^T\right] \tag{4.9}$$

In determining the uncertain parameters in Bayesian inference, the objective is to find the maximum value of the posterior PDF $p^{max}(\boldsymbol{\theta}, \sigma^2|\mathcal{D})$ which is a function of $\boldsymbol{\theta}$ and σ (see Figure 4.1). The optimal solution of the model parameters $\boldsymbol{\theta}$ and the prediction error covariance parameter σ^2 are the corresponding parameter set that lead $p(\boldsymbol{\theta}, \sigma^2|\mathcal{D})$ to the maximum:

$$\{\hat{\boldsymbol{\theta}}, \hat{\sigma}^2\} = \arg \max_{\boldsymbol{\theta}^l \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}^u, \sigma > 0} \left\{p(\boldsymbol{\theta}, \sigma^2|\mathcal{D})\right\} \tag{4.10}$$

where $\boldsymbol{\theta}^l$ and $\boldsymbol{\theta}^u$ are the lower and the upper parameter bound vectors.

4.2.4 The objective function for optimization

The sequential optimization method (also known as Gibbs steps [84]) can then be used to optimize the parameters in Equation (4.10). Firstly, we take the derivative of $p(\boldsymbol{\theta}, \sigma^2|\mathcal{D})$ w.r.t. σ^2 and let its value be zero so as to obtain the optimal value $\hat{\sigma}^2$, namely,

$$\frac{\partial p(\boldsymbol{\theta}, \sigma^2|\mathcal{D})}{\partial(\sigma^2)} = 0 \quad \Rightarrow \quad \hat{\sigma}^2 = \frac{2\beta + N_o N \mathfrak{J}_g(\boldsymbol{\theta}; \mathcal{D})}{2\alpha + N_o N + 2} \tag{4.11}$$

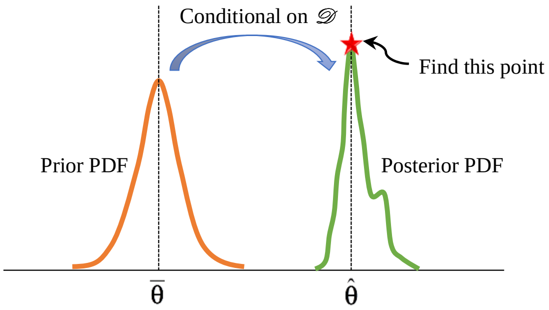


Figure 4.1: A schematic representation of the updating process: find the solution that maximizes the posterior PDF.

Because of the presence of $\mathfrak{J}_g(\boldsymbol{\theta};\mathcal{D})$, we can state that the optimal value $\hat{\sigma}^2$ is also a function of the parameters $\boldsymbol{\theta}$, *v.i.z.* $\hat{\sigma}^2(\boldsymbol{\theta})$. At this point, we treat $\hat{\sigma}^2$ as a known variable and plug it into Equation (4.9) to obtain:

$$\begin{aligned} p(\boldsymbol{\theta},\sigma^2|\mathcal{D}) &\propto \ln\left[p(\boldsymbol{\theta},\sigma^2|\mathcal{D})\right] \\ &\propto -\left[N_oN+2(\alpha+1)\right]\ln[\hat{\sigma}(\boldsymbol{\theta})] - \frac{1}{\hat{\sigma}^2(\boldsymbol{\theta})}\left(\frac{N_oN}{2}\mathfrak{J}_g(\boldsymbol{\theta};\mathcal{D})+\beta\right) \\ &\quad -\frac{1}{2}(\boldsymbol{\theta}-\bar{\boldsymbol{\theta}})\Sigma_\theta^{-1}(\boldsymbol{\theta}-\bar{\boldsymbol{\theta}})^T \end{aligned} \tag{4.12}$$

Maximizing $p(\boldsymbol{\theta},\sigma^2|\mathcal{D})$ is equivalent to minimizing the expression as follows (e.g., after substituting Equation (4.11) into (4.12)):

$$G(\boldsymbol{\theta}) = \underbrace{[N_oN+2(\alpha+1)]\ln[\hat{\sigma}(\boldsymbol{\theta})]}_{\text{part 1}} + \underbrace{\frac{1}{2}(\boldsymbol{\theta}-\bar{\boldsymbol{\theta}})\Sigma_\theta^{-1}(\boldsymbol{\theta}-\bar{\boldsymbol{\theta}})^T}_{\text{part 2}} + \underbrace{\frac{N_oN}{2}+\alpha+1}_{\text{part 3}} \tag{4.13}$$

The second part in Equation (4.13) is called the regularization term (or *regularizer*), which is used to improve the well-posedness of inverse problems (see the generalized Tikhonov regularization for nonlinear ill-posed problems in [85]). The value of $G(\boldsymbol{\theta})$ increases as the radial distance of $\boldsymbol{\theta}$ from the origin $\bar{\boldsymbol{\theta}}$ increases in any direction. If there exist two or more models fitting the measured data identically well (e.g., affected by modeling errors or high noise corruptions), the *regularizer* helps trimming down the set of optimal system parameters to be the unique one with the smallest 2nd norm [73].

In the case in which the prior distribution of σ^2 is unknown, $\hat{\sigma}^2$ can be separately approximated by a purely data-driven term as:

$$\hat{\sigma}^2 = \mathfrak{J}_g(\boldsymbol{\theta}; \mathcal{D}) \tag{4.14}$$

It is also noteworthy that this data-driven term is linked to a special case of the inverse Gamma distribution (e.g, given $\alpha \ll N_o N$ and $\beta \to 0$, we have $\hat{\sigma}^2 \to \mathfrak{J}_g(\boldsymbol{\theta}; \mathcal{D})$ leading to the identical form in Equation (4.14)). Finally, if necessary, the prediction error covariance can be approximated by $\hat{\sigma} = \sqrt{\mathfrak{J}_g(\hat{\boldsymbol{\theta}}; \mathcal{D})}$, after the optimal value of $\hat{\boldsymbol{\theta}}$ has been determined.

4.2.5 Posterior PDF approximation

As shown in the work by Yuen *et al.* [86], the posterior PDF of the system parameters, $p(\boldsymbol{\theta}|\mathcal{D})$, can be approximated by a normal distribution, after an optimal set of uncertain parameters, $\hat{\boldsymbol{\theta}}$, as well as the prediction error covariance, $\hat{\sigma}^2$, have been identified. For the case of a single set of data, the joint posterior PDF can be approximated by a Gaussian distribution centered at $\hat{\boldsymbol{\theta}}$:

$$p(\boldsymbol{\theta}|\mathcal{D}) \approx \mathcal{N}(\hat{\boldsymbol{\theta}}, \mathcal{H}^{-1}(\hat{\boldsymbol{\theta}})) \tag{4.15}$$

where $\mathcal{H}^{-1}$ represent the inverse of the Hessian matrix of the objective function $G(\boldsymbol{\theta})$ in Equation (4.13). When multiple data sets are used, the posterior PDF $p(\boldsymbol{\theta}|\mathcal{D})$ can be approximated by a weighted sum of Gaussian distributions centered at s optimal points $\hat{\boldsymbol{\theta}}^{(s)}$ [87]:

$$p(\boldsymbol{\theta}|\mathcal{D}) \approx \sum_{k=1}^{s} w_k \mathcal{N}(\hat{\boldsymbol{\theta}}^{(k)}, \mathcal{H}^{-1}(\hat{\boldsymbol{\theta}}^{(k)})) \tag{4.16}$$

where w_k are the weights, given by

$$w_k = \frac{p\left(\hat{\boldsymbol{\theta}}^{(k)}\right) p\left(\hat{\sigma}^{2(k)}\right) \left|\mathcal{H}(\hat{\boldsymbol{\theta}}^{(k)})\right|^{-1/2}}{\sum_{j=1}^{s} p\left(\hat{\boldsymbol{\theta}}^{(j)}\right) p\left(\hat{\sigma}^{2(j)}\right) \left|\mathcal{H}(\hat{\boldsymbol{\theta}}^{(j)})\right|^{-1/2}} \tag{4.17}$$

Herein, s represents the total number of data sets. It is noted that the marginal posterior PDFs can be extracted from the joint posterior PDF.

4.2.6 Iterative Bayesian model updating procedure

An iterative model updating procedure is proposed here. The pseudo-code for updating the uncertain parameters using Bayesian inference is illustrated in Algorithm 2.

Algorithm 2 – Bayesian updating procedure of uncertain parameters

```
Start with an admissible initial guess for the unknown parameters $\boldsymbol{\theta}^0$;
Compute the MSE $\mathfrak{J}_g(\boldsymbol{\theta}^0; \mathcal{D})$ using Equation (4.5);
Compute the prediction error covariance $\sigma_0^2$ using either Equation (4.11) or Equation (4.14);
Compute the objective function $G(\boldsymbol{\theta}^0)$ using Equation (4.13);
$k \leftarrow 0$;
while convergence is not satisfied do
    Update the parameters $\hat{\boldsymbol{\theta}}^k \leftarrow \boldsymbol{\theta}^k + \delta\boldsymbol{\theta}^k$ using optimization techniques (e.g., MABC-BFGS);
    Compute the MSE $\mathfrak{J}_g(\hat{\boldsymbol{\theta}}^k; \mathcal{D})$ using Equation (4.5);
    Compute the prediction error covariance $\sigma_k^2$ using either Equation (4.11) or Equation (4.14);
    Compute the objective function $G(\hat{\boldsymbol{\theta}}^k)$ using Equation (4.13);
    if $G(\hat{\boldsymbol{\theta}}^k) < G(\boldsymbol{\theta}^k)$ then
        $\boldsymbol{\theta}^k \leftarrow \hat{\boldsymbol{\theta}}^k$;
    else
        $\boldsymbol{\theta}^k \leftarrow \boldsymbol{\theta}^k$;
    end if
    $k \leftarrow k + 1$;
end while
Compute the posterior PDF of system parameters using Equation (4.15) or Equation (4.16).
```

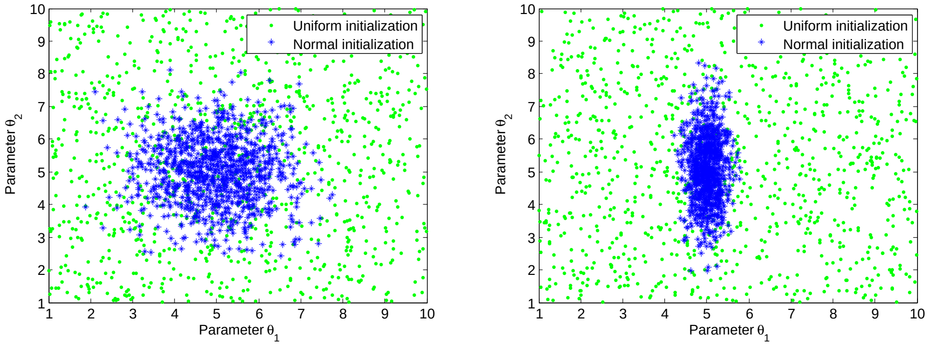
4.3 The proposed MABC-BFGS algorithm

4.3.1 The modified artificial bee colony algorithm

Herein, we propose four modifications to the standard ABC algorithm so as to improve its convergence. To wit, these modifications involve population initialization, solution updating, probability calculation/selection, and Scout solution rebirth.

4.3.1.1 Modification I: initialization

When one deals with Bayesian inference, the a priori PDF of system parameters, e.g. normal or lognormal distribution, may be reasonably assumed. In this case, the population can be initialized following such a priori distribution. Let's take the normal distribution as the a priori distribution: each component in a parameter set $\boldsymbol{\theta}$ follows a prior normal distribution



(a) $\bar{\theta}_1 = \bar{\theta}_2 = 5$, $\nu_1 = \nu_2 = 20\%$

(b) $\bar{\theta}_1 = \bar{\theta}_2 = 5$, $\nu_1 = 5\%$, $\nu_2 = 20\%$

Figure 4.2: Uniform initialization *vs.* normal initialization with 1000 samples.

with the mean value $\bar{\theta}_j$ and the corresponding coefficient of variation (c.o.v.) ν_j. Thus, each individual in the initial population Θ (of the order $N_{pop} \times N_\theta$) could be generated by:

$$\Theta_{ij} = \mathcal{N}\left(\bar{\theta}_j, \nu_j^2 \bar{\theta}_j^2\right) \tag{4.18}$$

where $i = 1, 2, \ldots, N_{pop}$ and $j = 1, 2, \ldots, N_\theta$.

This modified initialization takes the advantage of the a priori knowledge of the system parameter distributions. A typical comparison between the uniform initialization and the normal initialization is visualized in Figure 4.2. It can be observed that the initial samples are dependent of the prior PDFs of the parameters.

4.3.1.2 Modification II: solution updating

It can be observed that the solution updating strategy in Equation (2.7) generates a candidate solution by performing a single point perturbation based on a uniformly random combination of the original solution and another one. This has low efficiency because the new solution set comes from mutation of only a single parameter while other parameters are kept at the original values as shown in Figure 4.3. Thus, a multiple-point mutation scheme is presented to improve the solution updating efficiency (see Figure 4.3). Due to the nature of uniform randomness, the probability of generating a good solution is identical to that of generating a bad one. This is beneficial for solution exploration (global search) but is less powerful in solution exploitation (local search) [88]. Therefore, a best-guided updating strategy with multiple-point mutation is proposed here for generating candidate solutions, considering both the randomness and information associated with the current best solution. To wit, the modified solution updating strategy is written as:

$$\Theta_{ij}^{up} = \begin{cases} \Theta_{ij}, & \text{if} \quad \text{rand} \geq \rho \\ \Theta_{ij} + 2(\text{rand} - 0.5) \cdot (\Theta_{ij} - \Theta_{kj}) + \text{rand} \cdot (\Theta_{bj} - \Theta_{ij}), & \text{if} \quad \text{rand} < \rho \end{cases} \tag{4.19}$$

where the superscripts and the subscripts have the same meaning as shown in Section 2.3.2. The superscript 'b' denotes the current best solution in the population. ρ is a cut-off control parameter which is basically a measure of the likeliness that the solution updating process is activated. This parameter is similar to the crossover/mutation probability used in genetic algorithms. An optimal value $\rho = 0.1$ is used in this chapter, which is obtained from the numerical optimization test of six benchmark functionals (Sphere, Rosenbrock, Griewank, Rastrigin, Ackley, and Schwefel functions) as used in [88].

4.3.1.3 Modification III: probability selection

The probability-based selection process, usually conducted through the Roulette Wheel Strategy, can be carried out by generating a uniformly distributed random number r_i in [0, 1] and comparing it with the probability q_i. If $q_i \geq r_i$, then the i-th corresponding solution Θ_i will be selected; otherwise ($q_i < r_i$), it will not be selected. Looking at the nature of the probability defined in the standard ABC algorithm (see Equation (2.11)), we find that the solution associated probabilities are small. Due to the randomness of r_i, it might happen that $r_i > q^{max}$, e.g., $q^{max} = \max\{q_1, q_2, \ldots, q_{N_{pop}}\}$, leading to an undesired consequence that the best solution is not selected. Especially, in the extreme case that the

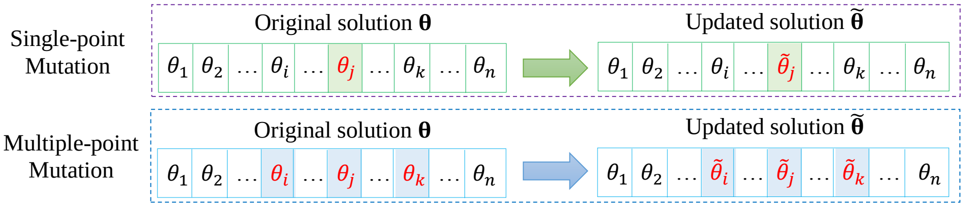

Single-point Mutation
Original solution $\boldsymbol{\theta}$
θ_1 θ_2 … θ_i … θ_j … θ_k … θ_n
Updated solution $\tilde{\boldsymbol{\theta}}$
θ_1 θ_2 … θ_i … $\tilde{\theta}_j$ … θ_k … θ_n
Multiple-point Mutation
Original solution $\boldsymbol{\theta}$
θ_1 θ_2 … θ_i … θ_j … θ_k … θ_n
Updated solution $\tilde{\boldsymbol{\theta}}$
θ_1 θ_2 … $\tilde{\theta}_i$ … $\tilde{\theta}_j$ … $\tilde{\theta}_k$ … θ_n

Figure 4.3: A schemtic representation of the solution updating with single- and multiple-point mutation. Note that the red marked parameters denote the points for mutation.

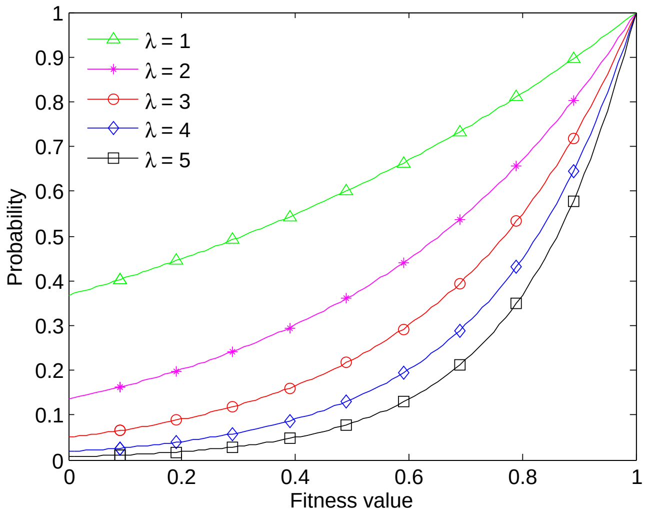


Figure 4.4: The variation of the probability with respect to the parameter λ

solutions, in a large size population, are quite close to each other with similar fitness values, the associated probability values are extremely small according to Equation (2.11); thus, the probability of selecting good solutions remains small, and this goes against the purpose of solution exploitation.

To prevent this situation, we propose a modified definition of probability based on exponentials, expressed as:

$$q_i = \exp\left[\lambda\left(\frac{fit(\Theta_i)}{\max_{1\leq i\leq N_{pop}} fit(\Theta_i)} - 1\right)\right] \tag{4.20}$$

where λ is a positive constant. The values of q_i in Equation (4.20) are between 0 and 1, with the best solution associated with $q^{max} = 1$. This formulation guarantees the selection of the best solution. Figure 4.4 presents the variation of the probability in Equation (4.20), as a function of the fitness value, with respect to the parameter λ. Parametric convergence studies on the mathematical benchmark functions shows that an excellent optimization result could be reached if given $\lambda = 3$.

4.3.1.4 Modification IV: Scout Phase search

In the standard ABC algorithm, the abandoned solution rebirth in the Scout Phase is carried out by generating random solutions uniformly sampled in the parameter search space using

Equation (2.6). This random feature is beneficial in the initial iterations; however, as the algorithm moves to later iterations the random generation of a fresh solution might be ineffective. Herein, a Gaussian solution rebirth strategy (also known as Gaussian mutation in evolutionary strategy [31]), taking into account the information associated within the population, is used in the Scout Phase search:

$$\Theta_{lj}^{sc} = \Theta_{lj} + \mathcal{N}(0, \varsigma_j^2) \tag{4.21}$$

where $j = 1, 2, \ldots, N_\theta$ and l denotes the abandoned solution; ς_j is the standard deviation (s.t.d.) of the j-th parameter among the population.

4.3.2 An overview of the BFGS method

In this work, we employ a well-known gradient-based Quasi-Newton approach, *v.i.z.*, the BFGS method, as our local search operator due to its great performance in solving unconstrained local optimization problems [89–92]. The starting point of this method is the initial guess provided by MABC, from which the BFGS is carried out zooming into the fine scale features of the parameters.

To solve the minimization problem in Equation (4.13), the BFGS method generates a sequence of line search iterative solutions $\{\boldsymbol{\theta}^k, k = 0, 1, 2, \ldots\}$ by

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \gamma_k \mathbf{d}_k \tag{4.22}$$

$$\mathbf{d}_k = -\mathbf{H}_k \nabla G(\boldsymbol{\theta}^k) \tag{4.23}$$

where γ_k is the step length, $\mathbf{d}_k$ is the search direction, $\mathbf{H}_k$ is the inverse Hessian approximation matrix expressed as

$$\mathbf{H}_{k+1} = \left[\mathbf{I} - \frac{\Delta\boldsymbol{\theta}^k \mathbf{q}_k^T}{\mathbf{q}_k^T \Delta\boldsymbol{\theta}^k}\right] \mathbf{H}_k \left[\mathbf{I} - \frac{\mathbf{q}_k (\Delta\boldsymbol{\theta}^k)^T}{\mathbf{q}_k^T \Delta\boldsymbol{\theta}^k}\right] + \frac{\Delta\boldsymbol{\theta}_k (\Delta\boldsymbol{\theta}^k)^T}{\mathbf{q}_k^T \Delta\boldsymbol{\theta}^k} \tag{4.24}$$

Here, $\Delta\boldsymbol{\theta}^k$ represents the solution change and $\mathbf{q}_k$ denotes the gradient change.

When the objective function is not analytically differentiable, the function gradient can be approximated using the central finite difference method:

$$\nabla G(\boldsymbol{\theta}^k) = \left[\frac{\partial G}{\partial \theta_1^k} \quad \cdots \quad \frac{\partial G}{\partial \theta_j^k} \quad \cdots \quad \frac{\partial G}{\partial \theta_n^k}\right]^T \tag{4.25}$$

with each component computed by

$$\frac{\partial G}{\partial \theta_j^k} \approx \frac{G(\theta_1^k, \ldots, \theta_j^k + \delta\theta_j, \ldots, \theta_n^k) - G(\theta_1^k, \ldots, \theta_j^k - \delta\theta_j, \ldots, \theta_n^k)}{2\delta\theta_j} \tag{4.26}$$

where $\delta\theta_j$ is an infinitesimal perturbation (e.g., 1×10^{-6}).

The pseudo code of the BFGS method is given in Algorithm 3. It is noteworthy that the selection of the step length γ_k in a standard line search is implemented according to the backtracking procedure and the Armijo rule [93]. The initial inverse Hessian approximation matrix $\mathbf{H}_0$ is set to be an identity matrix, namely, $\mathbf{H}_0 = \mathbf{I}$.

Algorithm 3 – The BFGS method

```
Given an admissible initial guess $\boldsymbol{\theta}^0$, convergence tolerance $\bar{\epsilon}$, maximum number of iterations $L_{max}$
and the initial inverse Hessian approximation matrix $\mathbf{H}_0$;
$k \leftarrow 0$;
while $(k \leq L_{max})$, $\left(\|\boldsymbol{\theta}^{k+1} - \boldsymbol{\theta}^k\| / \|\boldsymbol{\theta}^k\| > \varepsilon\right)$ and $\left(\|\nabla G(\boldsymbol{\theta}^k)\| > \varepsilon\right)$ do
  Compute the search direction $\mathbf{d}_k = -\mathbf{H}_k \nabla G(\boldsymbol{\theta}^k)$;
  Compute the solution change $\Delta\boldsymbol{\theta}^k = \gamma_k \mathbf{d}_k$ according to line search;
  Update the solution $\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + \Delta\boldsymbol{\theta}^k$;
  Compute the gradient change $\mathbf{q}_k = \nabla G(\boldsymbol{\theta}^{k+1}) - \nabla G(\boldsymbol{\theta}^k)$;
  Update the inverse Hessian approximation matrix using Equation (4.24);
  $k \leftarrow k + 1$;
end while
```

4.3.3 The hybrid MABC-BFGS scheme

A notable drawback of the BFGS method is its high dependency upon the initial guess. When the objective function is multimodal, a 'bad' initial guess might lead the BFGS method to be stagnated into a local minimum. To relieve this shortcoming, a 'good' solution found by a global optimizer, such as the MABC algorithm, could be used as the initial guess. The hybrid scheme between MABC and BFGS is presented as follows:

- *Algorithm Switching*: The BFGS method is added to MABC right after the Onlooker Phase search and is activated after the MABC reaches N_{start} iterations. Then the BFGS method is implemented for local solution search frequently, e.g., every other N_ω MABC iterations (N_ω denotes the frequency calling BFGS from MABC).

- *Solution Exchange*: The current best solution found by the MABC algorithm is taken as the initial guess ($\boldsymbol{\theta}^0$) for the BFGS method. When the stopping criteria are met in BFGS, its converged solution ($\boldsymbol{\theta}^{ls}$) is summarized as the local search representative solution. Afterwards, this solution is compared with its initial guess. If $\boldsymbol{\theta}^{ls}$ is "better" than $\boldsymbol{\theta}^0$ (solution is successfully improved by the local search), then $\boldsymbol{\theta}^0$ is replaced by $\boldsymbol{\theta}^{ls}$ and transfered back to the MABC algorithm; otherwise, the MABC algorithm restarts using $\boldsymbol{\theta}^0$ as its original value.

A flow chart of the proposed MABC-BFGS algorithm is given in Figure 4.5.

4.3.4 Mathematical benchmark function tests

In order to verify the effectiveness of the proposed MABC-BFGS in solving optimization problems, six mathematical benchmark functions with 30 unknown variables [88] were tested here. Herein, Bayesian inference is not employed. The MABC control parameters are set to be $N_{pop} = 60$, $N_{lim} = 200$ and $N_{max} = 5000$. The BFGS control parameters are $N_{start} = N_{max}/5 = 1000$, $L_{max} = 300$, $N_\omega = 500$, $\delta = 1 \times 10^{-8}$ and $\bar{\epsilon} = 1 \times 10^{-12}$. The stopping criteria for the MABC-BFGS are either when the objective function reaches 1×10^{-20} or when the total number of function evaluations reaches 2×10^5. Monte Carlo simulations were performed: each case was independently repeated 30 times so as to obtain

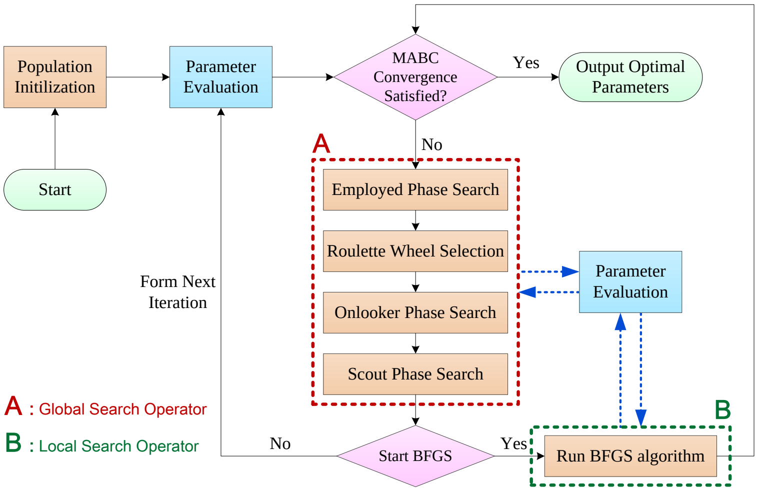


Figure 4.5: The flow chart of the proposed MABC-BFGS algorithm.

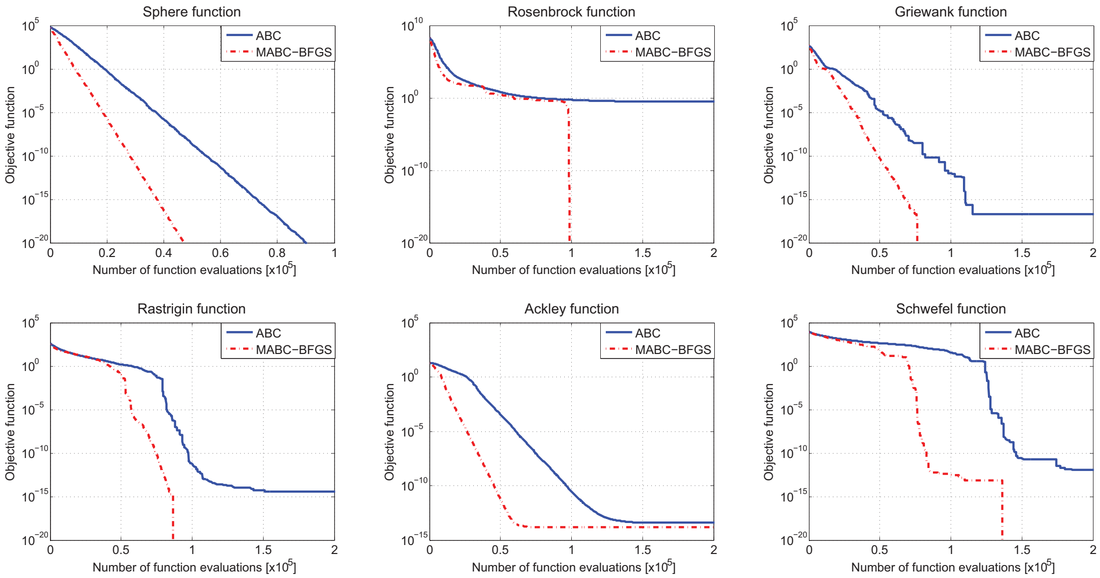


Figure 4.6: Convergence lines of six benchmark mathematical functions of 30 dimensions.

statistical results. Finally, the means are summarized as the representative solution.

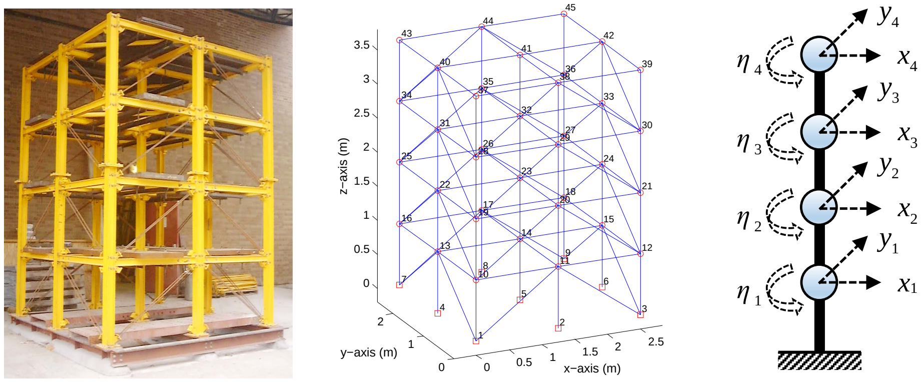
Figure 4.6 shows convergence lines of the six benchmark functions. It can be seen that the proposed MABC-BFGS algorithm has a superior convergence rate and a much better solution accuracy compared with the standard ABC algorithm. Due to its excellent performance in solving optimization problems, MABC-BFGS appears to be an attractive approach in application to Bayesian FEM updating.

It is noteworthy that the number of function evaluations is defined as the count of evaluating the objective functions. In ABC and MABC, it is identical to the number of fitness evaluations. In BFGS, it includes both the number of objective function evaluations and the number of gradient function evaluations.

4.4 Numerical example: phase I IASC-ASCE structure

To demonstrate the applicability and effectiveness of the proposed MABC-BFGS algorithm to probabilistic FEM updating and damage detection, the Phase I IASC-ASCE benchmark structure (see Figure 4.7(a)) is firstly considered in this section.

The structure is a 4-storey, 2-bay by 2-bay steel frame, with dimensions of $2.5 \times 2.5 \times 3.5$



(a) Benchmark structure (b) FEM (c) 12 DOF model

Figure 4.7: IASC-ASCE benchmark structure and its FEM.

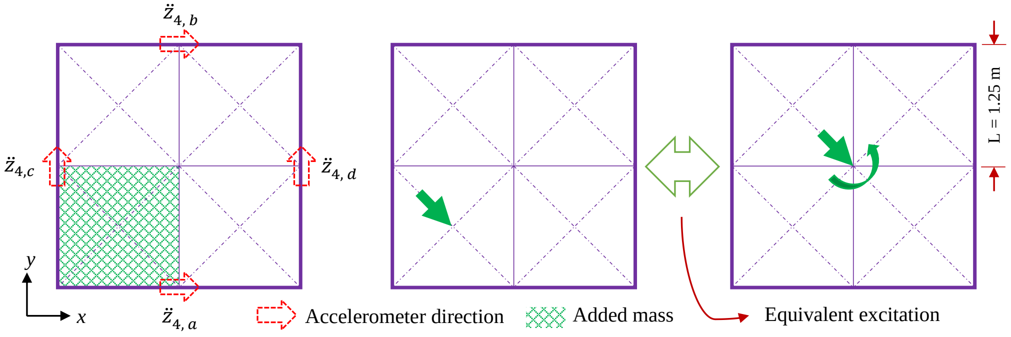


Figure 4.8: Plan view of the 4th floor of the IASC-ASCE benchmark structure: sensor placement, mass increase location and force excitation.

m as shown in Figure 4.7(a). The corresponding FEM considering rigid floor effect is shown in Figure 4.7(b). In the original Phase I task, several loading and damage cases were considered. In this work, we only considered the identification problems presented in Structural Case 4: the structure is asymmetric, due to an increasing slab mass of 150 kg on the top floor. As the input excitation, a concentrated force, modeled as a filtered Gaussian white noise, is applied on the 4th floor. Figure 4.8 gives the plan view of the 4th floor, which is analogous to other three floors. Accelerometers are horizontally placed on the slab edges. Other detailed description of this structure can be found in the work by [94].

The 12 DOF model is a 3D linear shear type system with rigid floors as shown in Figure 4.7(c). The mass and stiffness matrices corresponding to the DOFs of each floor can be written as:

$$\mathbf{M}^{(i)} = \begin{bmatrix} M_{11}^{(i)} & 0 & M_{13}^{(i)} \\ 0 & M_{11}^{(i)} & M_{23}^{(i)} \\ M_{13}^{(i)} & M_{23}^{(i)} & M_{33}^{(i)} \end{bmatrix} \text{ and } \mathbf{K}^{(i)} = \begin{bmatrix} K_{11}^{(i)} & 0 & K_{13}^{(i)} \\ 0 & K_{22}^{(i)} & K_{23}^{(i)} \\ K_{13}^{(i)} & K_{23}^{(i)} & K_{33}^{(i)} \end{bmatrix} \tag{4.27}$$

where the superscript (i) is the floor index ($i = 1, 2, 3, 4$). The standard modal damping is used here with a single damping ratio $\xi = 5\%$ for all the modes. Hence, we have 37 parameters to be identified in total, namely,

$$\boldsymbol{\theta} = \left\{ \xi, M_{11}^{(i)}, M_{13}^{(i)}, M_{23}^{(i)}, M_{33}^{(i)}, K_{11}^{(i)}, K_{13}^{(i)}, K_{22}^{(i)}, K_{23}^{(i)}, K_{33}^{(i)} \right\}_{i=1,2,3,4} \tag{4.28}$$

To link the connection between the measured horizontal accelerations and the 12 DOF model response, the following equations can be used:

$$\ddot{\mathbf{y}}_{i,x}^{12DOF} = \frac{\ddot{\mathbf{y}}_a + \ddot{\mathbf{y}}_b}{2}; \quad \ddot{\mathbf{y}}_{i,y}^{12DOF} = \frac{\ddot{\mathbf{y}}_c + \ddot{\mathbf{y}}_d}{2}; \quad \ddot{\mathbf{y}}_{i,\eta}^{12DOF} = \frac{\ddot{\mathbf{y}}_a - \ddot{\mathbf{y}}_b}{4l} + \frac{\ddot{\mathbf{y}}_d - \ddot{\mathbf{y}}_c}{4l} \tag{4.29}$$

where the superscript $12DOF$ denotes the 12 DOF model; the subscript i is the floor index; the subscripts x, y and η denote three directions as shown in Figure 4.7(c); the subscripts a, b, c and d represent four sensor locations as shown in Figure 4.8.

The time histories of the responses at the sensor locations are generated using the latest version of the *datagen* MATLAB® program which can be downloaded from `http://bc029049.cityu.edu.hk/asce.shm/phase1/ascebenchmark.asp`. The measured time histories of system response are presented in the form of accelerations recorded by a limited number of accelerometers. A 5-sec long signal with the sampling frequency of 1 kHz is used in the identification process ($N = 5 \times 10^3$). To validate the applicability of the proposed approach to model updating, only input and partial output measurements are considered here: acceleration time histories are available at the 2nd and the 4th floors ($N_o = 6$). Furthermore, to test the effect of measurement noise on parameter identification, noise pollution scenarios have been considered. The noisy input-output measurements are generated by adding a zero mean Gaussian white noise sequence, whose root-mean-square (RMS) is a certain percentage of the RMS of the correct signal, to the correct (noise free) signal. Both the noise free and the 20% RMS noise cases are considered in this example. The numerical analyses are programmed in MATLAB® (The MathWorks, Inc., MA, USA) on a standard Intel (R) Core (TM) i5-3570K 3.40 GHz PC with 16G RAM.

Table 4.1: Prior distribution of system parameters (mean and c.o.v. values)

Parameters	$M_{II}^{(i)}$	$M_{IJ}^{(i)}$	$K_{II}^{(i)}$	$K_{IJ}^{(i)}$	ξ
Mean	2.5	0	1.5	0	0.03
c.o.v.	20%	20%	30%	30%	30%

Note that $i = 1, 2, 3, 4$; $I, J = 1, 2, 3$ and $I \neq J$. The mean values of diagonal masses are expressed in 10^3 kg, polar moments of inertia in 10^3 kgm^2, tridiagonal masses in 10^3 kgm, diagonal stiffnesses in 10^8 Nm^{-1}, torsional stiffnesses in 10^8 Nm, and tridiagonal stiffnesses in 10^8 N.

The prior distributions of masses, damping ratio and stiffnesses are assumed to be Gaussian with the mean and c.o.v. values shown in Table 4.1. It is noteworthy that a confident interval can be applied to select samples so as to prevent sampling negative values of parameters. Herein, we assume the prior PDF of the prediction error covariance σ is unknown a priori. Thus Equation (4.14) can be used to approximate its posterior value $\hat{\sigma}$. The MABC-BFGS parameters for all independent runs used here are: $N_{pop} = 30$, $N_{lim} = 200$, $N_{max} = 500$, $N_{start} = 300$, $L_{max} = 300$, $N_\omega = 50$, $\delta = 1{\times}10^{-6}$ and $\bar{\epsilon} = 1{\times}10^{-8}$.

A single set of measurement data was considered here. Due to the zero components in the reference matrices, the relative error of a matrix is defined:

$$E_{ij} = \frac{\left|\hat{Q}_{ij} - Q_{ij}\right|}{\max\left\{|Q_{ij}|\right\}_{i=1,\ldots,12;\ j=1,\ldots,12}} \times 100\% \tag{4.30}$$

where $\hat{Q}_{ij}$ and Q_{ij} represent the identified and the true elements, in the mass or the stiffness matrix, respectively.

4.4.1 Updating of the undamaged “healthy” structure

The parameters representing the updated/posterior distributions of the structural parameters to be identified, for the case of 20% RMS noise, are presented in Table 4.2. The identification results for the noise free case are not listed here because the identified values are almost the exact values, e.g., with a maximum relative error less than 1×10^{-6} (see the parameter convergence lines in Figure 4.9). It can be observed from Table 4.2 that the probabilistic model updating results are excellent, with the identified mean values of masses and stiffnesses being very close to the true values. Reasonably small standard deviations are also obtained. The identified mean value of damping ratio ξ, in the 20% RMS noise case, is also quite accurate (e.g., $\xi = 4.94\%$) with the relative error of 1.2%. It is noteworthy that the prediction error covariance in the noise free case is 1.3362×10^{-9} m/sec^2 (the true value is zero), while the identified value in the 20% RMS noise case is 0.3095 m/sec^2 (the true value is 0.3204 m/sec^2).

Figures 4.9 and 4.10 show the plots of both the objective function and some typical system parameters as a function of the number of iterations in the model updating of the IASC-ASCE benchmark structure, for the noise free and the 20% RMS noise case, respectively. It can be seen that the identified mean value of each parameter converge

Table 4.2: Statistical estimates of the mass/stiffness parameters with 20% RMS noise

Parameter	True	Identified	Parameter	True	Identified
$M_{11}^{(1)}$	3.4524	3.4996 (0.018)	$K_{33}^{(1)}$	2.3202	2.3472 (0.016)
$M_{33}^{(1)}$	3.8194	3.8446 (0.027)	$K_{13}^{(1)}$	0	0.0201 (0.011)
$M_{13}^{(1)}$	0	0.0244 (0.020)	$K_{23}^{(1)}$	0	−0.0150 (0.010)
$M_{23}^{(1)}$	0	−0.0181 (0.027)	$K_{11}^{(2)}$	1.0660	1.0840 (0.007)
$M_{11}^{(2)}$	2.6524	2.7018 (0.018)	$K_{22}^{(2)}$	0.6790	0.6916 (0.004)
$M_{33}^{(2)}$	2.9861	2.9709 (0.022)	$K_{33}^{(2)}$	2.3202	2.3234 (0.015)
$M_{13}^{(2)}$	0	0.0237 (0.015)	$K_{13}^{(2)}$	0	0.0079 (0.010)
$M_{23}^{(2)}$	0	−0.0357 (0.014)	$K_{23}^{(2)}$	0	−0.0218 (0.010)
$M_{11}^{(3)}$	2.6524	2.6834 (0.014)	$K_{11}^{(3)}$	1.0660	1.0812 (0.006)
$M_{33}^{(3)}$	2.9861	2.9999 (0.020)	$K_{22}^{(3)}$	0.6790	0.6896 (0.004)
$M_{13}^{(3)}$	0	0.0474 (0.016)	$K_{33}^{(3)}$	2.3202	2.3148 (0.016)
$M_{23}^{(3)}$	0	−0.0550 (0.019)	$K_{13}^{(3)}$	0	0.0226 (0.010)
$M_{11}^{(4)}$	1.9599	1.9933 (0.007)	$K_{23}^{(3)}$	0	−0.0261 (0.009)
$M_{33}^{(4)}$	2.2131	2.2405 (0.011)	$K_{11}^{(4)}$	1.0660	1.0831 (0.005)
$M_{13}^{(4)}$	0.0938	0.1199 (0.008)	$K_{22}^{(4)}$	0.6790	0.6882 (0.003)
$M_{23}^{(4)}$	−0.0938	−0.1249 (0.008)	$K_{33}^{(4)}$	2.3202	2.3362 (0.015)
$K_{11}^{(1)}$	1.0660	1.0821 (0.005)	$K_{13}^{(4)}$	0	0.0220 (0.008)
$K_{22}^{(1)}$	0.6790	0.6891 (0.003)	$K_{23}^{(4)}$	0	−0.0205 (0.007)

Note that the parameter units are identical to those in Table 4.1. The values in the parentheses are the standard deviations.

to the correct value quite rapidly, after about 1×10^4 function evaluations (each function evaluation takes about 0.01 sec CPU time). From these figures, it appears that the MABC is able to find an "effective" initial point for the BFGS and that the BFGS converges quite rapidly. Hence, the iteraction between MABC and BFGS is proven to be effective and powerful. The proposed MABC-BFGS algorithm is able to overcome the drawbacks of both relatively slow convergence of MABC and high initial guess dependency of BFGS, while taking advantages of MABC great global search ability and of BFGS fast convergence.

Figure 4.11 depicts the identified probability density functions (PDFs) of some typical system parameters, which were determined using Equation (4.15). It can be seen from Figure 4.11 that the identified system parameters are quite close to the correct values, yet with

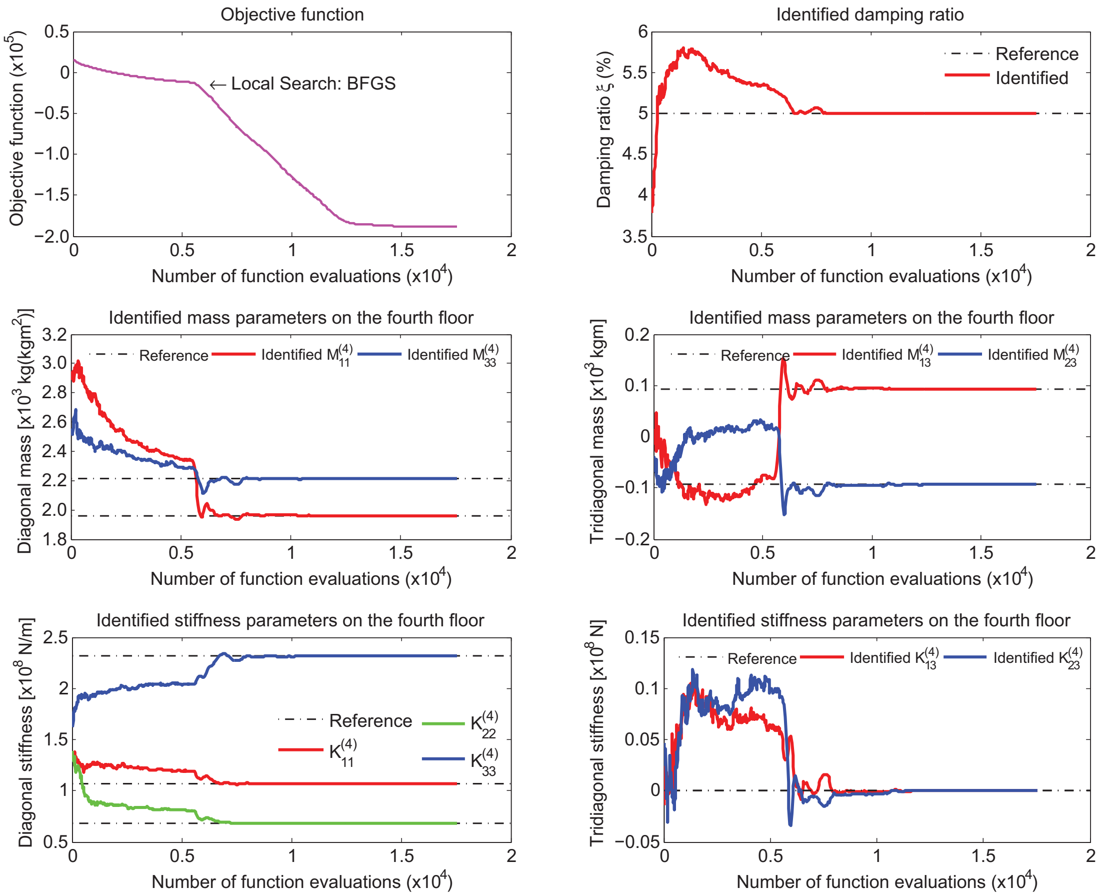


Figure 4.9: Convergence lines of some typical identified parameters (mean values) of the IASC-ASCE benchmark structure without measurement noise.

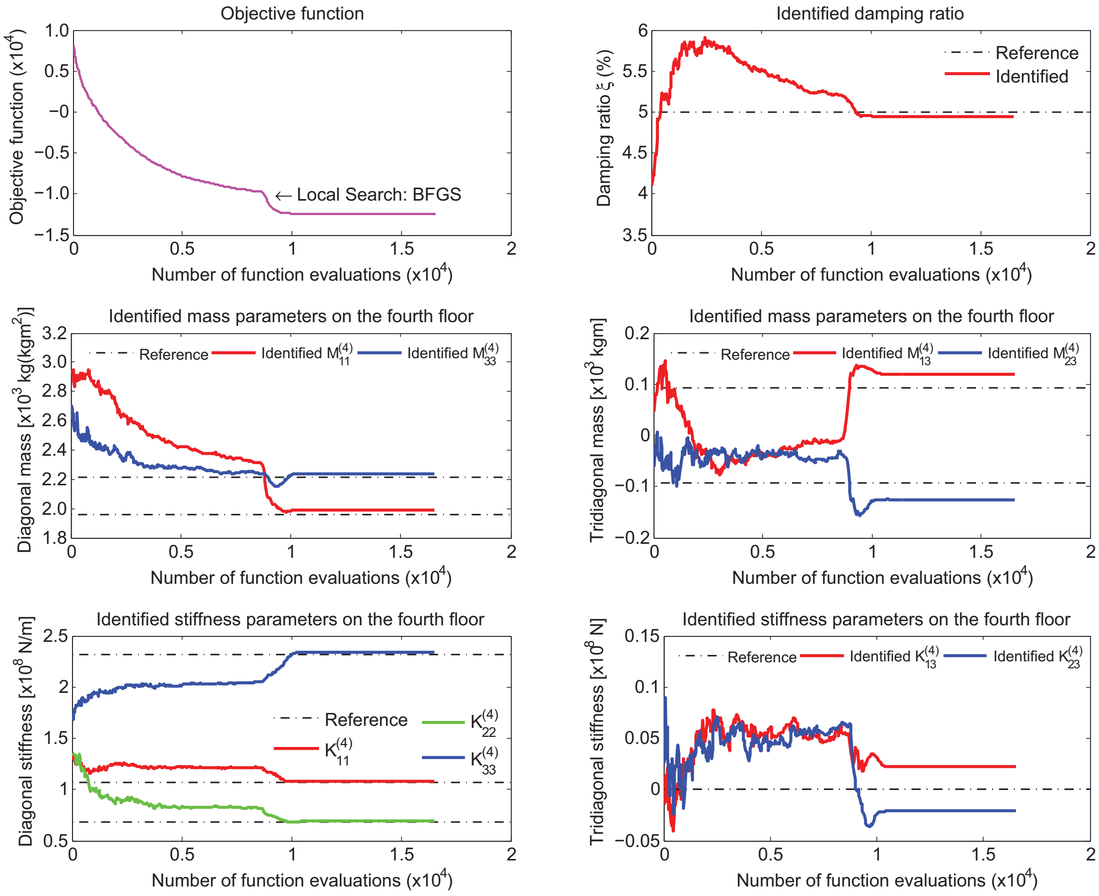


Figure 4.10: Convergence lines of some typical identified parameters (mean values) of the IASC-ASCE benchmark structure with 20% RMS measurement noise.

small standard deviations. This demonstrates the robustness of the proposed algorithm against measurement noise. Figure 4.12 shows the relative error plots of the identified mean values of mass and stiffness matrices in the 20% RMS noise case. It can be seen that, in general, the errors are very small with the maximum error less than 2%. The gratifying identification results proves the successful application of the proposed methodology to structural FEM updating, even in the presence of measurement noise. Figure 4.13 presents typical pairwise plots of the prior *vs*. the posterior samples for some mass and stiffness parameters. It can be seen that even if the knowledge of the prior PDF is not well known, the system parameters can still be updated quite well, with the updated/posterior parameter samples being accurately located in the correct parameter region.

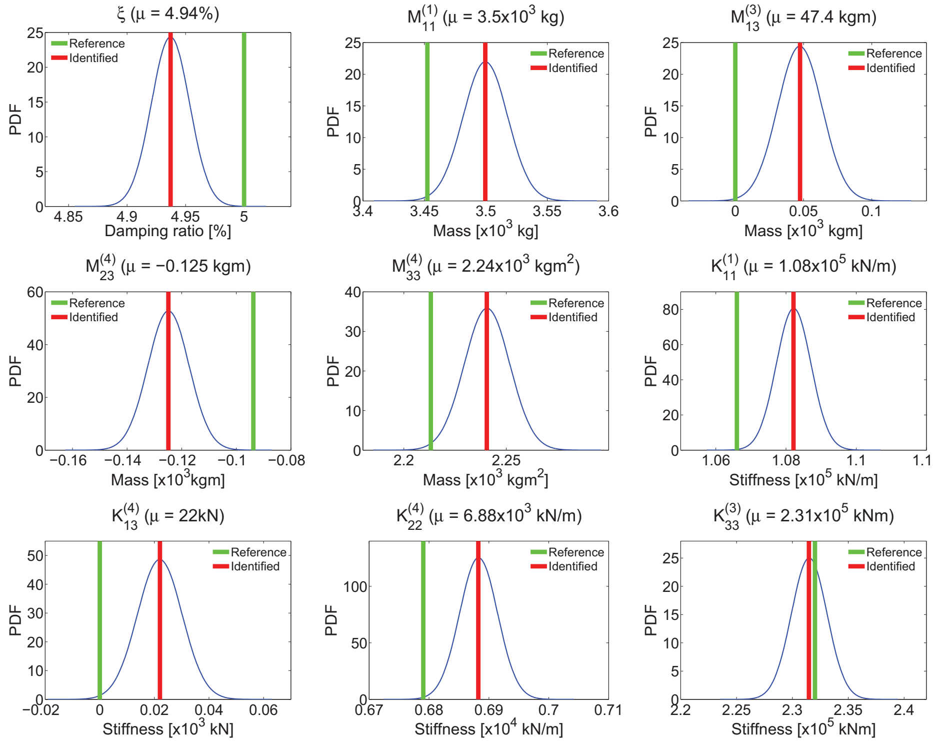
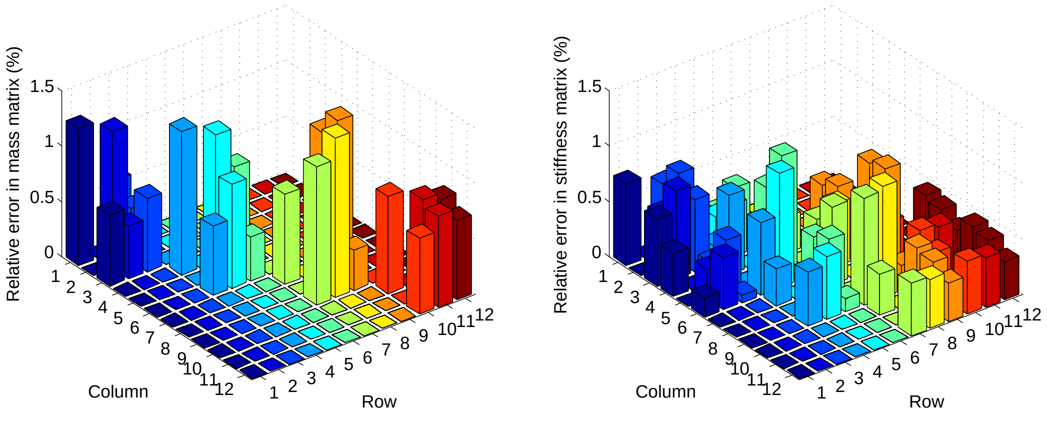


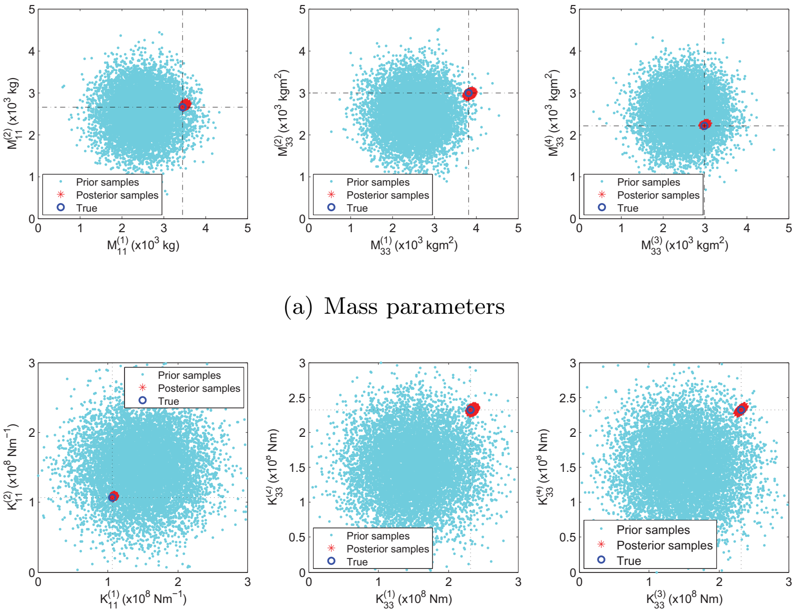
Figure 4.11: The identified posterior PDFs of some typical parameters of the IASC-ASCE benchmark structure with 20% RMS measurement noise.



(a) Relative error of mass matrix

(b) Relative error of stiffness matrix

Figure 4.12: Relative error of the identified mass and stiffness matrices (mean values) of the IASC-ASCE benchmark structure with 20% RMS measurement noise.



(a) Mass parameters

(b) Stiffness parameters

Figure 4.13: Pairwise prior *vs.* posterior samples for some mass and stiffness parameters in the identification of the IASC-ASCE benchmark structure.

4.4.2 Damage detection

Phase I of the benchmark structure also includes the task of damage detection. In this example, we consider six damage scenarios which are summarized here (see Figure 4.14):

- Damage Case 1: all braces of the 1st floors are completely broken (removed).
- Damage Case 2: four braces of the 3rd floor, two acting in x-direction and two in y-direction, are completely broken.
- Damage Case 3: one brace of the 1st floor acting in y-direction is completely broken.
- Damage Case 4: two braces, one of the 1st floor acting in y-direction and one of the 3rd floor in x-direction, are completely broken.
- Damage Case 5: two columns, one of the 2nd floor and one of the 4th floor, are completely broken.
- Damage Case 6: one brace of the 1st floor acting in y-direction is cut to 2/3 of its original cross-sectional area.

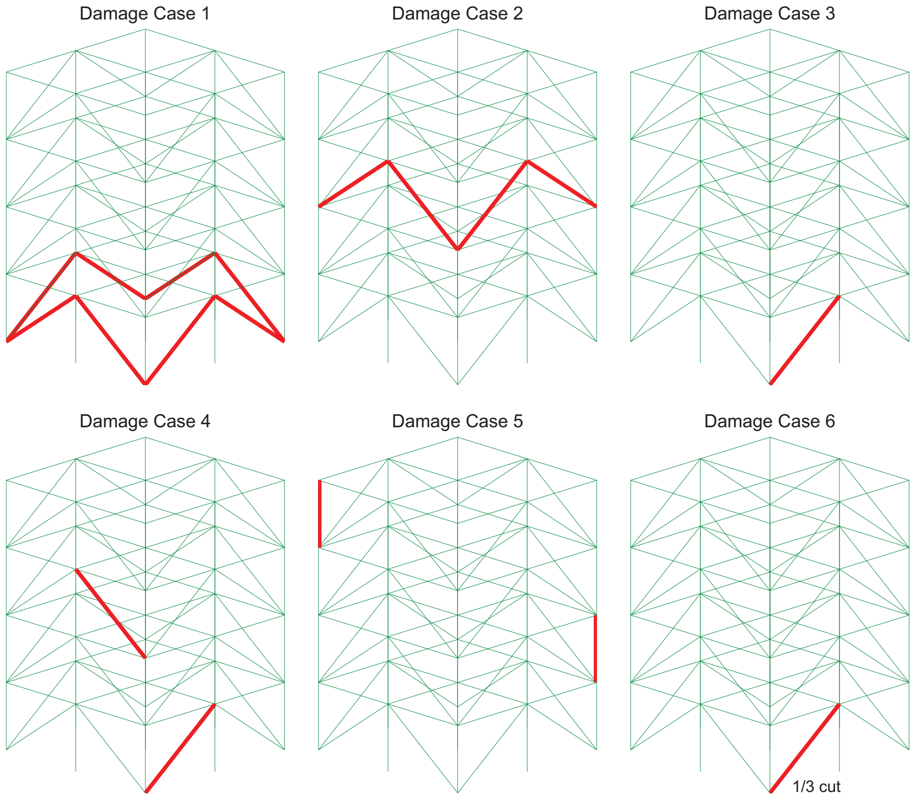


Figure 4.14: IASC-ASCE benchmark structure damage scenarios. The green lines represent intact elements and the red lines represent damaged elements (either removed or cut).

Only the case of 20% RMS measurement noise is carried out here for the purpose of damage detection. Damages are located and quantified by comparing two identified states of a consistent model (e.g., prior to vs. after the occurrence of damage). The parameters of the undamaged structure were first identified (see Section 4.4.1) and then were considered as the “reference mean values” in regard to the damage states. The damage index d_i is defined to quantify the damage level:

$$d_i = \frac{k_i^u - k_i^d}{k_i^u} \times 100\% \tag{4.31}$$

where i denotes the stiffness parameter index ($i = 1, 2, \ldots$); k_i^u represents the identified mean values of the undamaged stiffness while k_i^d represents the identified mean values of the damaged stiffness.

Figure 4.15 summarizes the identified damage levels (e.g., stiffness reductions) of the six damage scenarios. It is seen that the identified damage levels (either the lateral or the torsional stiffness reduction) are quite close to the reference values. It is obvious that the damages can be accurately located and quantified.

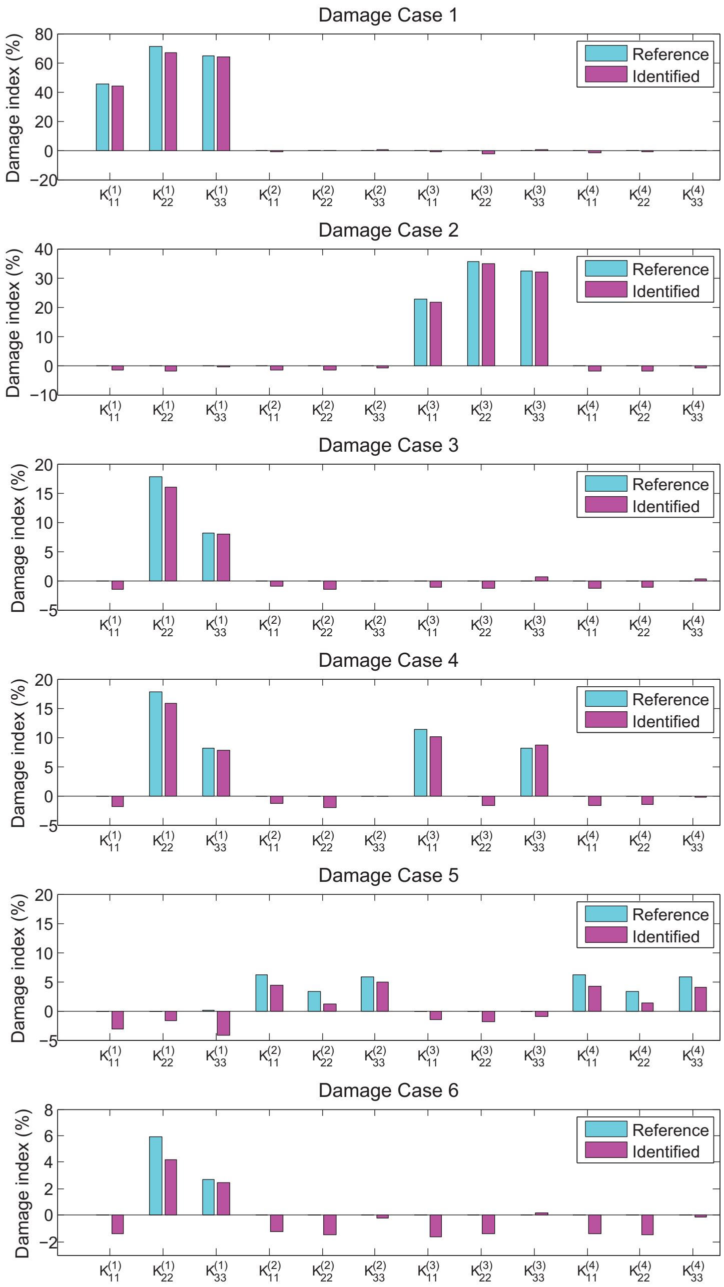
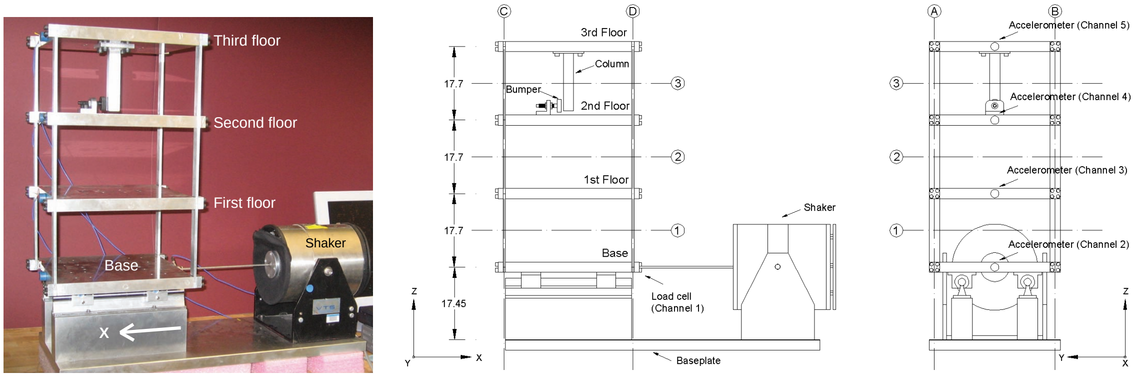


Figure 4.15: Identified damage levels of the IASC-ASCE benchmark structure.

4.5 Experimental example: LANL 3-storey frame structure

In this example, experimental data sets of a three-storey frame structure shown in Figure 4.16 are used to validate the proposed method for both FEM updating and structural damage detection.

The experiment on the test bed structure was conducted by the structural health monitoring group at the Los Alamos National Laboratory (LANL). The structure consists of aluminum columns and plates assembled using bolted joints. The structure slides on rails that allow movement in the x-direction only, which simulates the case of shaking table test with force excitations at the base floor. An accelerometer was attached at the centerline of each floor, including the base. The floors are made of aluminum plates (30.5 × 30.5 × 2.5 cm). At each floor, four aluminum columns (17.7 × 2.5 × 0.6 cm) are connected to the top and the bottom plates, essentially forming a four-DOF system subjected to base excitations. Moreover, a column (15.0 × 2.5 × 2.5 cm) is suspended from the center of the top floor and, together with a bumper mounted on the next floor (see Figure 4.16), is used to simulate damages. An electrodynamic shaker was used to provide a lateral excitation to the base floor along the centerline of the structure. Other detailed descriptions of this structure as well as of the data acquisition system can be found in the technical report by [95].



(a)

(b)

Figure 4.16: The LANL test structure setup: (a) three-storey frame structure with shaker and (b) basic dimensions (all dimensions are in cm).

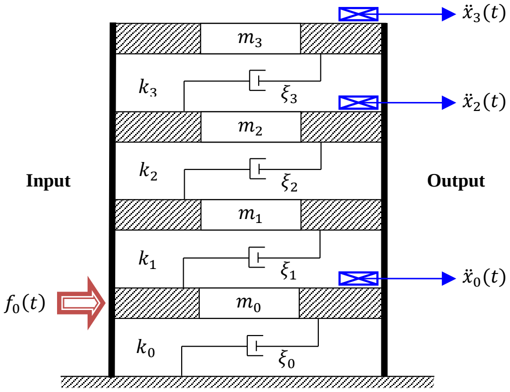


Figure 4.17: Shear type building model of the LANL test bed structure.

It is noteworthy that we model the structure as a 4-DOF shear-type system (e.g., the base is regarded as the first floor where an input force is applied) as shown in Figure 4.17. Two non-physical parameters, e.g., stiffness k_0 and damping ratio ξ_0, of the first DOF are presented to approximate the sliding frictions between the base floor and the rails. Therefore, we update the 4-DOF model in the context of input-output identification of 12 parameters (e.g., four lumped masses, four damping ratios and four stiffness parameters). The rigid body motion of the structure can be well simulated by adjusting k_0 and ξ_0. Afterwards, we carried out damage detection through identification of the 4-DOF model in regard to different damage states (e.g., the damage extent is defined as a relative change of the stiffness value).

To test the effectiveness of the proposed approach applied to model updating with partial instrumentation, only the recorded acceleration time histories at the base, first and top floors were employed in the identification process ($N_o = 3$) (see Figure 4.17). Signals, sampled at 320 Hz for a duration of 25.6 sec, were used at each floor. Since the accelerometers are located at the centerline of each floor, they are insensitive to the structural torsional modes. Classic modal damping was used to describe the energy dissipation mechanism of the structure. Nine recorded experimental scenarios were investigated here, including the undamaged state and eight damage states as shown in Table 4.3. For each scenario, the experiment was repeated 50 times and the corresponding system response was recorded. However, we herein only employed 15 data sets for identification in each state.

Table 4.3: Nine recorded experimental scenarios of the LANL three-storey frame structure

State	Damage	Damage level	Description (damage location)
0	No	0	Baseline condition (healthy structure)
1	No	0	Added mass (1.2 kg) at the base
2	No	0	Added mass (1.2 kg) on the first floor
3	Yes	21.88%	87.5% stiffness reduction in column 1BD
4	Yes	43.75%	87.5% stiffness reduction in column 1AD and 1BD
5	Yes	21.88%	87.5% stiffness reduction in column 2BD
6	Yes	43.75%	87.5% stiffness reduction in column 2AD and 2BD
7	Yes	21.88%	87.5% stiffness reduction in column 3BD
8	Yes	43.75%	87.5% stiffness reduction in column 3AD and 3BD

The total lumped masses are assumed to be known in the identification process (e.g., $M_{total} = 26$ kg for states 0 and 3–8, and $M_{total} = 27.2$ kg for states 1 and 2). Herein, we have 12 parameters to be updated in total, namely, $\boldsymbol{\theta} = \{m_0, m_1, m_2, m_3, \xi_0, \xi_1, \xi_2, \xi_3, k_0, k_1, k_2, k_3\}$. The parameters of the healthy structure were first identified as the "reference values" in regard to the damage states. The damage index d_i defined in Equation (4.31) is employed to quantify the damage level of each floor.

In the identification process, we assume the prediction error covariance σ follows a prior inverse Gamma distribution with the hyperparameters $\alpha = \beta = 1$ m/s^2. Thus Equation (4.11) is used to calculate its posterior value $\hat{\sigma}$. The prior PDFs for masses, damping ratios and stiffnesses are selected to be independent distributions, namely, m_i, ξ_i and k_i $(i = 0, \ldots, 3)$ follow the normal distribution with means equal to their nominal values $\bar{m}_i = 6.5$ kg $(i = 0, \ldots, 3)$, $\bar{\xi}_i = 3\%$ $(i = 0, \ldots, 3)$ and $\bar{k}_0 = 10$ N/m, $\bar{k}_i = 500$ kN/m $(i = 1, \ldots, 3)$, and the corresponding c.o.v. of 10% for masses and 30% for damping and stiffness parameters, respectively. The MABC-BFGS parameters used here are identical to those used in Section 4.4.

Table 4.4 presents the statistical results for physical parameter estimates of the test bed structure using 15 data sets. For all damage states, it is observed that the values of c.o.v. are reasonably small for mass and stiffness parameters. Nevertheless, posterior damping

Table 4.4: Statistical results for physical parameter estimates of the LANL structure

State	$\boldsymbol{\theta}$	m_0	m_1	m_2	m_3	ξ_1	ξ_2	ξ_3	k_1	k_2	k_3
0	Mean	7.34	7.50	6.09	5.07	6.20%	0.90%	1.00%	448.31	419.04	327.16
	c.o.v.	0.7%	0.8%	1.0%	0.9%	2.9%	6.5%	14.0%	0.8%	0.4%	1.0%
1	Mean	8.46	7.58	6.08	3.88	6.70%	0.80%	0.70%	440.67	418.05	331.10
	c.o.v.	0.8%	0.8%	1.3%	0.9%	2.7%	6.7%	9.0%	0.8%	0.5%	1.0%
2	Mean	7.32	9.10	5.94	3.63	6.00%	1.20%	0.90%	456.90	400.61	324.59
	c.o.v.	0.7%	0.8%	1.0%	1.2%	2.7%	7.2%	17.3%	0.7%	0.7%	1.0%
3	Mean	7.18	8.17	5.86	4.79	6.60%	1.10%	0.40%	368.51	411.32	318.78
	c.o.v.	0.8%	1.0%	1.2%	1.1%	3.6%	6.9%	10.3%	1.0%	1.0%	1.2%
4	Mean	6.94	8.92	5.70	4.44	8.10%	1.00%	0.30%	290.83	424.14	290.68
	c.o.v.	0.5%	0.8%	0.8%	1.2%	4.6%	4.8%	13.0%	0.7%	1.0%	1.1%
5	Mean	7.21	7.82	6.34	4.63	6.60%	1.00%	1.50%	465.60	342.93	313.99
	c.o.v.	0.8%	0.7%	0.9%	1.0%	4.1%	8.1%	9.3%	0.8%	0.6%	1.1%
6	Mean	7.15	7.81	6.19	4.85	7.23%	0.88%	6.30%	489.73	262.19	296.70
	c.o.v.	0.74%	1.57%	1.01%	2.70%	4.09%	5.38%	12.16%	1.41%	1.73%	1.36%
7	Mean	7.46	7.25	6.29	5.00	6.50%	1.10%	0.60%	438.85	417.71	259.50
	c.o.v.	0.7%	0.8%	0.8%	1.1%	3.0%	5.4%	10.5%	1.0%	0.4%	1.1%
8	Mean	7.48	7.37	6.56	4.58	6.40%	1.50%	0.50%	453.01	427.61	183.42
	c.o.v.	0.7%	0.9%	0.5%	1.5%	4.6%	5.6%	10.1%	0.9%	0.7%	1.4%

Note that the unit for mass is kg and the unit for stiffness parameters is kN/m.

parameters have larger uncertainties. The identified mean frequencies are summarized in Table 4.5. Figure 4.18 shows the identified posterior PDFs of masses, damping ratios and stiffness parameters, determined by Equation (4.16), of the healthy test bed structure. It can be seen that the standard deviations are small thanks to the robustness of the proposed model updating algorithm. Figure 4.19 illustrates the identified mean mass normalized mode shapes for the healthy state and three damage states. It is seen that mode shapes change due to stiffness reductions in the damage states. In addition, an obvious rigid body motion is observed in the first mode.

Figure 4.20 depicts the predicted acceleration time histories and their power spectra density functions on the base and the top floors, in comparison with the recorded signal from experiments. It is observed that those two sets of signals have a quite good agreement.

Table 4.5: The identified mean frequencies

State	Second Mode	Third Mode	Fourth Mode
0	31.09 (0.10%)	55.05 (0.05%)	72.23 (0.12%)
1	30.19 (0.12%)	53.88 (0.04%)	71.74 (0.06%)
2	30.99 (0.13%)	54.21 (0.04%)	69.72 (0.09%)
3	30.59 (0.26%)	51.79 (0.04%)	70.30 (0.03%)
4	29.97 (0.22%)	47.52 (0.07%)	69.08 (0.05%)
5	30.05 (0.21%)	55.30 (0.07%)	68.10 (0.09%)
6	27.40 (0.80%)	54.83 (0.36%)	65.22 (0.83%)
7	29.88 (0.21%)	51.62 (0.06%)	70.48 (0.05%)
8	28.60 (0.23%)	47.88 (0.09%)	69.42 (0.05%)

Note that the unit of the frequencies is Hz. The values in the parentheses represent c.o.v..

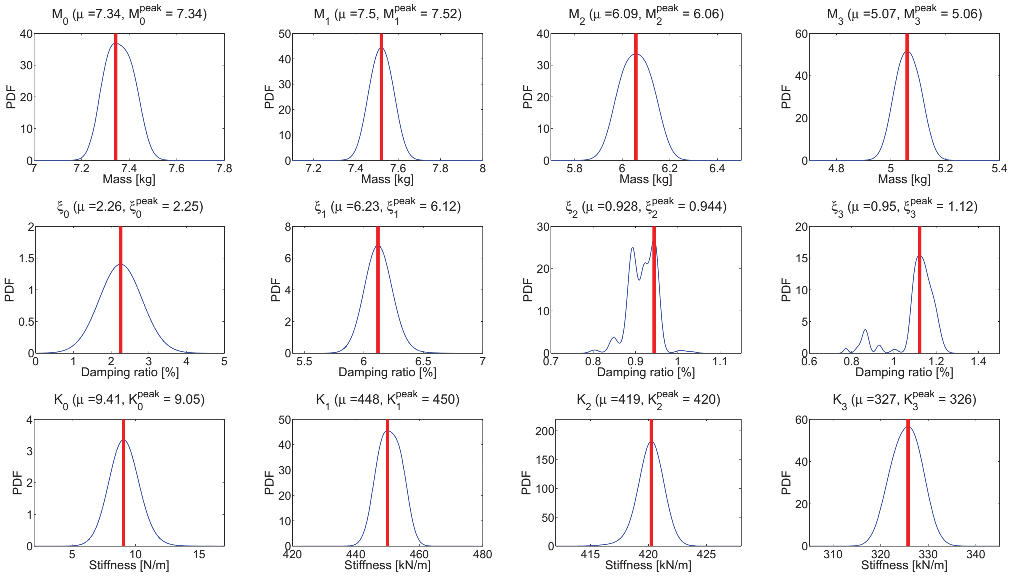


Figure 4.18: The identified posterior PDFs of damping and stiffness parameters of the LANL three-storey structure.

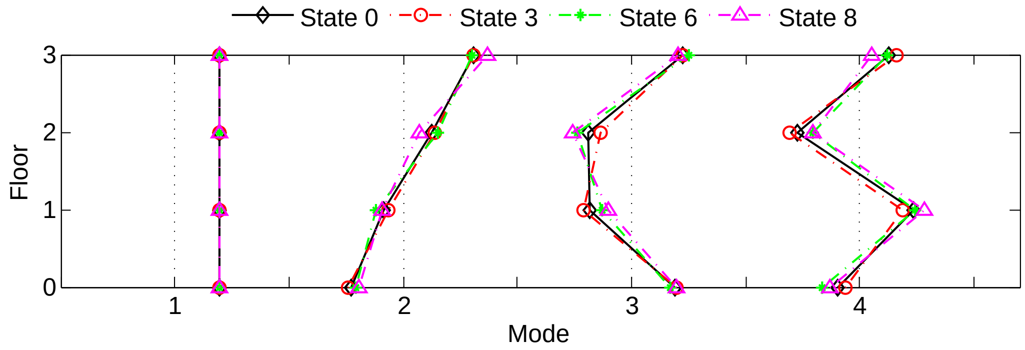
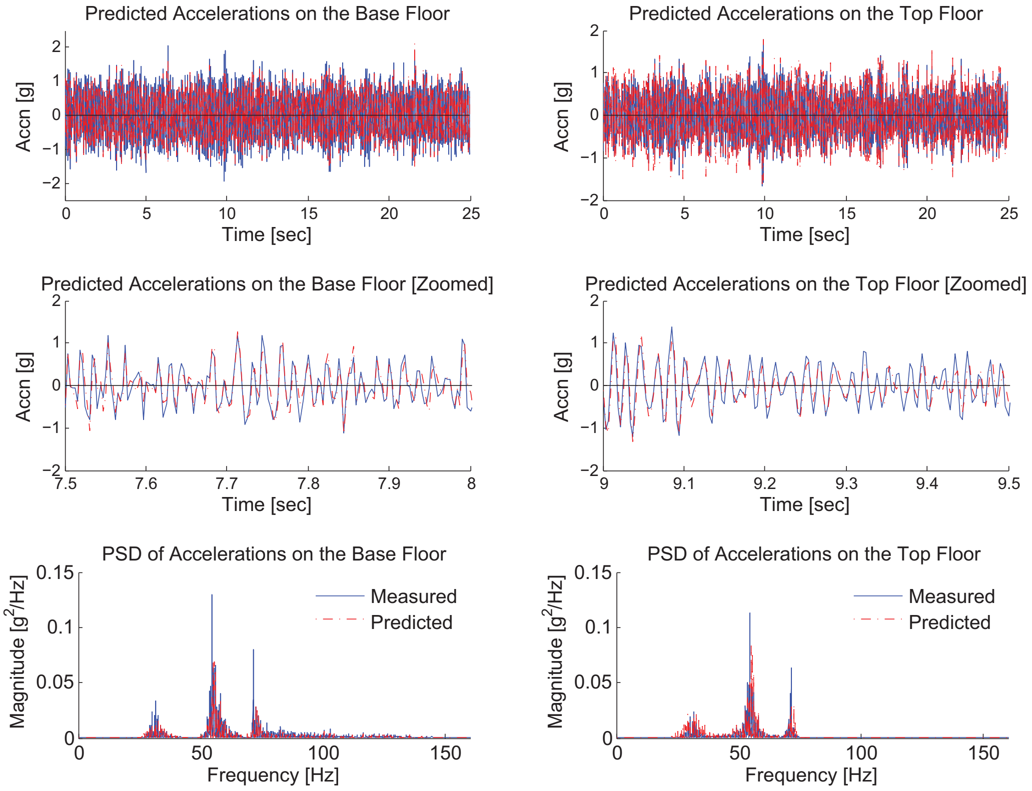


Figure 4.19: The identified mean mass normalized mode shapes for the healthy state and three damage states.



(a) Base floor

(b) Top floor

Figure 4.20: Predicted accelerations on the top floor of the LANL three-storey structure.

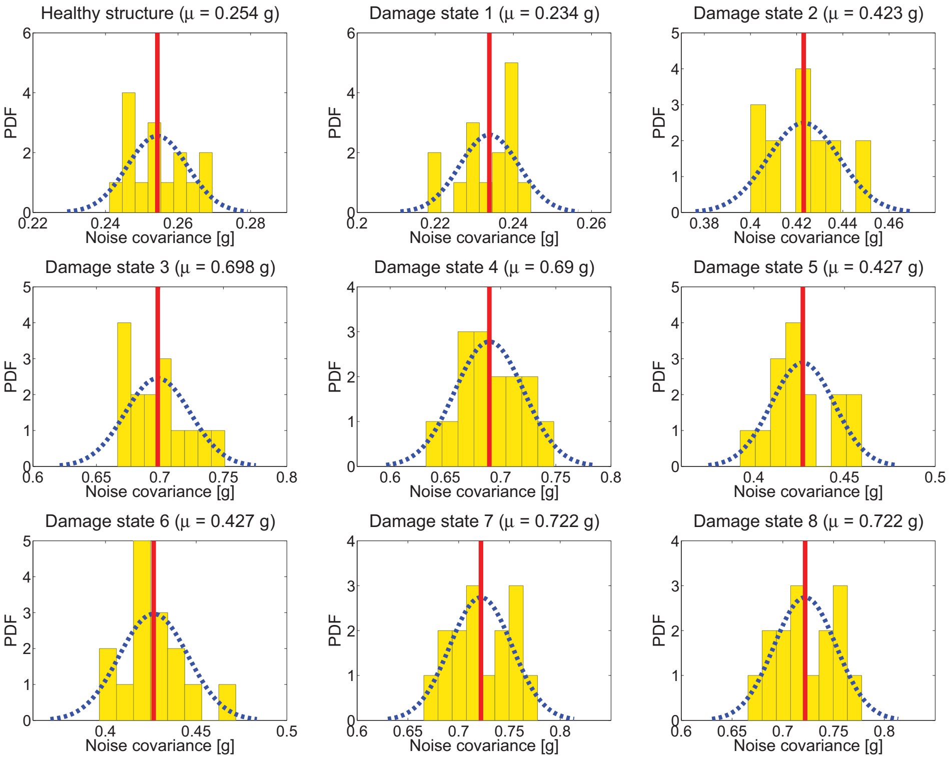


Figure 4.21: Some typical identified PDFs of the prediction error covariance σ in the identification of the LANL three-storey structure.

The identified PDFs of the prediction error covariance σ are summarized in Figure 4.21. The distributions of σ of different damage states show quite small variations accounting for the algorithm robustness.

Figure 4.22 presents the identified damage indices at each floor of different damage states. It is obvious that the damage assessment result is fairly good, e.g., the identified damage extents are close to the reference (theoretical) values. The locations as well as the levels of damages can be well identified with slight errors.

Furthermore, Figure 4.23 shows the identified posterior PDFs of damaged stiffnesses corresponding to six damage states (e.g., states 3–8) of the test bed structure. Obvious stiffness reductions are obtained. Narrow bands of standard deviations can be observed. The probabilities of stiffness reductions can be quantified after integrating the PDFs to obtain cumulative distribution functions, which can then be used for probabilistic damage assessment of structures.

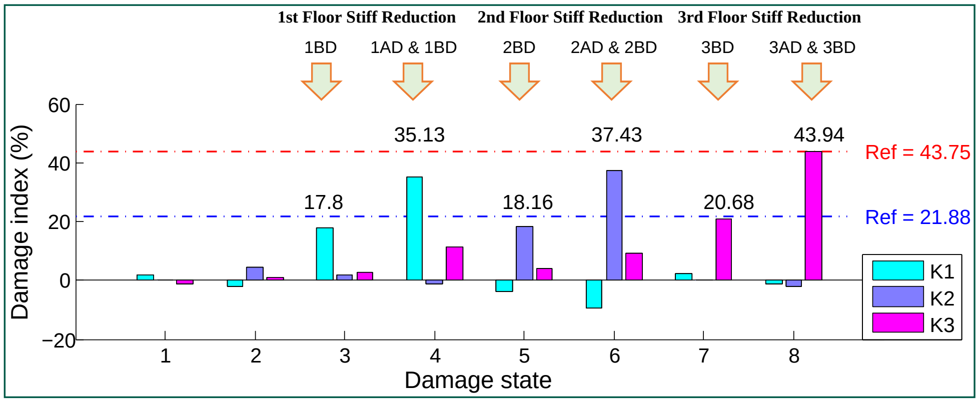


Figure 4.22: Identified damage indices corresponding to eight damage states of the LANL three-storey structure.

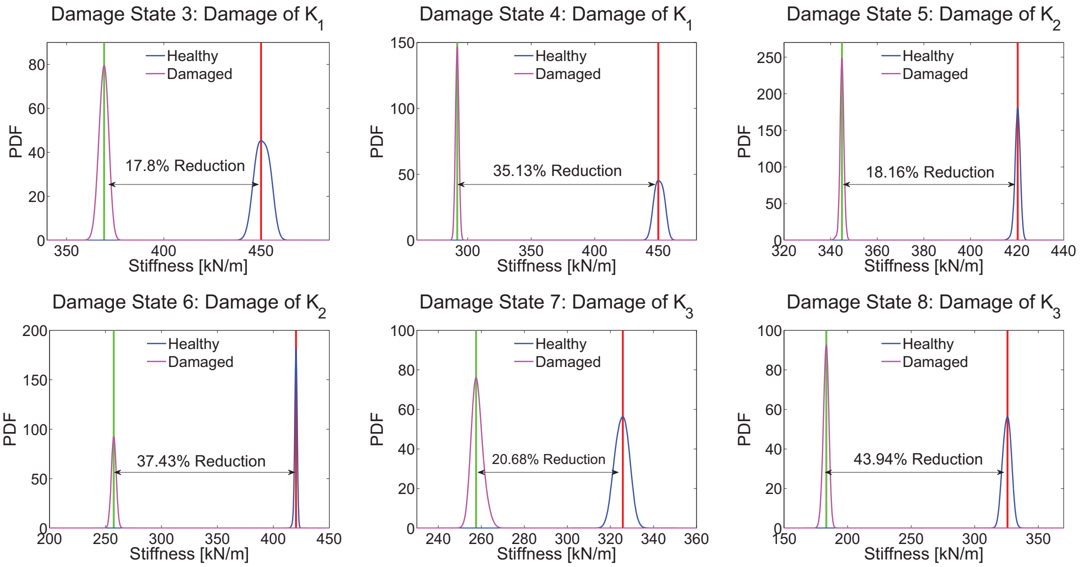


Figure 4.23: The identified posterior PDFs of damaged stiffnesses corresponding to six damage states of the LANL three-storey structure.

4.6 Conclusions

In this chapter, a hybrid optimization methodology for the probabilistic finite element model updating of structural systems is presented. The model updating process is first formulated as an inverse problem analyzed by Bayesian inference and is solved using a hybrid optimization algorithm. The proposed hybrid approach consists of a global search operator (the modified artificial bee colony (MABC) algorithm) and a local search operator (the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method). Parametric studies and numerical optimization of mathematical functions show that MABC-BFGS has faster convergence and much better solution search ability compared with the standard ABC algorithm.

The proposed MABC-BFGS approach is then applied to FEM updating through minimizing the regularized objective function derived from Bayesian inference. System parameters as well as the prediction error covariance are updated iteratively in the optimization process. Posterior PDFs of system parameters can be determined using a weighted sum of Gaussian distributions. Both numerical and experimental examples, e.g., the 3D Phase I IASC-ASCE benchmark structure and an LANL three-storey frame structure, are finally illustrated to verify the validity of the proposed methodology. Simulation results successfully show the effectiveness, robustness and efficiency of the proposed method in FEM updating and damage detection. It can accurately find the correct parameter values with small uncertainties even in the presence of measurement noise and modeling errors. Furthermore, even if the a priori knowledge of the parameter distributions is not well known, the posterior system parameters can still be updated quite well. In general, the proposed algorithm is successfully applied to probabilistic structural FEM updating as well as damage detection even with incomplete noisy input-output measurements.

Part II

Meso-scale (fine-scale) analysis

Chapter 5

Detection of Multiple Flaws Using Single-Scale Optimization

In this chapter, a novel adaptive approach based on XFEM and a hybrid ABC algorithm is presented to detect and quantify multiple flaws in critical structural components without a priori knowledge about the number of flaws. Topological variables are introduced into the search space and used to adaptively activate/deactivate flaws. This chapter is reproduced from the paper co-authored with Professors Haim Waisman and Raimondo Betti, which was published in the International Journal for Numerical Methods in Engineering [96].

5.1 Introduction

Nondestructive identification of flaws and defects (e.g., cracks and voids) in critical structural components, formulated as an inverse problem or a problem of system identification, is an important subfield of SHM, that has drawn significant attention during the past few decades [40, 97–100]. Clearly, flaw detection is extremely important for assessment of the reliability and durability of structures, determination of the appropriate maintenance procedures, and prediction of its remaining service life. Nondestructive Evaluation (NDE) of damage is an attractive methodology to assess flaws without disassembling or damaging the structure in the process.

Nondestructive testing methods such as ultrosanic, radiographic and electrical impedance tomography have been widely used for structural damage detection [101–103]. While these methods are attractive under some conditions, they are limited to laboratory settings and specific flaw conditions. In addition, traditional nondestructive evaluation approaches such as the time-of-flight models and image-processing tools [104] are also limited in that they sometimes unable to provide accurate information on the location, size and shape of damage.

A robust computational model that can identify various flaw types under general settings (not necessarily in well controlled laboratory environments and perhaps even under real operational time as those occur) are necessary. Such models could provide accurate prediction of damage and prevent catastrophic structural failures. Such computational model consists of the processes of solving both the forward and the inverse problems. The inverse problem is usually solved by an iterative optimization technique through minimization of measured and predicted system response error. In this iterative process, the unknown physical properties (e.g., the flaw boundaries) of the system are characterized by a set of parameters which are updated by the optimization method until the simulated system response best matches the measurements. Flaw parameters vary 'on the fly' and converge to the 'best fit' values. In every iteration, the predicted system response is approximated by solving a forward problem, where flaw characteristics have been determined by the optimizer.

Numerical methods such as the standard finite element method (FEM) can be employed to solve the forward problem. However, these require remeshing of the domain in order to adjust the mesh to the varying flaw characteristics, where flaw boundaries have to conform to the mesh. While fast remeshing algorithms become increasingly available [105], it is still not a trivial task and require significant computational effort, in particular when the number of forward solves is large. The number of iterations in such inverse problems often depends on flaw and mesh sizes, number of flaws and sensors, and proximity of flaws to the sensors.

One way to avoid costly remeshing of the forward problem, is via the boundary element method (BEM), which has been popular in flaw identification problems [40, 97, 106–110]. BEM transforms the governing equations of elasticity into a boundary integral equation and solves the problem using Green's function. Nonetheless, if the flaw boundaries are arbitrarily shaped, calculation of the Green's function within a heterogeneous solid becomes quite difficult. Therefore, applying BEM to these problems is still challenging.

Only recently, a new numerical approach called the eXtended Finite Element Method (XFEM) was proposed in application to flaw detection problems [1–4, 111, 112]. XFEM provides an attractive alternative to standard finite elements in that they do not require fine spatial resolution in the vicinity of discontinuities nor do they require repeated re-meshing to properly address propagation of cracks or detection of flaws in such inverse problems. The XFEM is a Partition of Unity (PU) based technique that locally enriches the approximation space with some physical knowledge of the problem at hand [113–115]. In many applications, the XFEM is also combined with Level set methods that provides a convenient framework for identifying enriched nodes in the mesh [114, 115].

In the context of flaw detection, Rabinovich *et al.* [1, 2] were the first to present a computational scheme based on XFEM and a genetic algorithm (GA) to detect cracks in flat membranes subjected to either static or dynamic excitations. In a later work by Waisman *et al.* [3], the XFEM-GA technique was extended to the detection of various types of flaws such as straight cracks, circular holes and irregular shaped holes in elastostatics. Several GAs, e.g. classic GA, micro-GA, and Sawtooth-GA, were employed. The scheme was further improved by Chatzi *et al.* [4], proposing a generic XFEM formulation of an elliptical hole which is utilized to detect any type of flaw (cracks or holes) of any shape. In addition, their work included experimental validation studies. More recently, Jung *et al.* [111] proposed an identification scheme based on a gradient-type optimization method and dynamic XFEM for identifying scatters (e.g., cracks, voids and inclusions) in an elastic heterogeneous media. Yan *et al.* [112] proposed a guided Bayesian inference approach for multiple-flaw parameter detection and uncertainty quantification. Employing an optimization scheme within an XFEM framework to identify model parameters, such as the order of various singularities or thickness of boundary layers, has also been studied by Waisman and Belytschko [116].

Optimization algorithms that find the global minimum such as GAs are more suited for these inverse problems as there is no straightforward dependency of the objective function on the design variables. Though gradient-based optimization methods have faster convergence, they possess the issue of initial guess. Though the algorithms in literature have proven to be successful methods for detection of flaws, they focused on single flaw detection problems. They assumed that only one flaw exists and hence the number of specified parameters in the search space, used for identification, is predefined a priori.

While detection of a single flaw is important, multiple flaws may exist in many practical cases. Since we usually have no a priori knowledge on the number of flaws to be identified, detection of multiple flaws is a difficult task as one cannot directly determine the number of flaw parameters in the optimization process. Hence, an algorithm that can handle multiple flaws without a priori knowledge on the number of flaws, must be developed. To this end, we propose an elegant identification approach for multiple flaws, in which the number of flaws (which is an unknown beforehand), becomes part of the optimization process. The idea is to introduce topological variables (that take the binary values of 0 or 1), as additional parameters in the search space, to automatically switch on and off flaws as the iteration continues, until convergence is reached. Details on such an approach can be found in Section 5.2 of this chapter.

To summarize, in this chapter, we propose a new algorithm based on the XFEM and the HABC algorithm (proposed in Chapter 3) to detect and quantify multiple flaws in structures. The algorithm offers several novel concepts, namely: (1) the use of topological variables in the optimization process to activate/deactivate flaws during the analysis (hence the number of flaws need not be known a priori), and (2) the HABC algorithm which provides a robust, efficient, and fast way to solve these non-unique inverse problems.

The chapter is organized as follows. Section 5.2 describes the formulations of the forward and the inverse problems. Section 5.3 presents the XFEM formulation for multiple circular void flaws modeling. In section 5.4, the XFEM-HABC flaw detection scheme is presented. Section 5.5 presents several numerical examples with increasing complexities, and finally, discussions and conclusions are given in section 5.6.

5.2 Formulation of the identification problem

Detection and quantification of flaws in structures, e.g., voids, cracks, damage regions, can be viewed as an inverse engineering problem in that it involves the solution of both the forward (direct) and the inverse (indirect) problems. In general, when one solves a forward problem, the problem domain, material properties, governing equations, boundary

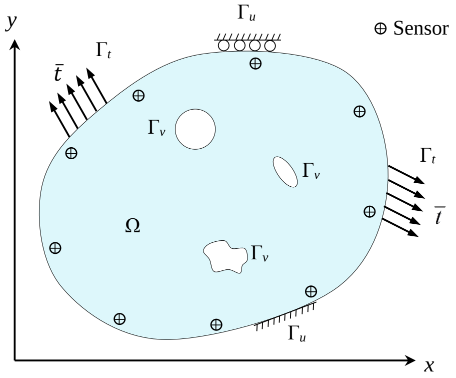


Figure 5.1: Generic solid with traction-free flaws and sensors used to measure its response.

conditions, etc., are given and the response of the system is obtained. However, when some of the aforementioned information is incomplete, as in most real applications, e.g., only measurements are available at some given points while the system characteristics are unknown, the problem becomes an inverse problem.

Inverse problems can be solved by first guessing the missing information, solving the forward problem and then repeatedly updating the guess in the forward model until the simulated system response best fits the measurements recorded by sensors. In this section, the formulation of the forward and of the inverse problems are discussed.

5.2.1 The forward problem

Consider the forward problem of an elastic solid in domain $\Omega \in \mathbb{R}^2$ with the boundaries Γ_u, Γ_t and Γ_v as shown in Figure 5.1, where Γ_u is the outer displacement boundary, Γ_t the traction boundary and Γ_v the inner traction-free void boundary. Note that $\Gamma = \Gamma_u \cup \Gamma_t \cup \Gamma_v$ and $\emptyset = \Gamma_v \cap \Gamma_u \cap \Gamma_t$. The strong form of this forward problem can be written as

$$\begin{aligned}
\nabla \cdot \sigma + \mathbf{b} &= \mathbf{0} && \text{in } \Omega \\
\sigma &= \mathbf{D} : \epsilon && \text{in } \Omega \\
\epsilon &= \nabla^s \mathbf{u} && \text{in } \Omega \\
\sigma \cdot \mathbf{n} &= \bar{\mathbf{t}} && \text{on } \Gamma_t \\
\sigma \cdot \mathbf{n} &= \mathbf{0} && \text{on } \Gamma_v \\
\mathbf{u} &= \bar{\mathbf{u}} && \text{on } \Gamma_u
\end{aligned} \tag{5.1}$$

where "$\cdot$" and ":" are tensor products; σ denotes the Cauchy stress tensor; $\mathbf{b}$ is the body force vector per unit volume; $\mathbf{D}$ is the elasticity constitutive tensor; ϵ is the strain tensor; $\mathbf{u}$ is the displacement vector; $\mathbf{n}$ denotes the outward unit normal vector; $\bar{\mathbf{t}}$ and $\bar{\mathbf{u}}$ represent the prescribed tractions and displacements on Γ_t and Γ_u, respectively. ∇ represents the Nabla operator while $(\nabla\cdot)$ denotes the divergence operator; ∇^s denotes the symmetric part of the Nabla operator, defined as $\nabla^s = \frac{1}{2}\left(\nabla^T + \nabla\right)$.

The weak form of Equation (5.1) can be written as:

Find $\mathbf{u} \in \mathscr{U}$ in domain Ω such that

$$\int_{\Omega} \nabla^s \mathbf{w} : \mathbf{D} : \nabla^s \mathbf{u} \; \mathrm{d}\Omega = \int_{\Omega} \mathbf{w} \cdot \mathbf{b} \; \mathrm{d}\Omega + \int_{\Gamma_t} \mathbf{w} \cdot \bar{\mathbf{t}} \; \mathrm{d}\Gamma_t \qquad \forall \mathbf{w} \in \mathscr{W} \tag{5.2}$$

with the trial and weight function spaces ($\mathscr{U}$ and $\mathscr{W}$) expressed as

$$\begin{aligned} \mathscr{U} &= \{\mathbf{u} | \mathbf{u} \in \mathscr{H}^1, \mathbf{u} = \bar{\mathbf{u}} \text{ on } \Gamma_u\} \\ \mathscr{W} &= \{\mathbf{w} | \mathbf{w} \in \mathscr{H}^1, \mathbf{w} = \mathbf{0} \text{ on } \Gamma_u\} \end{aligned} \tag{5.3}$$

where $\mathbf{u}$ is the trial function; $\mathbf{w}$ is the weight function; and $\mathscr{H}^1$ is the *Sobolev* space of continuous functions with square integrable derivatives.

After discretization of the displacement trial function $\mathbf{u}^h$, the weight function $\mathbf{w}^h$ and their spaces $\mathscr{U}^h$ and $\mathscr{W}^h$ by finite elements, we obtain the discrete weak form

$$\int_{\Omega^h} \nabla^s \mathbf{w}^h : \mathbf{D} : \nabla^s \mathbf{u}^h \; \mathrm{d}\Omega^h = \int_{\Omega^h} \mathbf{w}^h \cdot \mathbf{b} \; \mathrm{d}\Omega^h + \int_{\Gamma_t^h} \mathbf{w}^h \cdot \bar{\mathbf{t}} \; \mathrm{d}\Gamma_t^h \qquad \forall \mathbf{w}^h \in \mathscr{W}^h \tag{5.4}$$

where Ω^h and Γ_t^h are the discretized domain and traction boundaries, respectively, with $\Omega^h \subset \Omega$ and $\Gamma_t^h \subset \Gamma_t$. It is also noteworthy that $\mathbf{u}^h \in \mathscr{U}^h$ and $\mathbf{w}^h \in \mathscr{W}^h$ with $\mathscr{U}^h \subset \mathscr{U}$ and $\mathscr{W}^h \subset \mathscr{W}$. Here, we employ XFEM to obtain and solve the governing set of discretized equations (see Section 5.3).

5.2.2 The inverse problem

Flaw detection problems can be formulated as an inverse problem in which one's objective is to find the set of parameters describing the physical flaw/flaws, given a number of measured data of the system response (e.g., displacements, accelerations, strains, etc.). The measurements are, in general, recorded by a set of sensors placed on the structure. Typically, such problems are solved using an iterative process by minimizing the difference between measured data and predicted data obtained from each forward model. The iteration proceeds by updating the forward model parameters until the optimization converges to the flaws or the damage regions that minimize the global error set by the objective function.

The inverse problem can be summarized as follows: Given Ω, Γ_t, Γ_u, $\bar{\mathbf{u}}$, $\bar{\mathbf{t}}$, $\mathbf{D}$, and some specific measured response at some given locations (the strain field $\boldsymbol{\varepsilon}$ in this work), ones objective is to find the inner flaw boundary Γ_v.

In the current work, we assume that each flaw is approximated by a circular shape void, hence in two dimensions each flaw is represented by its center coordinates and radius, (x, y, r), respectively. The problem of identifying the inner void boundary becomes then to identify these parameters for each flaw. In general the solution to this problem is non-unique, unless sufficient number of sensors are employed and enough measurements are employed. Mathematically, this inverse problem is casted as an optimization problem whose objective function is defined as the error between the actual measured structural response and the estimated response of a parametric model. The parametric model is described by a set of unknown variables in a vector form:

$$\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_n\} \in \mathbb{R}^n \tag{5.5}$$

where n represents the number of parameters. These parameters are identified once the optimization process has converged. Herein, convergence is determined when the objective function is minimized, that is

$$g(\boldsymbol{\theta}) = \sum_{\alpha=1,\beta=1}^{\mathcal{D}(\alpha\leq\beta)} \frac{\|\boldsymbol{\varepsilon}^s_{\alpha\beta}(\boldsymbol{\theta}) - \boldsymbol{\varepsilon}^m_{\alpha\beta}\|}{\|\boldsymbol{\varepsilon}^m_{\alpha\beta}\|} \tag{5.6}$$

where $\boldsymbol{\varepsilon}^s$ is the computed strain vector; $\boldsymbol{\varepsilon}^m$ is the measured strain vector at several specific points along the surface of the structure where sensors (strain gauges) are located; subscripts α and β denote the direction of strain components; $\mathcal{D}$ is the dimension of the problem (i.e. 1D, 2D, 3D); $\|\cdot\|$ is the Euclidean norm of a vector. $\boldsymbol{\varepsilon}^s$ is a function of $\boldsymbol{\theta}$ which is obtained through the forward analysis. In general, the identification problem can be summarized as:

$$\text{Find } \boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_n\} \in \mathscr{S} \text{ such that } g(\boldsymbol{\theta}) \rightarrow \min \tag{5.7}$$

where $\mathscr{S}$ is the feasible n-dimensional parameter search space which can be generally written as:

$$\mathscr{S} = \{\boldsymbol{\theta} \in \mathbb{R}^n | \theta_j^{\min} \leq \theta_j \leq \theta_j^{\max},\ \forall j = 1, 2, \ldots, n\} \tag{5.8}$$

where $\theta_j^{\min}$ is the lower bound of the j-th parameter and $\theta_j^{\max}$ is the corresponding upper bound. More specifically, $\mathscr{S}$ is defined by providing an initial guess on the number of flaws. For simplicity, let us assume that n_v flaws are present in the structure. Thus the number of optimization variables is known and, $\mathscr{S}$ can be written as:

$$\mathscr{S} = \mathscr{S}_1 \cup \cdots \cup \mathscr{S}_k \cup \cdots \cup \mathscr{S}_{n_v} \tag{5.9}$$

with

$$\mathscr{S}_k = \{x_k, y_k, r_k \in \mathbb{R}^{n_v} | x_k^{\min} \leq x_k \leq x_k^{\max}, y_k^{\min} \leq y_k \leq y_k^{\max}, r_k^{\min} \leq r_k \leq r_k^{\max}\} \tag{5.10}$$

and

$$\boldsymbol{\theta} = \{x_1, y_1, r_1, \ldots, x_k, y_k, r_k, \ldots, x_{n_v}, y_{n_v}, r_{n_v}\} \tag{5.11}$$

where $k = 1, 2, \ldots, n_v$; n_v is the number of flaws (voids); (x_k, y_k) are k-th circle center coordinates; r_k is the k-th radius; $[\cdot]_k^{\min}$ and $[\cdot]_k^{\max}$ are the lower and upper bounds of the k-th flaw for x, y and r, respectively.

In practice, the number of flaws to be detected and quantified is often unknown. When considering multiple flaws detection using an optimization technique based approach, an important question arises: how to determine the number of parameters in the optimization scheme to be identified?

One possible brute force solution is the sequential introduction of flaws into the optimization process, i.e. increasing the number of unknown parameters to be identified as the iterative scheme continues, as illustrated in Figure 5.2(a). For example, it is shown that if one starts with a single flaw and adaptively adds more flaws to the search space, assuming that some critical number of iterations or some critical predefined objective function value are reached, then the overall convergence behavior to the global objective function is erratic and not smooth, and in some cases may not converge at all. Hence such scheme may require tremendous computational effort to obtain global convergence as it is not clear how to efficiently set the criterion for the adaptive incrimination of flaws.

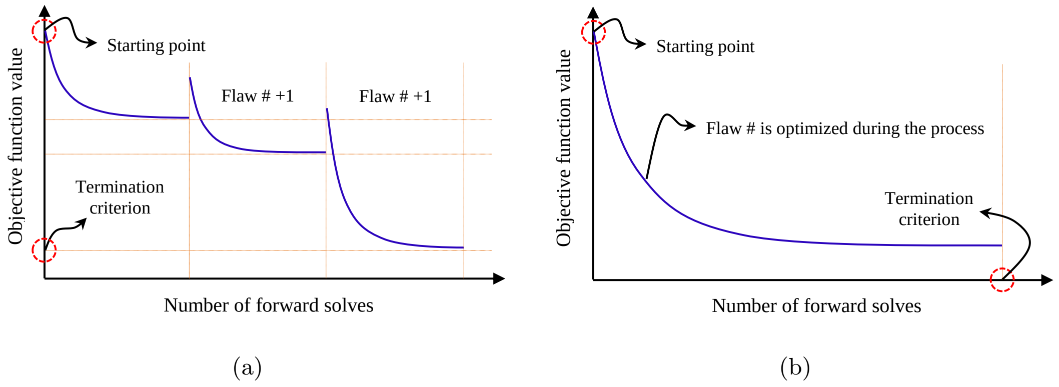


Figure 5.2: Convergence to the global minimum: (a) brute force sequential method: the number of flaws increases adaptively until the termination criterion is reached (global convergence is non-smooth) and (b) a topological variable approach proposed in this chapter: the number of flaws is optimized during the process (global convergence is smooth).

To this end, we propose a more elegant approach to detect multiple flaws, where the unknown number of flaws becomes part of the optimization. The idea is to introduce topological variables (that take the binary values of 0 or 1) to determine the number of flaws during the optimization process. These topological variables are added as additional parameters to the search space $\boldsymbol{\theta}$, and automatically switch on and off flaws as the iteration continues. Figure 5.2(b) illustrates the global convergence of such method. It is shown to have smooth convergence to the target compared to the brute force sequential optimization method. Moreover, the user only needs to specify the final convergence criteria and the number of flaws comes out naturally from the optimization scheme. Clearly, this approach leads to convergence with a significantly smaller effort than the sequential approach as shown in the results Section 5.5.2.

Let us assume that the number of these topological variables is n_t. Then $\mathscr{S}$ can be expressed as:

$$\mathscr{S} = \mathscr{S}_1 \cup \cdots \cup \mathscr{S}_k \cup \cdots \cup \mathscr{S}_{n_t} \tag{5.12}$$

with

$$\mathscr{S}_k = \{x_k, y_k, r_k, \tau_k \in \mathbb{R}^{n_t} | x_k^{\min} \leq x_k \leq x_k^{\max}, y_k^{\min} \leq y_k \leq y_k^{\max}, r_k^{\min} \leq r_k \leq r_k^{\max}, \tau_k = 0 \text{ or } 1\} \tag{5.13}$$

and

$$\boldsymbol{\theta} = \{\hat{x}_1, \hat{y}_1, \hat{r}_1, \ldots, \hat{x}_k, \hat{y}_k, \hat{r}_k, \ldots, \hat{x}_{n_t}, \hat{y}_{n_t}, \hat{r}_{n_t}\} \tag{5.14}$$

where $\hat{x}_k = \tau_k x_k$, $\hat{y}_k = \tau_k y_k$ and $\hat{r}_k = \tau_k r_k$; $k = 1, 2, \ldots, n_t$; τ_k is the k-th topological variable. The topological variable is essentially a switch that controls the existence of flaws. For instance, if $\tau_k = 0$, then $x_k \tau_k = y_k \tau_k = r_k \tau_k = 0$ which means the k-th flaw size is zero. In other words, such flaw does not exist and its corresponding parameters are canceled during the optimization process. On the other hand, if $\tau_k = 1$, then $x_k \tau_k = x_k$, $y_k \tau_k = y_k$, and $r_k \tau_k = r_k$, the k-th flaw is still active and the topological variable have no impact. Typically, we set $n_t \geq n_v$ so that the topological variables will adaptively turn on and off the number of flaws in cases. Finally, the total number of flaws can be obtained by summation of τ_k, i.e.

$$n_v = \sum_{k}^{n_t} \tau_k \tag{5.15}$$

For instance, consider an example where five topological variables are used for a trial identification process, i.e. $n_t = 5$. Say, if the identified $\boldsymbol{\tau}$ is $\{0, 1, 0, 1, 0\}$, then $n_v = 1 + 1 = 2$ and the identified number of flaws will be two. In addition, the relevant information of flaw locations and sizes will be obtained from the nonzero sets (x_k, y_k, r_k) in $\boldsymbol{\theta}$ (see Equation (5.14)).

Figure 5.2(b) illustrates the global convergence of such method. It is shown to have a smooth convergence to the target compared to a brute force sequential optimization method, shown in Figure 5.2(a). Moreover, the user only needs to specify the final convergence criteria and the number of flaws is determined naturally from the optimization scheme.

The computational domain may have complex geometries in which case one would need to ensure the convergence of parameters within the problem domain Ω. However, directly restricting the parameters to the feasible domain might be difficult because the constraints are not simple, i.e. the shape of the structure Ω may be non-regular as illustrated in Figure 5.3. Hence, a penalty strategy is implemented here to renew the objective function in order to handle the complex constraints. If a candidate parameter set is outside the feasible region, then an exaggerated objective function value is returned. Since this value is much larger than the actual objective function value, its parameter values are easily eliminated

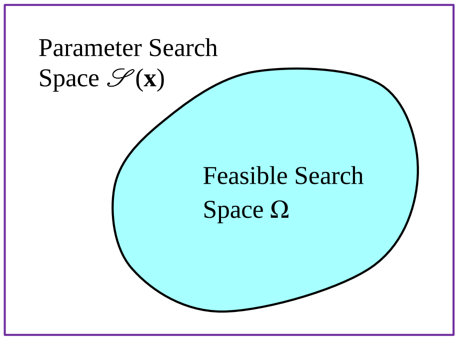


Figure 5.3: Parameter feasible search space.

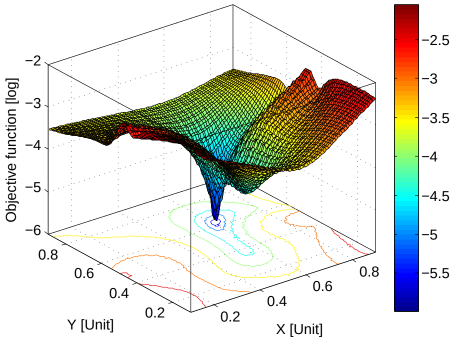


Figure 5.4: An example plot of the objective function for identification of a single circular void with the center at (0.617, 0.627) units and the radius 0.04 units. The axes X and Y represent the problem domain (e.g., the dimension of a plate). Note that the radius is fixed.

and replaced by other optimal values. The penalty objective function can be written as:

$$G(\boldsymbol{\theta}) = \begin{cases} g(\boldsymbol{\theta}), & \text{if} \quad p(x_k, y_k) \in \Omega \\ \kappa, & \text{if} \quad p(x_k, y_k) \notin \Omega \end{cases} \tag{5.16}$$

where κ is a user-defined large number and $p(x_k, y_k)$ denotes the center point of a circle with coordinate (x_k, y_k). Finally, the overall identification problem is summarized as:

$$\text{Find } \boldsymbol{\theta} \in \mathscr{S} \text{ such that } G(\boldsymbol{\theta}) \to \min \tag{5.17}$$

In general, an inverse problem in flaw identification applications is ill-posed and the uniqueness of the solution cannot be guaranteed unless sufficient sensor measurements are utilized or the flaws to be identified are ‘large’ enough compared with the finite element discretization. Moreover, ill-posedness of a problem may also be manifested through sensitivity of parameters to small perturbations in measurements, even in cases that a solution is unique. Nonetheless, such ill-posedness may be circumvented by modifying the type and location of the sensors [1]. The corresponding iterative solution process is usually treated as a nonlinear optimization problem. The objective function of $G(\boldsymbol{\theta})$ is highly non-convex with irregularity of multi-dimensional surface as illustrated in Figure 5.4. Thus, multiple local minima might exist and the optimization algorithm is faced with tremendous difficulties when the algorithm attempts to converge to the global minimum. Therefore, an optimization algorithm with both powerful global and local search abilities should be employed to solve such inverse problems so as to obtain accurate solutions.

5.3 XFEM-based multi-circular-void flaws modeling

Solution of the inverse problem described in Section 5.2 typically requires re-meshing of the domain in each cycle/iteration of the optimization method since the flaw locations and sizes are updated in each forward analysis [3, 117]. While fast meshing algorithms are becoming increasingly available, this task still remains a significant drain on computational resources. An alternative approach to overcome the limitation of standard FEM is the XFEM.

The XFEM is a partition of unity based method, where special functions describing the actual physics of the problem at hand may be incorporated locally into the finite element approximation space to capture features of interest. Aside from a more accurate modeling of the physics, the geometry of internal boundaries such as flaws turns out to be separated from the actual mesh. This feature makes XFEM a powerful tool for modeling the forward problem in these inverse problem setting [113–115].

5.3.1 XFEM formulation for circular voids

In this work we only consider circular voids in the forward problem. Thus the enrichment functions are as follows. Consider a domain $\Omega \in \mathbb{R}^2$, as shown in Figure 5.1, partitioned into finite elements. The displacement field $u^h(\mathbf{x})$ is enriched with a weak-discontinuity function [115]:

$$u^h(\mathbf{x}) = \sum_{I \in \mathbb{N}} \mathbf{N}_I(\mathbf{x})\mathbf{u}_I + \sum_{J \in \mathbb{N}^{\mathrm{en}}} \mathbf{N}_J(\mathbf{x}) V(\mathbf{x}, \boldsymbol{\theta})\mathbf{a}_J \qquad (\mathbf{u}_I, \mathbf{a}_J \in \mathbb{R}^2) \tag{5.18}$$

where $\mathbf{N}$ is the shape function, $\mathbb{N}$ represents a set of nodes in the mesh and $\mathbb{N}^{\mathrm{en}}$ is a subset of $\mathbb{N}$ that contains all the nodes enriched by the function V. The enriched nodes, belonging to elements in which the weak discontinuity passes through, are augmented with additional degrees of freedom $\mathbf{a}_J$, which add more unknowns to the global system of equations. Hence, the discretized weak form for an enriched element is given as:

$$u_e^h(\mathbf{x}) = \sum_{I \in \mathbb{N}_e} \mathbf{N}_I\,(\mathbf{x})(\mathbf{u}_I + V(\mathbf{x}, \boldsymbol{\theta})\mathbf{a}_I) \tag{5.19}$$

where $\mathbb{N}_e$ refers to the number of element nodes. It may be seen from Equation (5.19) that the physical displacement at an enriched node I is provided in terms of both the standard DOF $\mathbf{u}_I$ and the enriched DOF $\mathbf{a}_I$. Following the work of Sukumar *et. al* [115], we employ a shifted version of the enrichment function such that

$$u_e^h(\mathbf{x}) = \sum_{I \in \mathbb{N}_e} \mathbf{N}_I(\mathbf{x}) \left[\mathbf{u}_I + \left(V(\mathbf{x}, \boldsymbol{\theta}) - V(\mathbf{x}_I, \boldsymbol{\theta})\right)\mathbf{a}_I\right] \tag{5.20}$$

5.3.2 Discontinuity modeling for multi-circular voids

The level set function is used to represent weak discontinuities such as voids within the framework of XFEM. The enrichment function used for voids commonly takes the following Heaviside form in Equation (5.21) [115]:

$$V(\mathbf{x}, \boldsymbol{\theta}) = \begin{cases} 1, & \text{if} \quad \phi(\mathbf{x}, \boldsymbol{\theta}) \geq 0 \\ 0, & \text{if} \quad \phi(\mathbf{x}, \boldsymbol{\theta}) < 0 \end{cases} \tag{5.21}$$

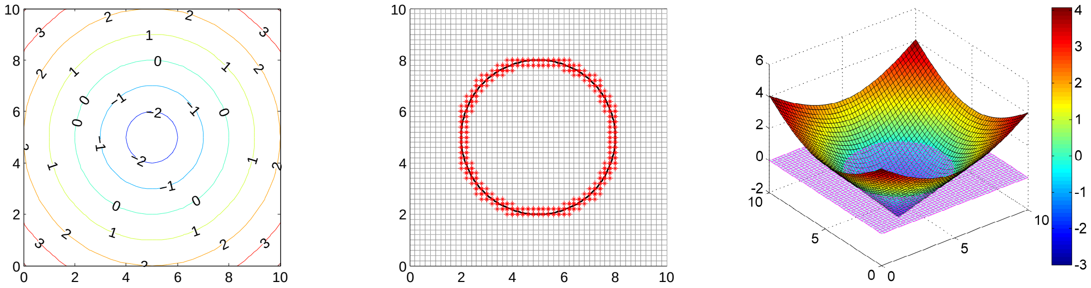
where $\phi(\mathbf{x})$ is the level set function. The nodes that lie inside the void and whose nodal support is not intersected by the void are removed from the calculations. This is done usually by removing the degrees of freedoms associated with those nodes from the system of equations and solving the system only for the remaining degrees of freedoms. For a circular void, the minimum signed distance function is used to construct the level set function:

$$\phi(\mathbf{x}, \boldsymbol{\theta}) = \|\mathbf{x} - \mathbf{x}^c\| - r^c \tag{5.22}$$

where $\mathbf{x}$ is any query point, $\mathbf{x}_c = (x^c, y^c)$ is the center coordinates of the circle, and r^c is the radius of the circle. Note that, when this function is zero, the actual void geometry is obtained. Thus, in this case, the level set function will have a positive value outside the circle, a negative value inside the circle and the points that lie on the circle will be represented as a zero value of the function. If a body contains several circular discontinuities, the level set function for all the discontinuities can be defined as:

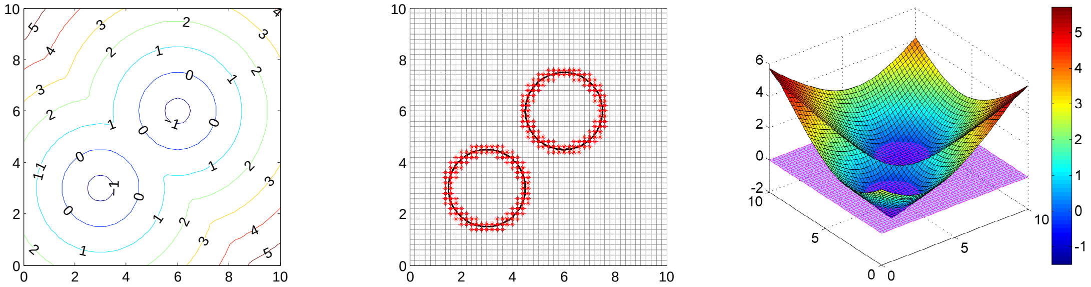
$$\phi(\mathbf{x}, \boldsymbol{\theta}) = \min\left\{\|\mathbf{x} - \mathbf{x}_i^c\| - r_i^c\right\}_{i=1,2,\ldots,n_c} \tag{5.23}$$

where n_c is the number of circles; $\mathbf{x}_i^c = (x_i^c, y_i^c)$ and r_i^c refer to the center coordinates and the radius of the i-th circle. Figures 5.5–5.7 show the level sets and enrichments visualization for the cases of a single void, two separate voids and two overlapping voids, respectively.



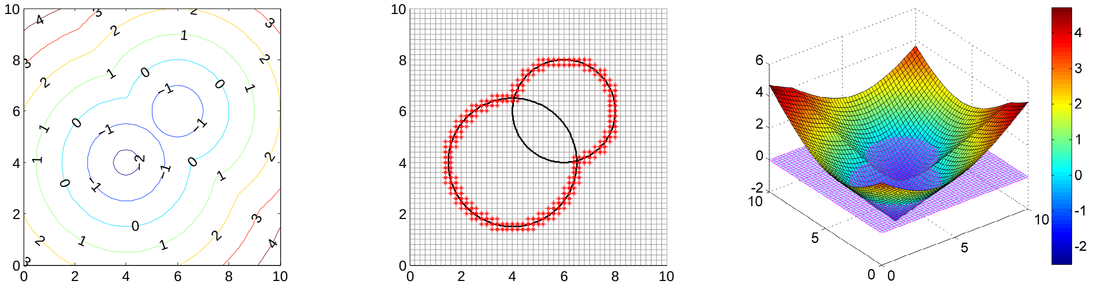
(a) Contour plot (b) Enriched nodes (c) Level set function

Figure 5.5: Level set and enrichment visualization of a single void.



(a) Contour plot (b) Enriched nodes (c) Level set function

Figure 5.6: Level set and enrichment visualization of two separate voids.



(a) Contour plot (b) Enriched nodes (c) Level set function

Figure 5.7: Level set and enrichment visualization of two overlapping voids.

5.4 The XFEM-HABC identification scheme

The inverse problem in this work is solved by the HABC algorithm proposed in Chapter 3 (see Section 3.3.1). Nevertheless, in order to improve the efficiency of the Onlooker Phase search, an alternative solution updating strategy is proposed to take advantage of the global best solution to guide the search of candidate solutions. In other words, the "guided-to-best" solution updating strategy is expressed as:

$$\Theta_{ij}^{up} = \Theta_{ij} + 2\chi_{iter}(\text{rand} - 0.5) \cdot (\theta_j^b - \Theta_{kj}) \tag{5.24}$$

where the superscripts and the subscripts have the same meaning given in Section 2.3.2. The vector $\boldsymbol{\theta}^b$ denotes the current global best solution with the best fitness value in the population. And χ_{iter} is the convergence control parameter proposed in Equation (2.8) (note that $m = 3$ and $\delta = 0.2$ are used to determine χ_{iter} in this chapter).

Note that Equation (5.24) is only used in the Onlooker Phase search while the Employed Phase search still takes the updating strategy in Equation (2.9). Without loss of generality, the combination of Equations (2.9) and (5.24) consist of a global random search and a guided-to-best search. This change is assumed to make the algorithm converge more quickly since the solution can be biased towards the best solution using Equation (5.24).

To summarize, the identification process of XFEM-HABC algorithm is initiated by generation of a set of individuals randomly sampled in the search space and is updated by the operation of bee phases, combined with the local search operator (see NMSM in Section 3.3.1), proceeding towards a global optimum target. As discussed in Section 3.3.3, the proposed algorithm depends on the following user-defined parameter: lower bound vector ($\boldsymbol{\theta}^{\min}$), upper bound vector ($\boldsymbol{\theta}^{\max}$), number of parameters to be identified (n), number of population size (N_{pop}), maximum number of iterations for HABC (N_{max}), number of limit iteration (N_{lim}), starting point for local search (N_{ls}), and maximum number of iterations for NMSM (N_{lsiter}). While large values of N_{pop}, N_{max} and N_{lsiter} will more likely to result in global convergence with a small number of iterations, yet the CPU time will increase dramatically because a significantly large number of forward analysis and fitness evaluations will be required. Thus a smart choice of these parameters (which typically depends on the problem) will determine a good trade-off between number of iterations and CPU time. It is noteworthy that the system parameters ($\boldsymbol{\theta}$) are evaluated by the XFEM solver. In general, the evolutionary process is repeated until the termination criterion is met. The purpose of this procedure is to eventually lead to the best parameter set which can best "fit" the flaws.

Remark. It is noteworthy that the HABC algorithm can directly be applied to optimization problems with continuous parameters/variables according to the nature of the solution updating strategies. However, while the optimization process involves discrete variables such as topological variables in this work, the following equation can be used to revise the updated solution of topological variables:

$$\Theta_{iq}^{\tau} = \begin{cases} 0, & \text{if} \quad \theta_{iq}^{\tau} \leq 0.5 \\ 1, & \text{if} \quad \theta_{iq}^{\tau} > 0.5 \end{cases} \tag{5.25}$$

where $i = 1, 2, \ldots, N_{pop}$ and $q = 1, 2, \ldots, n_t$; the superscript τ denotes the topological variables in $\boldsymbol{\Theta}_i$. Usually, the bounds of the topological variables are set to be $[0, 1]$ so that the cut-off point of 0.5 in can fairly make the probabilities of choosing 0 or 1 identical.

5.5 Numerical experiments

Four numerical examples are presented in this section to investigate the convergence of the proposed XFEM-HABC technique in the detection and quantification of multiple flaws in structures. The examples are structured in an increasing order of difficulty considering different mesh types, flaw shapes, multiple flaws and their proximity to each other and given/unknown a priori knowledge on the number of flaws. The system response in terms of strains is recorded at some given points by strain gauges and is used as the measured data to compute the error norm in Equation (5.6) and minimize the objective function in Equation (5.16). In the process of the XFEM-HABC algorithm for flaws detection, the most time-consuming part is the fitness evaluation (mainly in forward analysis), whereas the time taken by the HABC algorithm itself is negligible. Since the HABC algorithm is a stochastic algorithm, each independent run might bring slightly different solutions. To avoid solution uncertainty and to ensure good results, five independent runs are conducted in the statistical simulations. Finally, the result with a minimum objective function value among the independent runs is selected as the final representative solution for each case. The algorithm convergence has been analyzed in terms of the number of XFEM forward solves (analyses) in the numerical experiments.

The strain measurements used herein as structural response are obtained by solving a reference problem either with an XFEM code or FEM codes. In future work, actual experimental tests will be explored. The XFEM-HABC algorithm was implemented in MATLAB$^{\circledR}$ on a standard Intel (R) Core (TM) 2 Q9550 2.83GHz PC with 2G RAM.

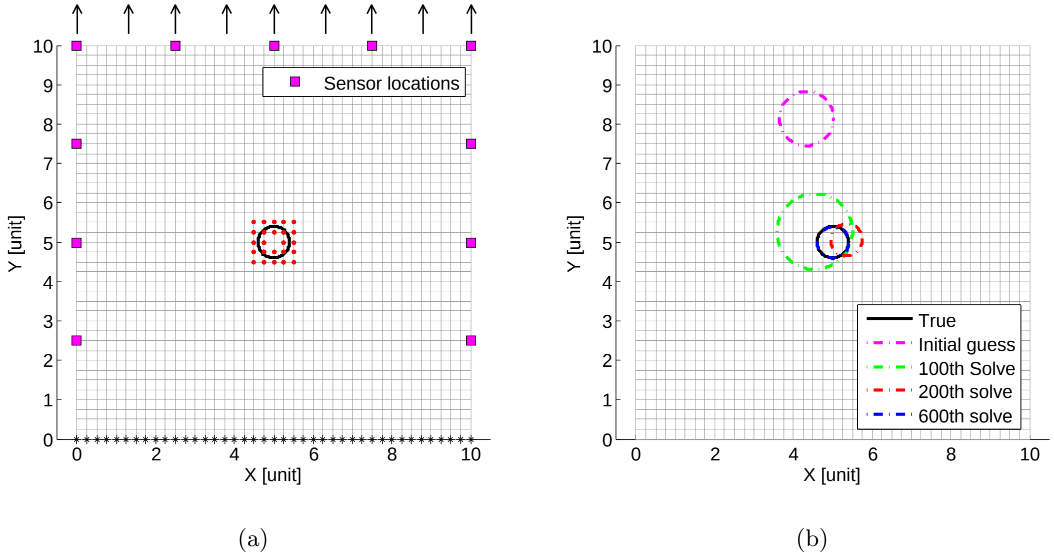


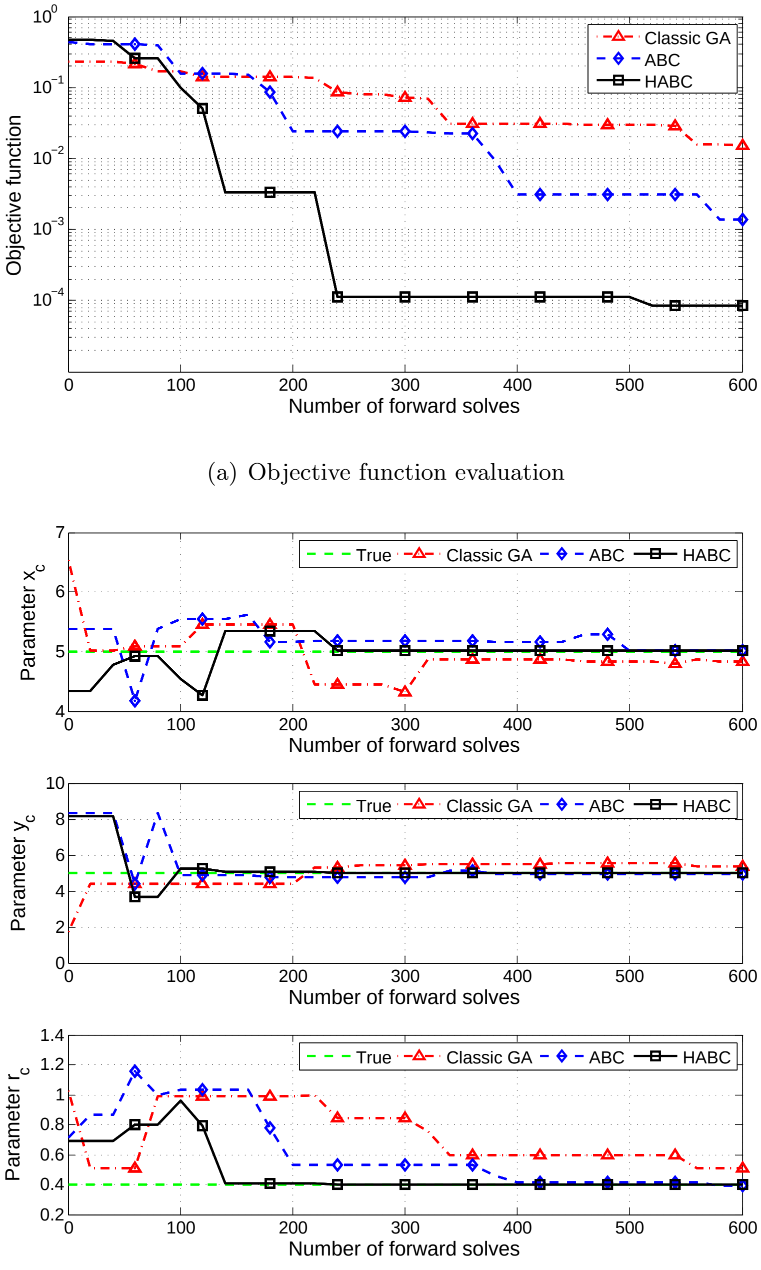
Figure 5.8: Detection of a single flaw: (a) mesh generation, loading condition and sensor placement and (b) snapshots of the XFEM-HABC evolutionary process for the target of a circular void of radius 0.4 (units) located at the center of a rectangular plate.

5.5.1 Detection of a single flaw within a rectangular plate

In this example, we study the performance of XFEM-HABC scheme on the identification of a circular void in a rectangular plate of dimensions 10 by 10 (units) (see Figure 5.8(a)). The purpose of the example is to study convergence behavior of the HABC algorithm (proposed as improvement in this work) as compared with classic GAs and the standard ABC algorithm. For this problem, the number of parameters to be identified is three; namely, the radius r^c of the void and the Cartesian coordinates (x^c, y^c) of its center. Fixed boundary conditions are applied on the bottom edge of the plate. The target circular void dimensions are set equal to $(x^c, y^c) = (5, 5)$, while the radius size is chosen equal to 0.4 units. Eleven strain gauges are assumed to be distributed on the non-fixed boundaries. Reference measurements are obtained by the circular void(s) enrichment using an XFEM code. The overall geometry of the problem as well as the 41 × 41 node structured mesh used for the identification procedure is presented in Figure 5.8(a).

The parameter bounds for the circular void detection problem are defined as $x^c \in [0.5\ 9.5]$, $y^c \in [0.5\ 9.5]$ and $r^c \in [0\ 2]$. The HABC parameters are set as follows: $N_{pop} = 20$, $N_{max} = 30$, $N_{lim} = 30$, $N_{ls} = 12$ and $N_{lsiter} = 15$. For comparison purposes, the standard ABC algorithm was tested, in which the algorithmic parameters are the same as those used in the HABC algorithm. A classic rank-based genetic algorithm (GA) was also implemented in this example by using the MATLAB® GA Toolbox™. In this GA, the population size is chosen to be equal to N_{pop} and the maximum generation number equals to N_{max}. The rate of crossover was selected equal to 0.8 while a Gaussian function with zero mean value was adopted for mutation strategy with a scale factor of 0.5 and shrink factor of 0.75 [43].

Table 5.1 illustrates the identification results using the three different algorithms after 600 forward solves. It can be seen that HABC converges to a much smaller final objective function value of 8.5670×10^{-5} than those obtained by GA and ABC (0.0150 and 0.0014, respectively). Figure 5.9 shows the convergence comparison of the three algorithms for both objective function evaluations and parameter estimation. Clearly, the proposed HABC converges much faster than GA and ABC and shows superior efficiency. However, a standard ABC algorithm can also yield good results that are better than those obtained by GA. Figure 5.8(b) depicts the snapshots of the XFEM-HABC evolutionary process for the target of the



(a) Objective function evaluation

(b) Parameter estimation

Figure 5.9: Comparison of convergence for alternative algorithms (identification of a single circular void).

Table 5.1: Identification results using different algorithms in example 5.5.1.

Flaw parameters	True value	Identified value		
		GA	ABC	HABC
x^c	5.0	4.8304	5.0087	5.0097
y^c	5.0	5.3839	4.9490	4.9891
r^c	0.4	0.5059	0.3891	0.3999
Final objective function value G		0.0150	0.0014	8.5670×10^{-5}

Note that the algorithm is terminated after 600 forward solves.

circular void. In fact, the flaw parameters are able to converge to the true values after only several iterations, which shows the attractiveness of the proposed XFEM-HABC algorithm.

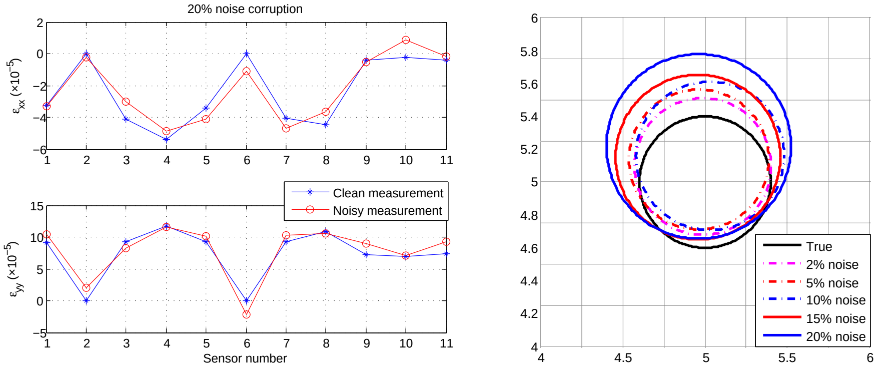
5.5.1.1 Parametric study on measurement noise effect.

To analyze the effect of measurement noise on the identification of the flaw parameters, various noise pollution levels in measurement data have been considered. The measured strains have been polluted with Gaussian zero-mean white noise sequences, whose RMS values are defined as some percentages of the clean signal. Namely, the noisy measured strain vector $\boldsymbol{\varepsilon}^m$ is obtanied by:

$$\boldsymbol{\varepsilon}^m = \boldsymbol{\varepsilon}^c + \mathbf{r} \times \beta_\varepsilon \times q\% \tag{5.26}$$

where $\boldsymbol{\varepsilon}^c$ denotes the correct strain; $\mathbf{r}$ the Gaussian white noise variable; β_ε the RMS value of the correct strain; $q\%$ the noise level (e.g., 2%, 5%, 10%, 15%, and even worse 20%).

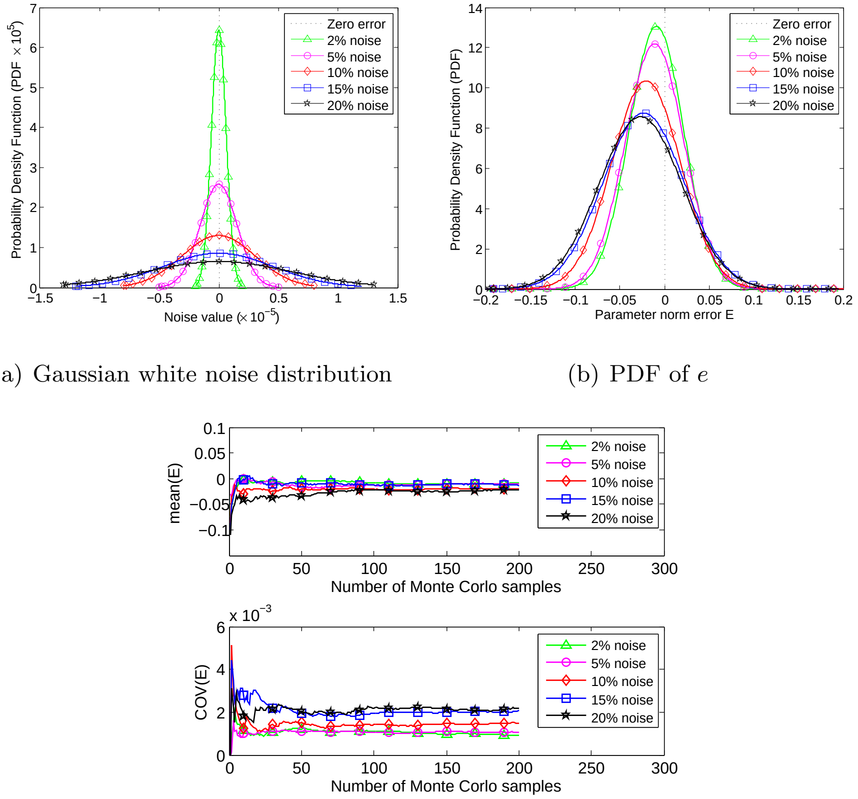
A Monte Carlo simulation, of 200 independent sample runs, has been carried out so as to obtain statistical insights into the measurement noise effect on flaws identification. Each run generates a new Gaussian white noise sequence. For example, Figure 5.10(a) depicts the strain comparison between measured signal with 20% noise corruption and the clean signal. A typical set of identification result is shown in Figure 5.10(b). It can be observed that the proposed XFEM-HABC is able to successfully identify flaw size and location even with high level of noise corruption (i.e., 10%, 15%, 20% RMS noise levels).



(a) Measured strains with 20% RMS noise

(b) Identified flaws

Figure 5.10: Flaw identification with noisy measurements of one Monte-Carlo realization.



(a) Gaussian white noise distribution

(b) PDF of e

(c) Convergence of mean and COV of e

Figure 5.11: Monte Carlo simulation for identification of a single circular void flaw

The norm error is used to evaluate the Monte Carlo simulation results, namely

$$e = 1 - \frac{\|\theta^{id}\|}{\|\theta^{exact}\|} \tag{5.27}$$

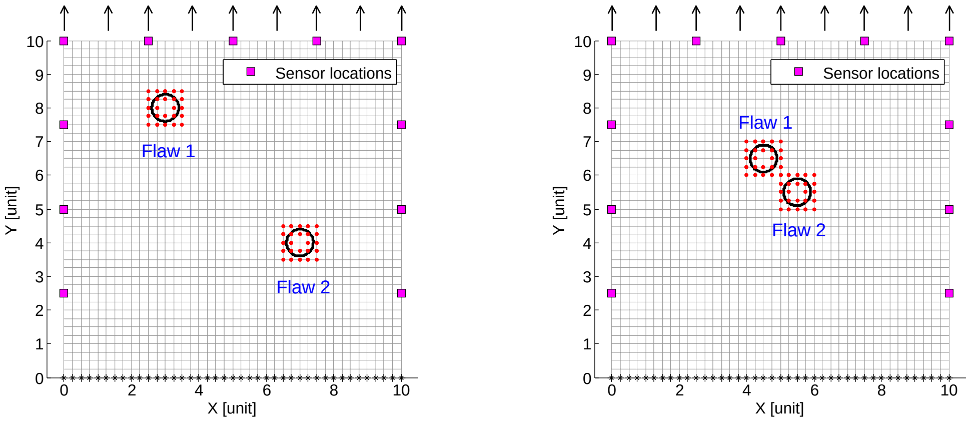
where θ^{id} represents the identified parameter vector and θ^{exact} represents the exact parameter vector. Figure 5.11 summarizes the results of the Monte Carlo simulation. Figure 5.11(a) shows the Gaussian white noise distribution with different RMS levels. Figure 5.11(b) shows the Probability Density Function (PDF) of e. It can be seen that, as the noise level reduces, the mean value of e is closer to zero and the corresponding Gaussian PDF is sharper (e.g., the spread or uncertainty is smaller).

Figure 5.11(c) shows the convergence of the mean and covariance (COV) of E with increasing number of Monte Carlo realizations, where a statistical convergence can be observed. Thus, we conclude that the Gaussian white noise leads to a Gaussian distribution of the identified flaw parameters and the algorithm is robust against high-level noise.

5.5.2 Detection of two flaws within a rectangular plate

The performance of the proposed identification scheme is further studied on the detection of two voids within a rectangular plate. The geometry of the problem is shown in Figure 5.12. The mesh generation, loading condition, sensor placements and essential boundary conditions are the same as those in example 5.5.1. We study the convergence of the proposed algorithm on: 1) two far away flaws and 2) two close flaws. Note that the number of flaws is not known beforehand and topological variables are employed to converge to the true number of flaws. The flaw parameter settings in terms of center coordinates and radii are given in Table 5.2. In all cases, the parameter bounds for the circular voids detection problem are defined as: $x_i^c \in [0.5\ 9.5]$, $y_i^c \in [0.5\ 9.5]$ and $r_i^c \in [0\ 2]$.

In this example, $3n_t$ void parameters along with n_t topological variables need to be identified, where n_t represents the number of topological variables: the radii r_i^c, the Cartesian coordinates (x_i^c, y_i^c) of void centers and the topological variables τ_i, where $i = 1, 2, \ldots, n_t$ and $n_t \geq 2$. Herein, we use $n_t = 4$ for the simulation so that sixteen parameters in total are identified. The HABC parameters used herein are $N_{pop} = 30$, $N_{max} = 100$, $N_{lim} = 40$, $N_{ls} = 15$ and $N_{lsiter} = 15$. Table 5.3 reports the identification results of flaw parameters.



(a) Two far away circular voids

(b) Two close circular voids

Figure 5.12: Mesh generation, loading condition, sensor placement and two circular flaws locations within a rectangular plate.

Table 5.2: Flaw parameter settings in example 5.5.2.

Distance of flaws	Flaw 1 (Units)			Flaw 2 (Units)		
	x_1^c	y_1^c	r_1^c	x_2^c	y_2^c	r_2^c
Case 1: Far away	3.0	8.0	0.4	7.0	4.0	0.4
Case 2: Close	4.5	6.5	0.4	5.5	5.5	0.4

Note that the algorithm is terminated after 3000 forward solves. The performance of the proposed XFEM-HABC algorithm is excellent and yields good estimates of flaw parameters for both far and close flaws scenarios. However, as expected, it can be observed from Table 5.3 and the final objective function value (G) that it is much easier to converge to far flaws. This might be because of the more pronounced non-uniqueness that is more likely to occur in the close flaws scenario (e.g., if the flaws are close enough, they could be approximated by a single flaw instead of two). Also, it can be understood as if there are more flaws than sensor per given area which increases the non-uniqueness of the problem. In general, for both scenarios, the proposed XFEM-HABC converges to the exact number of flaws by switching the topological parameters on and off, and to the true flaw locations and sizes very well. It is also noteworthy that the proposed algorithm is able to converge to a much

Table 5.3: XFEM-HABC identification results in example 6.2

Flaw parameters		Far flaws		Close flaws	
		True value	Identified value	True value	Identified value
Flaw 1	x_1^c	3.0	3.2853	4.5	4.4646
	y_1^c	8.0	7.8413	6.5	6.6908
	r_1^c	0.4	0.3927	0.4	0.4093
Flaw 2	x_2^c	7.0	7.0291	5.5	5.4285
	y_2^c	4.0	3.8890	5.5	5.3844
	r_2^c	0.4	0.3519	0.4	0.4155
Number of flaws ($\Sigma\boldsymbol{\tau}$)		2	2	2	2
Final objective function value G)		8.0269e-4		0.0095	

Note that the algorithm is terminated after 3000 forward solves.

better result with the objective function less than 1×10^{-6} if a larger number of iterations are used (e.g., 60000 forward solves with the CPU time about 41 min not shown here).

Figures 5.13 and 5.14 depict the typical snapshots of the XFEM-HABC evolutionary process for the identification of far flaws and close flaws, respectively. Figures 5.15 and 5.16 show the XFEM-HABC convergence of both objective function evaluations and the number of flaws for detection of both test cases. In these figures, the topological variable approach was also compared with the brute force sequential method, in which flaws are simply added to the search space if no convergence is observed.. It can clearly be seen that the overall objective function convergence behavior of the brute force sequential method is erratic and not smooth (e.g., the jumping-up-phenomena at the 3000th forward analysis), while the topological variable approach has a smooth decreasing convergence. Furthermore, the brute force sequential method requires a larger number of iterations to reach the same level of objective function value compared with the topological variable approach, increasing the computational cost significantly. Since the number of flaws is part of the optimization in the topological variable approach, it keeps changing adaptively until a favorable estimate is reached. After 3000 forward solves, the identified number of flaws by the XFEM-HABC algorithm successfully returns 2, the true value.

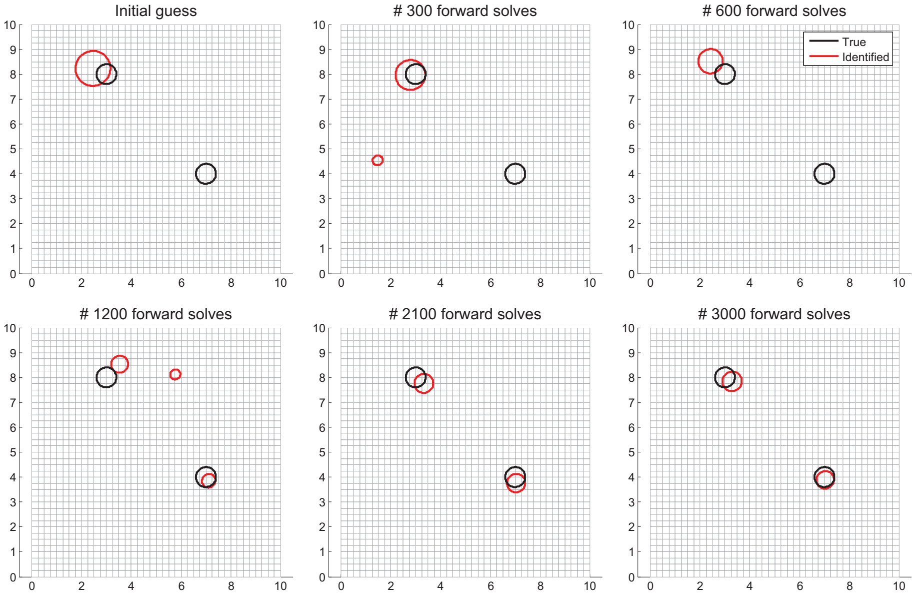


Figure 5.13: Snapshots of the XFEM-HABC evolutionary process for the target of two far away circular voids. Note that the number of identified voids keeps changing dynamically (due to the topological variables) until the algorithm converges to two.

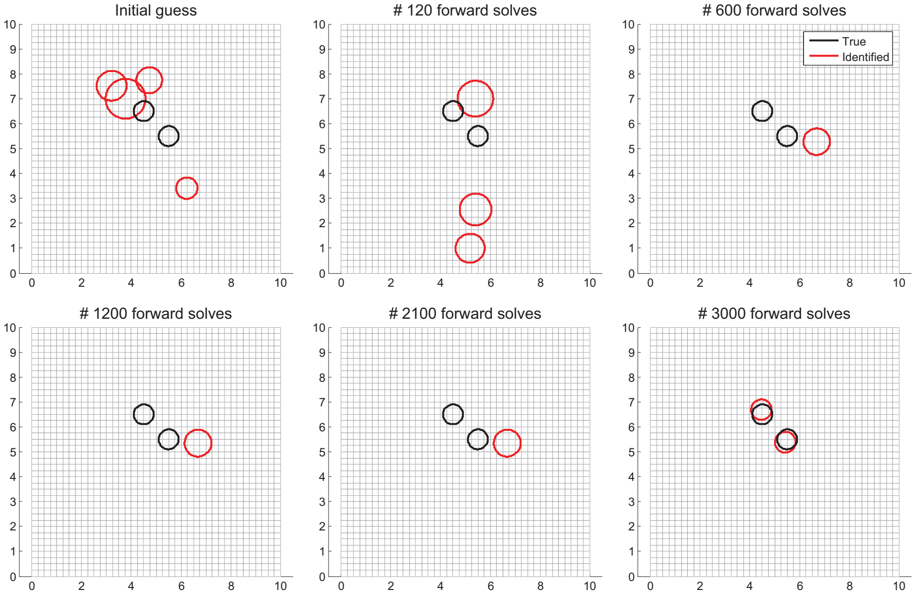
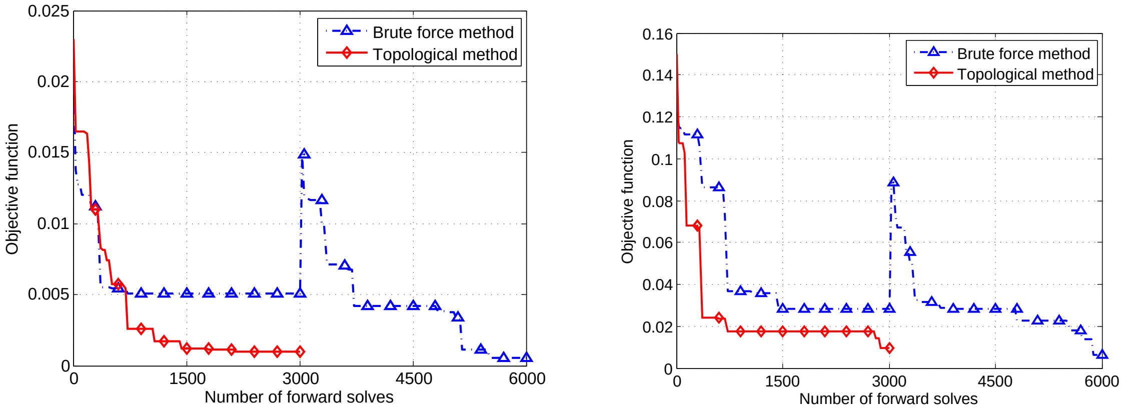
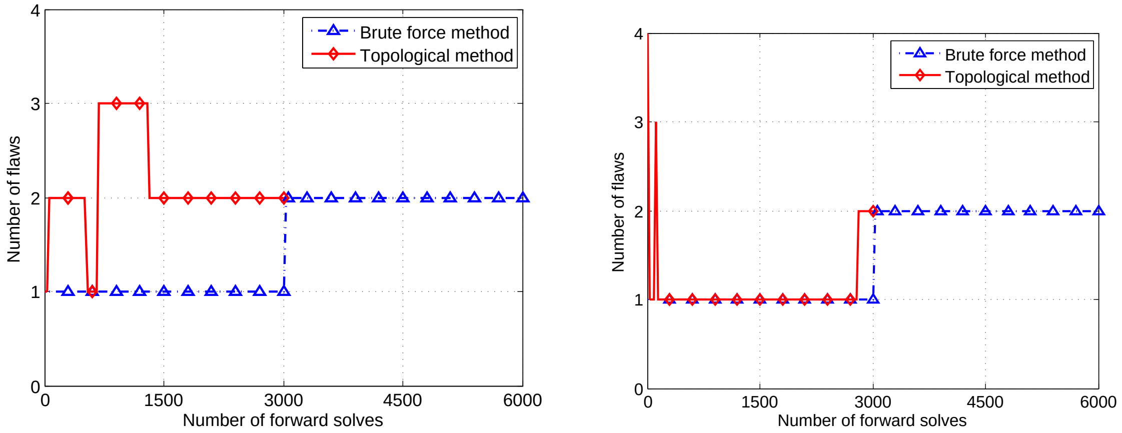


Figure 5.14: Snapshots of the XFEM-HABC evolutionary process for the target of two close circular voids. Note that the number of identified voids keeps changing dynamically (due to the topological variables) until the algorithm converges to two.



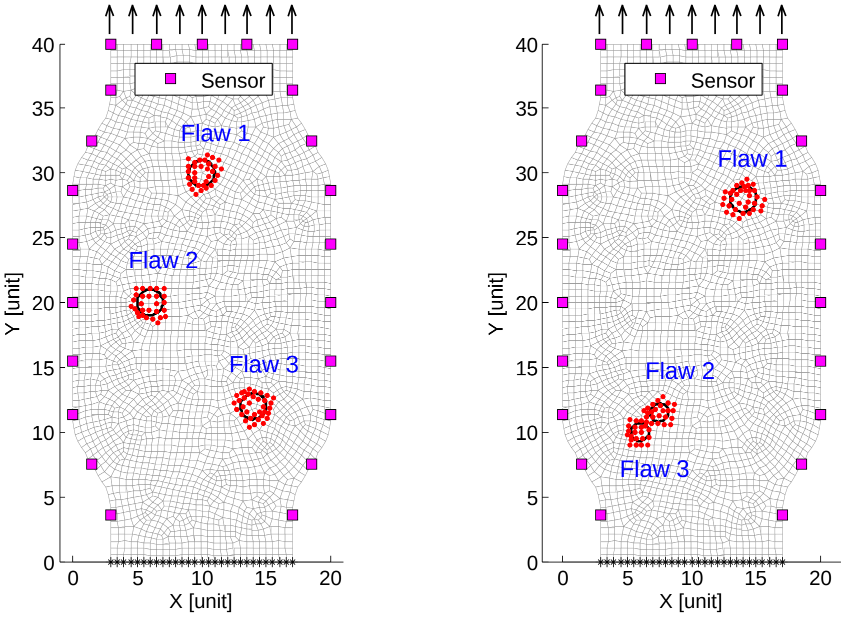
(a) (b)

Figure 5.15: XFEM-HABC convergence of objective function for the topological variables approach versus a sequential brute force approach for the case of (a) two far away circular voids and (b) two close circular voids.



(a) (b)

Figure 5.16: XFEM-HABC convergence of number of flaws for the case of (a) two far away circular voids and (b) two close circular voids.



(a) Case 1

(b) Case 2

Figure 5.17: Mesh generation, loading condition, sensor placement and flaw locations within a rectangular-like plate.

5.5.3 Detection of three flaws within a rectangular-like plate

The next test case studies the performance of the proposed XFEM-HABC algorithm in detection of three flaws within a physical shaped rectangular-like plate with unstructured finite element mesh. Figure 15 depicts the geometry of the problem and the unstructured mesh that consist of 3297 nodes, that is used for the simulation. Fixed boundary conditions are applied on the bottom edge of the 2D plate while a tension traction is applied along the vertical direction on the top edge. Twenty three equally spaced strain gauges are placed on the edges and used for data collection. The reference measurements are simulated by using the circular void enrichments XFEM code. As illustrated in Figure 5.17, two cases regarding the proximity of flaws are considered, Case 1: three circular voids far from each other, and Case 2: three circular voids where two are close and the third is far away . In both cases, the number of flaws is unknown beforehand, thus topological variables are added for the identification.

In this example, we have three flaws for identification whose parameters are given in Table 5.4 for both cases 1 and 2. Since the number of flaws is unknown a priori, four topological variables are used for flaw identification in this case, namely $n_t = 4$. Hence, there are sixteen parameters in total to be identified. The flaw parameter bounds, for each circular void, are defined as: $x_i^c \in [0.5\ 19.5]$, $y_i^c \in [0.5\ 39.5]$ and $r_i^c \in [0\ 2]$, where $i = 1, 2, \ldots, n_t$. The shaped boundaries of the plate are used for objective function penalization checking. This case utilizes the HABC parameters as follows: $N_{pop} = 30$, $N_{max} = 200$, $N_{lim} = 40$, $N_{ls} = 15$ and $N_{lsiter} = 15$ and identification results are summarized in Table 5.4. Note that the algorithm is terminated after 6000 forward solves. Snapshots showing the convergence of the XFEM-HABC evolutionary process are shown in Figures 5.18 and 5.19 for cases 1 and 2, respectively. Figures 5.20 and 5.21 show the convergence of the objective function evaluations and the number of flaws for both cases, respectively. It can be seen that the identification results are excellent. The flaw parameters as well as their quantities are identified accurately. It is interesting to note that two small close target flaws may be approximated by a single candidate void during the optimization process, as illustrated but the snapshots in Figure 5.19. Again, a similar conclusion can be reached as in the previous example that the proposed algorithm is easier to converge to the flaws if they are far from each other. The XFEM-HABC proves to work well on unstructured finite element mesh based flaws detection.

5.5.4 Detection of three irregular-shaped flaws within an arch-like plate

In order to study flaws detection in the presence of artificial noise caused by model error, we study the validity of the proposed algorithm in the case of three non-regular-shaped flaws within an arch-like plate. The geometry of the plate is shown in Figure 5.22. The plate is fixed on the bottom edges. Tension traction is applied on the top edge. Two sets of strain gauges (twenty seven in total) are placed on the free surface boundaries for measurements acquisition. The reference solution is obtained from an FEM model with an unstructured mesh that consists of 5827 nodes. The forward analysis in the identification process is carried out by using the circular voids enrichment XFEM code with the 3429 node structured mesh shown in Figure 5.23. The irregular shaped voids and the use of

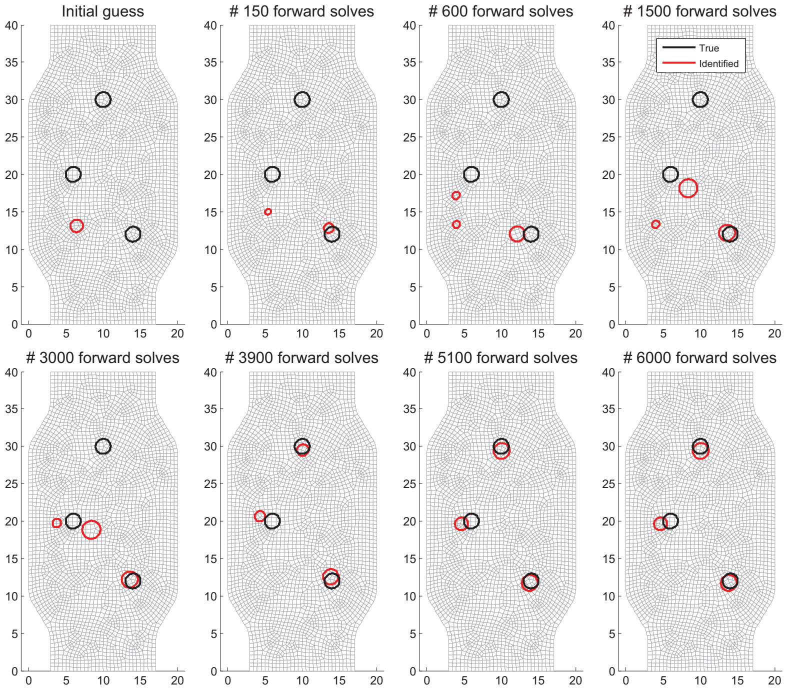


Figure 5.18: Snapshots of the XFEM-HABC evolutionary process in Case 1 (example 5.5.3).

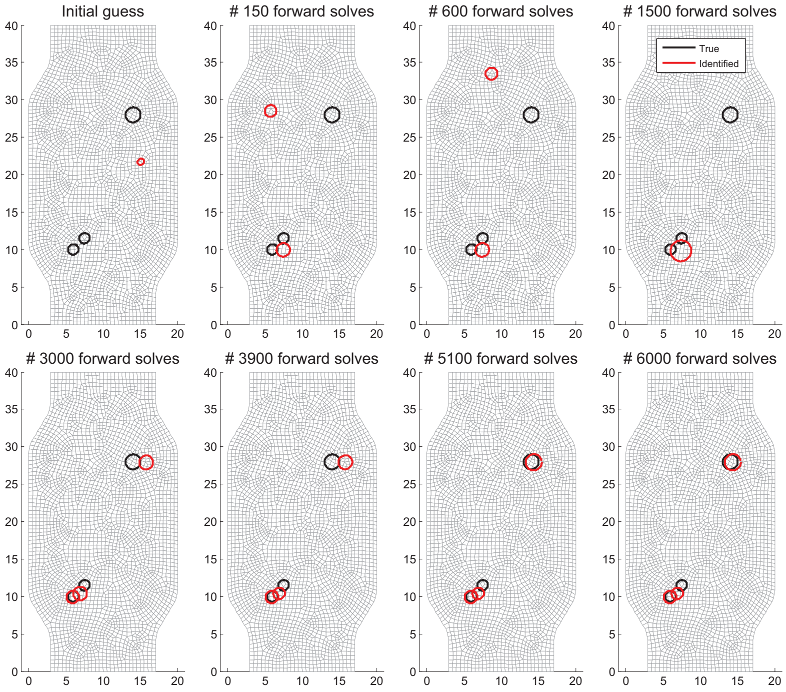
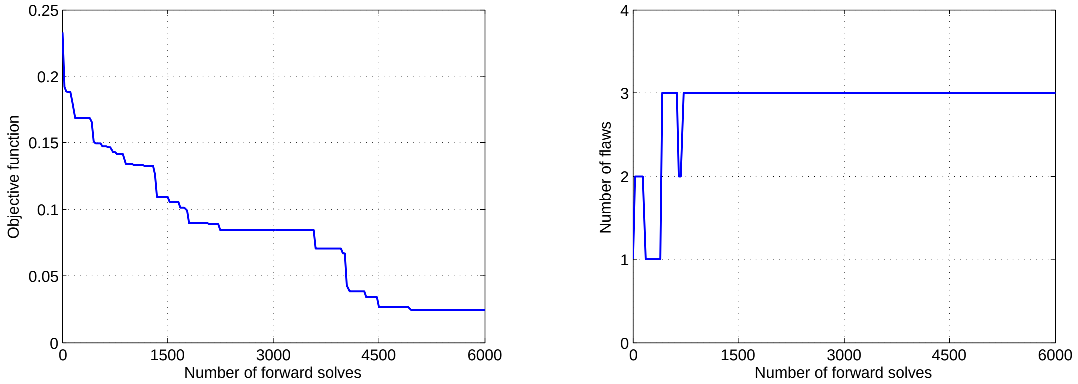


Figure 5.19: Snapshots of the XFEM-HABC evolutionary process in Case 2 (example 5.5.3).

Table 5.4: XFEM-HABC identification results of three circular void flaws in example 5.5.3.

Flaw parameters		Case 1		Case 2	
		True value	Identified value	True value	Identified value
	x_1^c	10	10.0446	14	14.3345
Flaw 1	y_1^c	30	29.3978	28	27.9345
	r_1^c	1	1.0612	1	1.0802
	x_2^c	6	4.6717	7.5	6.9145
Flaw 2	y_2^c	20	19.6535	11.5	10.3912
	r_2^c	1	0.8693	0.7	0.7685
	x_3^c	14	13.7983	6	5.9505
Flaw 2	y_3^c	12	11.7354	10	9.9472
	r_3^c	1	1.0573	0.7	0.8440
Number of flaws ($\Sigma\boldsymbol{\tau}$)		3	3	3	3
Final objective function value G)		0.0248		0.0225	

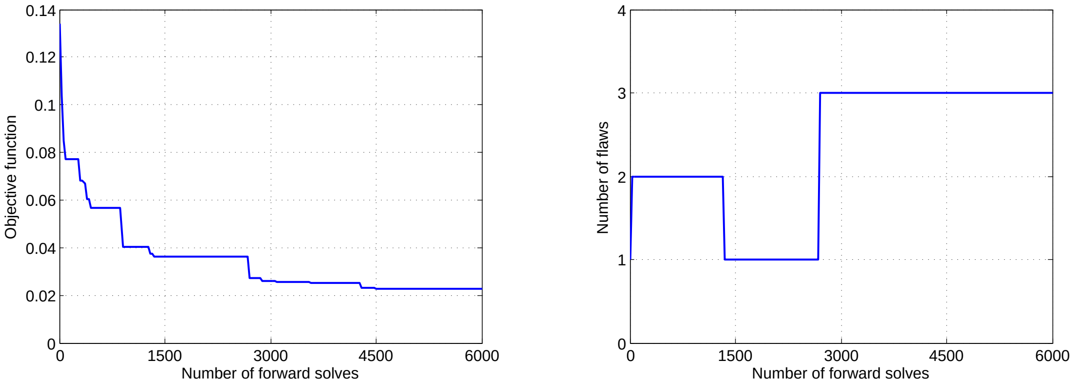
Note that the algorithm is terminated after approximate 6000 forward solves.



(a) Objective function evaluation

(b) Number of flaws

Figure 5.20: Detection of three circular voids in Case 1: XFEM-HABC convergence.



(a) Objective function evaluation

(b) Number of flaws

Figure 5.21: Detection of three circular voids in Case 2: XFEM-HABC convergence.

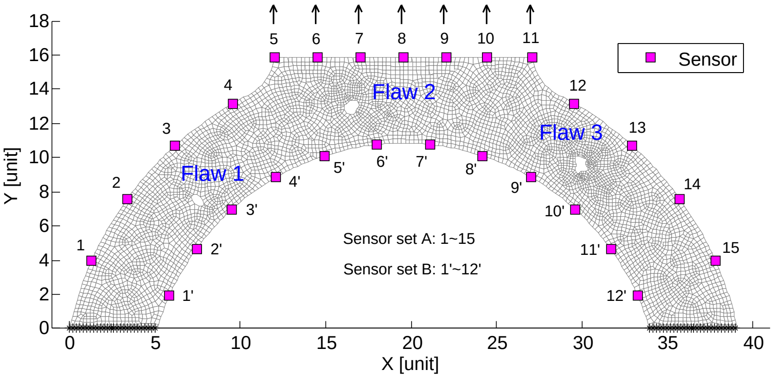


Figure 5.22: Unstructured FEM mesh, loading conditions, sensor placement and flaw locations within an arch-like plate used as the reference solution.

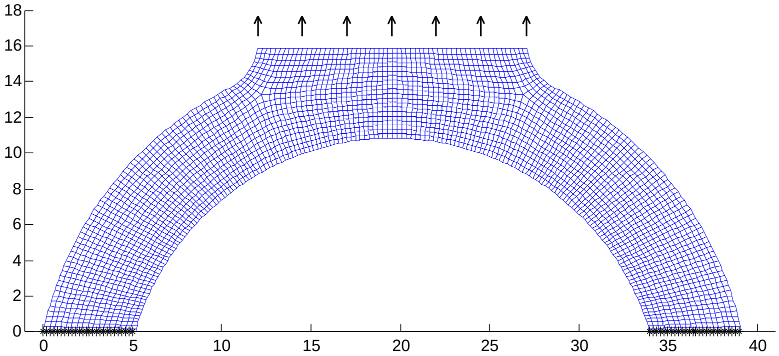


Figure 5.23: Mesh and boundary conditions for the XFEM forward problem (example 5.5.4).

Table 5.5: XFEM-HABC identification results of three non-regular shaped void flaws in example 5.5.4

Flaw parameters		Flaw 1 (Units)	Flaw 2 (Units)	Flaw 3 (Units)	Final objective function value G
Approximate flaw parameter values	x^c	7.5	16.5	30.0	N/A
	y^c	7.5	12.9	9.6	
	r^c	0.4	0.5	0.6	
Number of flaws unknown	x^c	8.7291	15.8439	29.2430	0.0256
	y^c	9.4135	14.0303	9.5034	
	r^c	0.5272	0.4584	0.5738	

FEM with unstructured mesh as reference is equivalent to a model with artificial noise in measurements. This example is the most difficult problem compared with the previous three examples and is a major step toward application of this algorithm to real structures.

In this example, we set $n_t = 4$ so that we have sixteen parameters to be identified during the optimization process; namely, four topological variables and four sets of circular void parameters whose bounds are set as $\{x_1^c, x_2^c, x_3^c, x_4^c\} \in [0.5\ 38.5]$, $\{y_1^c, y_2^c, y_3^c, y_4^c\} \in [0.5\ 26.5]$ and $\{r_1^c, r_2^c, r_3^c, r_4^c\} \in [0\ 1]$. The parameter feasible search space is defined by the plate outer boundaries (edges). Here in this case, the HABC parameters used for simulation are $N_{pop} = 30$, $N_{max} = 400$, $N_{lim} = 40$, $N_{ls} = 15$ and $N_{lsiter} = 20$.

Table 5.5 presents the identification results of the flaw parameters. Figure 5.24 presents the snapshots of the XFEM-HABC convergence process. Figure 5.25(a) and 5.25(b) show the XFEM-HABC convergence curves of the objective function evaluations and of the number of flaws, respectively. It is evident from the identification results that the proposed algorithm is able to truly converge to the correct number of flaws and approximates the irregular shaped flaws quite well. Figure 5.26 describes the comparison of measured strains by the FEM and the simulated/estimated strains by the XFEM. It can be seen from this figure that the differences are very small.

It can be finally concluded that the performance of the proposed XFEM-HABC algorithm is robust and efficient even with the artificial noise in strain measurements caused

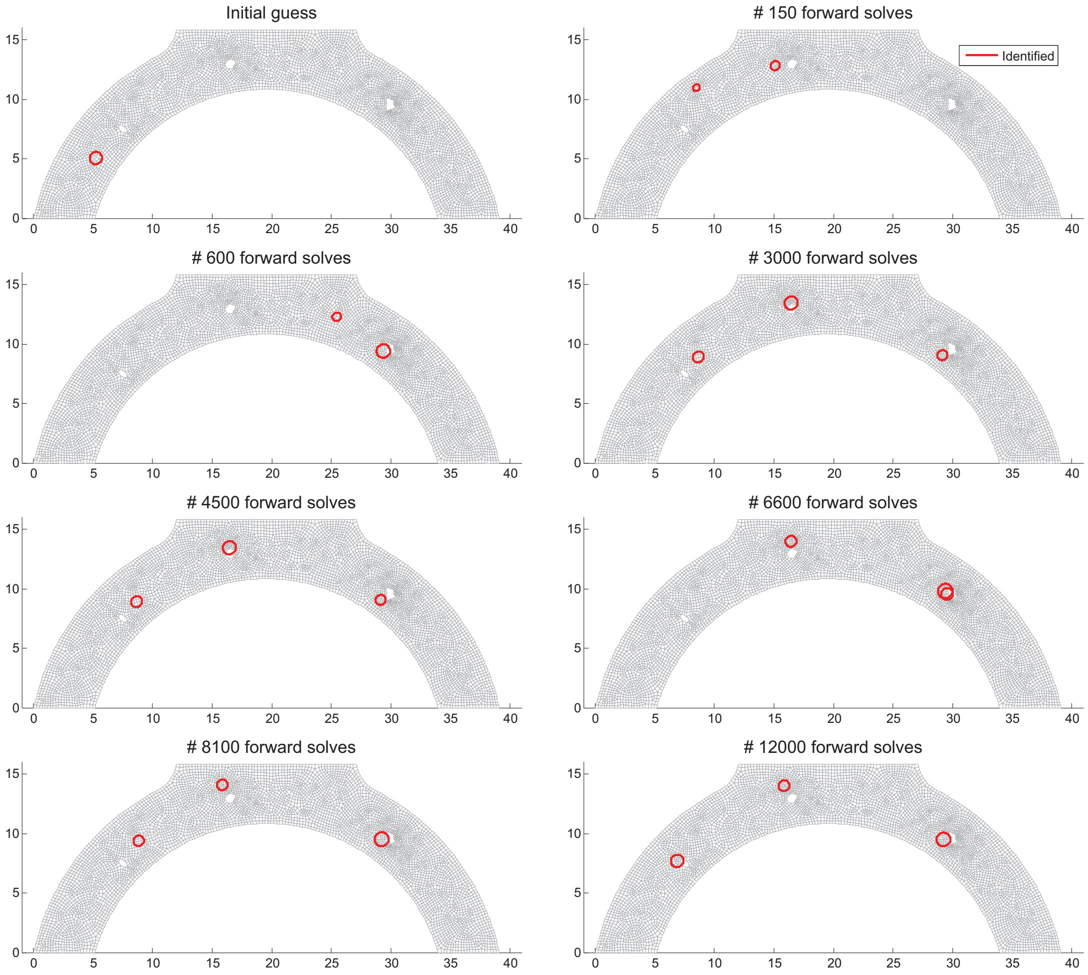
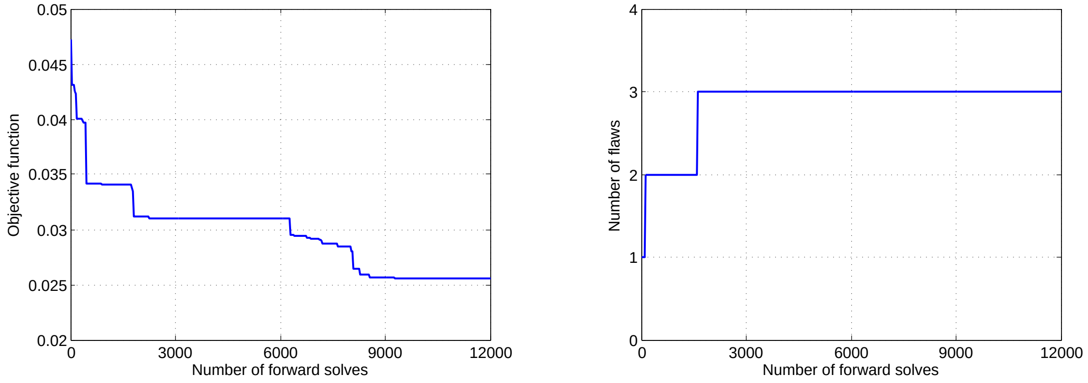


Figure 5.24: Snapshots of the XFEM-HABC evolutionary process in the identification of three non-regular shaped voids (the number of flaws is unknown a priori).

by irregular shaped flaws and by the numerical/modeling difference (error) between the FEM and the XFEM with different meshes. In terms of the CPU time used for each run by the proposed algorithm coded in MATLAB®, it took on average 27 min. in this case. The computational cost can be significantly decreased if the scheme is set to run in parallel on a multi-processor unit coded in more efficient computer languages (e.g., FORTRAN, C, C++, etc.).

A parametric study of the effect of different initial guesses of the number of topological variables (n_t) on multiple flaws detection is also conducted. The initial guess of n_t is varied from two to seven for the simulation. Figure 5.27 shows the convergence plots of



(a) Objective function evaluation

(b) Number of flaws

Figure 5.25: Detection of three non-regular shaped voids under condition of number of flaws unknown: XFEM-HABC convergence.

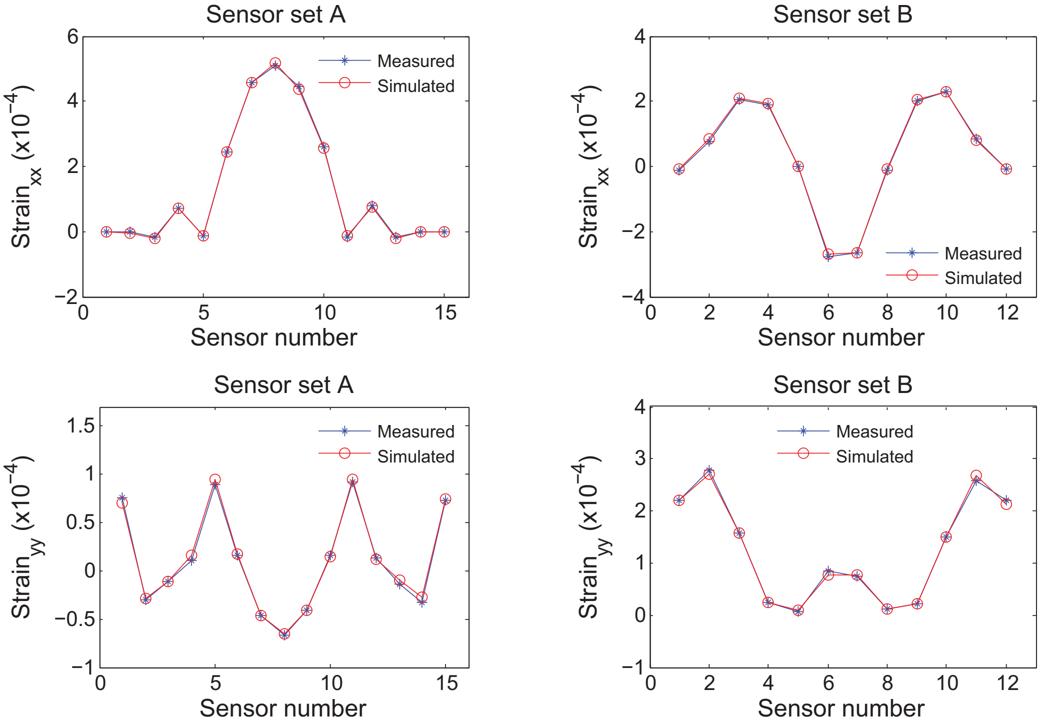
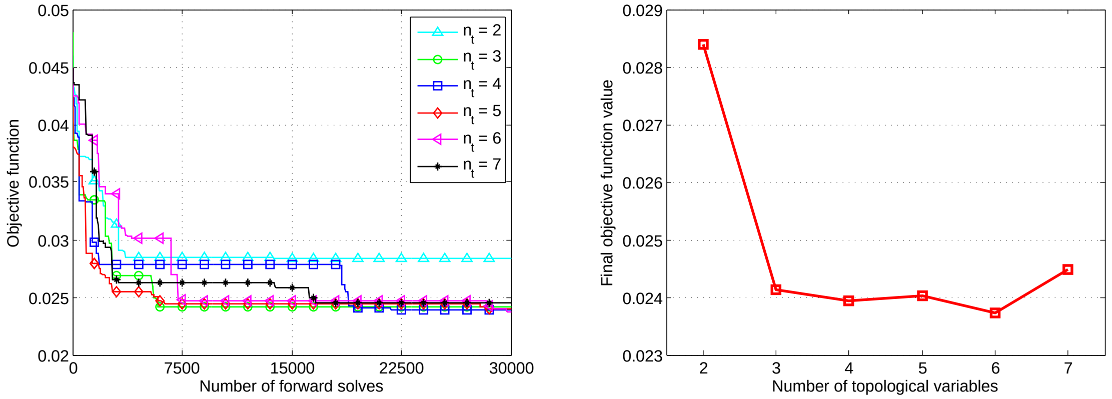


Figure 5.26: Comparison of measured strains and simulated (estimated or predicted) strains.



(a) (b)

Figure 5.27: XFEM-HABC convergence with different initial guesses of the number of topological variables: (a) objective function evaluation and (b) zoomed view of the final objective function values *vs.* the number of topological variables.

the objective function for different initial guesses of n_t. It can be seen that if $n_t \geq 3$ and sufficient number of iterations are used, the objective function is able to converge to a constant value (about 0.024 in this example). Thus, we conclude that the initial guess of the number of topological variables does not have big impact on the proposed XFEM-HABC algorithm for multiple flaws detection, provided that sufficient number of topological variables are adopted.

5.6 Conclusions

This work presents a novel computational scheme, namely XFEM-HABC algorithm, for solving inverse problems for identification of multiple flaws in structures through a limited number of strain measurements. This algorithm further improves the XFEM-GA detection algorithms recently proposed in literature [1–4].

While the previous work only considered quantification of a single flaw, we propose an adaptive algorithm that can identify multiple flaws by introducing a topological variable into the search space which turns on and off flaws during run time. Hence the number of flaws to be detected becomes part of the optimization algorithm.

The proposed HABC algorithm combines the guided-to-best solution updating strategy based ABC algorithm with a local search operator of the Nelder-Mead simplex method (NMSM), which has proven to be more efficient and more effective than the standard ABC and classic GA. During the identification process, each flaw is approximated by a circular void with three variables: center coordinates and radius. Four examples in the numerical experiments are chosen to test various issues associated with the robustness of the algorithm such as mesh types, flaw geometries, multiple flaws, proximity of flaws and known/unknown number of flaws. It can be concluded that the proposed numerical approach preformed quite well producing accurate estimated results. For simple cases with known number of flaws, a few iterations can yield very good identification results. For more realistic cases with unknown number of flaws, convergence becomes more difficult and thus a larger number of iterations is needed to capture both flaw locations/sizes and the quantity of flaws. In the case of irregular shaped flaws, the proposed algorithm converges to the best circular voids that minimize the error of the actual flaws. Moreover, the performance of the XFEM-HABC algorithm is robust even with artificial noise involved in measurements. Overall, the satisfactory results are encouraging for potential implementation of this algorithm in the field of multi-void flaws detection.

Chapter 6

Detection of Multiple Flaws Using Two-Scale Optimization

In this chapter, a novel multiscale algorithm is presented for nondestructive detection of multiple flaws in structures without a priori knowledge about the number and the type of flaws. The key idea is to apply a two-step optimization scheme, where first rough flaw locations are quickly determined and then fine tuning is applied in these localized subdomains to obtain global convergence to the true flaws. This chapter is reproduced from the paper co-authored with Professors Haim Waisman and Raimondo Betti, which was published in the International Journal for Numerical Methods in Engineering [118] and presented in the SPIE Smart Structures/NDE conference [119].

6.1 Introduction

XFEM-based nondestructive identification of multiple flaws/defetcs in structures using a single-scale continuous optimization technique (e.g. the HABC algorithm) has been proposed in Chapter 5. In this approach, topological variables are employed and the number of flaws becomes additional parameters of the optimization. Thus, the detection of the number of flaws and their controlling parameters (positions and sizes) is done in one shot.

The objective of this chapter is to develop a novel multiscale detection scheme that combines heuristic and gradient based optimization, as applied to detection of multiple flaws and/or damage regions and leads to significant improvements in convergence rates compared with the HABC algorithm proposed in the previous chapter.

The basic concept is to apply a two-step optimization framework by combining strengths of heuristic- and gradient-based optimization strategies. The first step employs a discrete type optimization scheme where the optimizer is limited to specific locations and shapes of a flaw, thus converting a continuous optimization problem into a coarser discrete optimization with limited number of choices. In this work, we develop a discrete ABC (DABC) algorithm, which converges to the rough flaw subdoamins. Then a gradient-based optimization of the BFGS method, which is developed specifically for XFEM systems, is applied and leads to fast convergence to the true flaws. In the forward XFEM analyses, both circular and elliptical enrichments are presented to model and approximate the arbitrarily shaped flaws and damage regions.

The chapter is organized as follows. Section 6.2 describes the general formulation of the flaw detection problem. Section 6.3 discusses the multiscale flaw detection framework using hierarchical optimizers. Section 6.4 presents the forward problem solver, *v.i.z.*, the XFEM formulation for multiple void flaws modeling. Section 6.5 proposes the DABC and BFGS algorithms for the two-step discrete and continuous optimization. Section 6.6 presents the sensitivity analysis of flaw parameters in XFEM modeling. Section 6.7 reports on several benchmark numerical examples with increasing complexities, and finally, concluding remarks are provided in Section 6.8.

6.2 The inverse problem formulation

As discussed in Section 5.2, detection of flaws can be regarded as solving an inverse problem, in which an iterative optimization technique can be employed. The iteration proceeds like a "feed-back" process by updating the forward model (see Section 5.2.1) until an optimal set of system parameters are returned (e.g. the objective function is minimized).

Let's recall the inverse problem for flaw detection: Find the inner flaw boundary Γ_v by updating the forward model, given Ω, Γ_t, Γ_u, $\bar{\mathbf{u}}$, $\bar{\mathbf{t}}$, $\mathbf{D}$, and some specific measured response at some given locations (e.g., the displacement field $\tilde{\mathbf{u}}$ in this chapter).

Assuming the inner flaw boundary Γ_v is described by a set of unknown parameters $\boldsymbol{\theta}$, as shown in Equation (5.5), which is adjusted by solving an optimization problem. To this end, the objective function g of the optimization problem is defined as the normalized least square error between the actual measured structural response and the estimated response of such parametric model, *v.i.z.*,

$$g(\boldsymbol{\theta}) = \frac{\left[\mathbf{S}\hat{\mathbf{u}}(\boldsymbol{\theta}) - \tilde{\mathbf{u}}\right]^T \left[\mathbf{S}\hat{\mathbf{u}}(\boldsymbol{\theta}) - \tilde{\mathbf{u}}\right]}{\tilde{\mathbf{u}}^T \tilde{\mathbf{u}}} \tag{6.1}$$

where $\tilde{\mathbf{u}}$ denotes the measured displacement vector; $\hat{\mathbf{u}}$ denotes the simulated displacement vector which is a function of $\boldsymbol{\theta}$ obtained by solving the forward problem; $\mathbf{S}$ represents the information matrix of sensor locations. In general, the identification problem can be summarized as:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in \mathscr{S}} \{g(\boldsymbol{\theta})\} \tag{6.2}$$

where $\mathscr{S}$ is the feasible n-dimensional parameter search space as defined in Equation (5.8) and n represents the number of parameter in $\boldsymbol{\theta}$. More specifically, the dimension of $\mathscr{S}$ is determined by the initial guess on the number of flaws.

It is noteworthy that detection of flaws with circular void enrichments in XFEM, corresponds to a three-parameter optimization problem per each void: center coordinates and radius (x, y, r).

If elliptical void enrichments are employed to approximate flaws (see Figure 6.1), then each flaw is represented by five parameters optimization problem, that is center coordinates, major and minor axes and orientation angle (x, y, a, b, β). In the case that the number of flaws is unknown a priori, an initial guess on the number of flaws n_v, is made. Thus the system parameter vector to be identified, taking the ellipses as an example, becomes

$$\boldsymbol{\theta} = \bigcup_{i=1}^{n_v} \{x_i, y_i, a_i, b_i, \beta_i\} \tag{6.3}$$

where $\cup$ denotes the union operator. The existence of flaws are judged by their sizes. For example, the candidate flaw is activated if its size is bigger than a threshold value such as the mesh size; otherwise, the flaw is deactivated and deleted in the corresponding forward analysis. Alternatively, a topological variable (0/1) based approach proposed in Chapter 5 can also be used to determine the number of flaws in the optimization process. While the topological approach is able to well identify the flaw number as well as the flaw position and size, it introduces additional parameters into the optimization process.

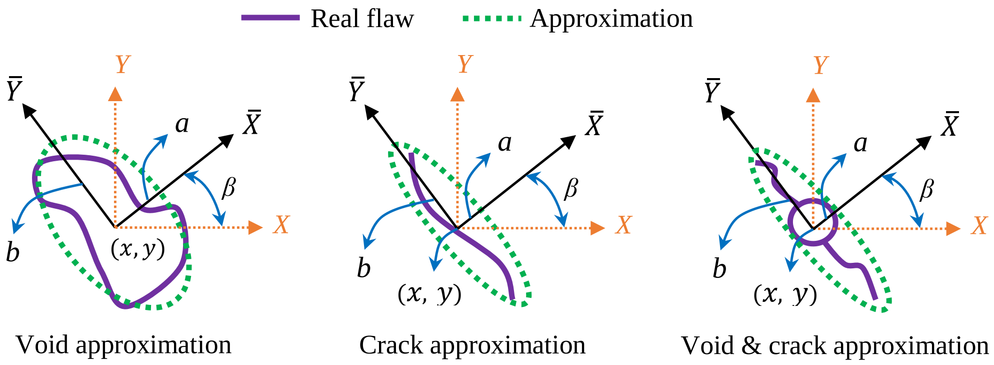


Figure 6.1: Flaws are approximated by elliptical voids.

As discussed in the previous chapter, the computational domain may have complex geometries in which case one would need to ensure the convergence of parameters within the problem domain Ω. Therefore, a feasible search space should be considered as shown in Figure 5.3. In this case, the identification problem becomes an optimization problem with constraints. However, directly restricting the parameters to the feasible domain might be difficult from a computational standpoint since the constraints are not simple and may introduce discontinuity of gradients when gradient-based methods are used. Hence, we replace a constrained minimization problem by an unconstrained minimization process by adding the penalty functions. To wit, the objective function is modified as follows in order to handle the complex constraints:

$$G(\boldsymbol{\theta}) = g(\boldsymbol{\theta}) + c\left[\boldsymbol{\kappa}^T(\boldsymbol{\theta})\boldsymbol{\kappa}(\boldsymbol{\theta}) + \boldsymbol{\rho}^T(\boldsymbol{\theta})\boldsymbol{\rho}(\boldsymbol{\theta})\right] \tag{6.4}$$

where c is a user-defined positive penalty parameter, $G(\boldsymbol{\theta})$ is the penalized objective function, $\boldsymbol{\kappa}$ and $\boldsymbol{\rho}$ are the penalized parameters, $v.i.z.$,

$$\boldsymbol{\kappa}(\boldsymbol{\theta}) = [\boldsymbol{\kappa}(\theta_1), \ldots, \boldsymbol{\kappa}(\theta_i), \ldots, \boldsymbol{\kappa}(\theta_n)]^T \tag{6.5}$$

$$\boldsymbol{\rho}(\boldsymbol{\theta}) = [\boldsymbol{\rho}(\theta_1), \ldots, \boldsymbol{\kappa}(\theta_i), \ldots, \boldsymbol{\kappa}(\theta_n)]^T \tag{6.6}$$

with each component computed by

$$\boldsymbol{\kappa}(\theta_i) = \begin{cases} 0, & \text{if} \quad \theta_i \geq \theta_i^{\min} \\ \theta_i - \theta_i^{\min}, & \text{if} \quad \theta_i < \theta_i^{\min} \end{cases} \tag{6.7}$$

and

$$\boldsymbol{\rho}(\theta_i) = \begin{cases} 0, & \text{if} \quad \theta_i \leq \theta_i^{\max} \\ \theta_i - \theta_i^{\max}, & \text{if} \quad \theta_i > \theta_i^{\max} \end{cases} \tag{6.8}$$

where $\theta_i^{\min}$ is the lower bound of the ith parameter and $\theta_i^{\max}$ is the corresponding upper bound. Since Equations (6.7) and (6.8) have the characteristics of Ramp function (e.g., the product of a linear function and the Heaviside function), the computation of their derivative with respect to the Heaviside center (cut-off point) will introduce singularities when gradient-based methods are used in the optimization. Therefore, we approximate these two functions by the smooth sigmoid function so as to obtain a smooth gradient, e.g., written as follows:

$$\boldsymbol{\kappa}(\theta_i) := (\theta_i - \theta_i^{\min})H(\theta_i^{\min} - \theta_i) \approx (\theta_i - \theta_i^{\min})H^s(\theta_i^{\min} - \theta_i) \tag{6.9}$$

$$\boldsymbol{\rho}(\theta_i) := (\theta_i - \theta_i^{\max})H(\theta_i - \theta_i^{\max}) \approx (\theta_i - \theta_i^{\max})H^s(\theta_i - \theta_i^{\max}) \tag{6.10}$$

where H is the Heaviside function and H^s is the smooth sigmoid function given by

$$H^s(x - x_0) = \frac{1}{1 + \exp\left[-2\alpha(x - x_0)\right]} \tag{6.11}$$

Here, x denotes the generic variable, α is a positive number corresponding to a sharp transition at the cut-off point. The effect of α on the approximation of Heaviside function is shown in Figure 6.2. It is noteworthy that Equations (6.9) and (6.10) are important when one computes the derivative of $G(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$.

Finally, the overall identification problem can be summarized as:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \mathscr{S}} \{G(\boldsymbol{\theta})\} \tag{6.12}$$

which is solved by the two-step optimization approach proposed in Section 6.5.

6.3 The multiscale flaw identification scheme

Herein, we present a multiscale flaw detection scheme (coarse- vs. fine-scale search) to solve the above mentioned inverse problem. The key idea of the multiscale framework includes a two-scale search, namely,

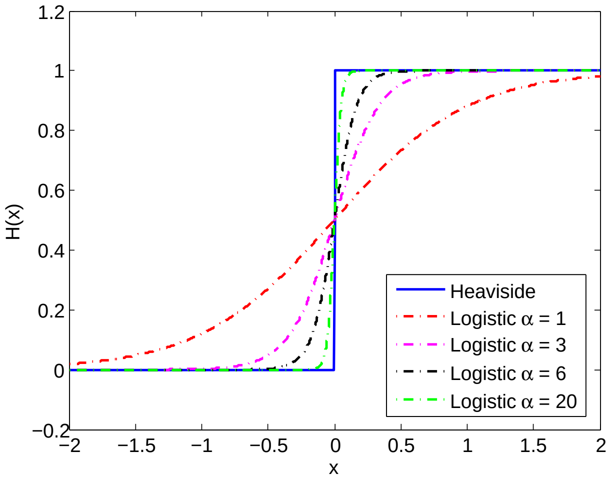


Figure 6.2: The effect of α on the approximation of Heaviside function.

- *Coarse Scale Search*: The search space is firstly partitioned into a finite number of nodes defined by a set of pseudo grids (see Figure 6.3). The candidate solutions are then restricted to the pseudo grid nodes, thus, converting a continuous optimization problem in the entire domain into discrete optimization with a small number of choices. In other words, the coarse scale step becomes an optimization problem with a search over a finite number of solutions in the discrete search space instead of infinite sets of solutions in the continuous space. The discrete optimization problem is solved using a carefully designed DABC algorithm (presented in Section 6.5.1).
  - *Subdomain Convergence*: The result of the coarse scale search is the identification of rough damage regions, which are used to define local subdomains where the possible candidate flaws are located. Each subdomain is defined as a square region centred at the identified flaw center. The edge length of each subdomain is termed the "subdomain band l_{sub}" whose dimension depends on the dimension of the problem domain.
- *Fine Scale Search*: A continuous optimization algorithm (e.g., BFGS presented in Section 6.5.2) is employed to solve the optimization problem within the newly defined subdomains. The identified flaw parameters obtained from the coarse scale search are taken as the initial guess of the fine scale search algorithm, leading to fast convergence.

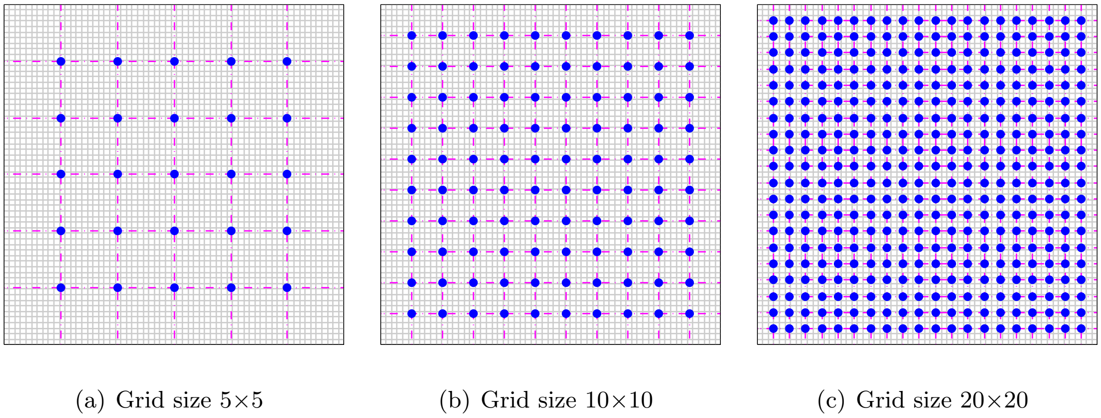
- *Global Convergence*: The algorithm proceeds until global convergence is obtained or a stopping criteria is activated. This process results in the identification of flaw parameters.

To summarize, the first step aims at a coarse/rough scale search over the parameter space. The second step plays a role as a search space reduction so as to improve the optimization convergence. The third step is a fine scale search targeting highly accurate solution. In general, the flaw chart hierarchy of the proposed algorithm is illustrated in Figure 6.4.

For example, Figure 6.5 illustrates a schematic representation of the multiscale algorithm for detection of a single crack embedded in a 2D plate, from scattered plate sensing, pseudo grids generation, coarse scale search, subdomain definition, to fine scale search.

6.4 Solution of the forward problem by XFEM

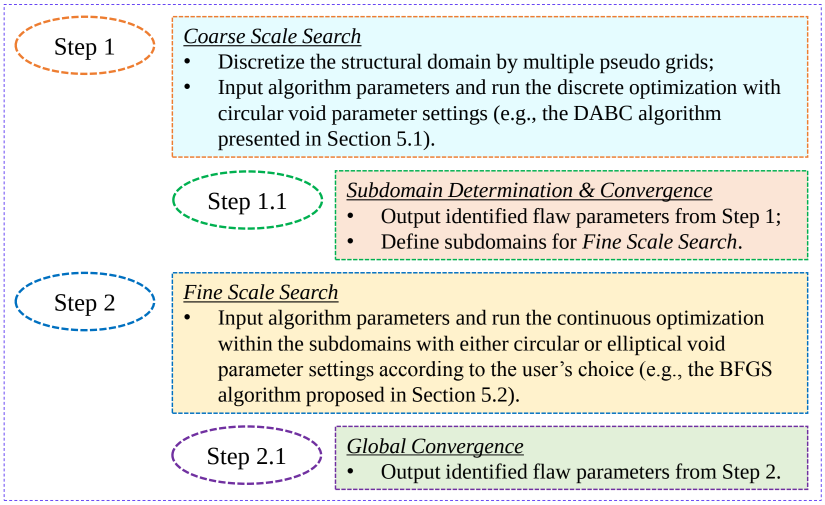
Similar to the previous work in Chapter 5, XFEM is employed as the forward problem solver in the present work. The enriched finite displacement field is given in Equation 5.18. In general, different enrichment functions can be used depending on the problem (e.g., cracks,


(a) Grid size 5×5

(b) Grid size 10×10

(c) Grid size 20×20

Figure 6.3: Example of the pseudo grids with different sizes. The candidate solution of the flaw position in the identification process is fixed on the pseudo grid nodes (blue dots "●"). The background gray grids are the finite element mesh of the physical domain.



Coarse Scale Search

- Discretize the structural domain by multiple pseudo grids;
- Input algorithm parameters and run the discrete optimization with circular void parameter settings (e.g., the DABC algorithm presented in Section 5.1).

Subdomain Determination & Convergence

- Output identified flaw parameters from Step 1;
- Define subdomains for *Fine Scale Search*.

Fine Scale Search

- Input algorithm parameters and run the continuous optimization within the subdomains with either circular or elliptical void parameter settings according to the user's choice (e.g., the BFGS algorithm proposed in Section 5.2).

Global Convergence

- Output identified flaw parameters from Step 2.

Figure 6.4: Flow chart of the multiscale flaw detection methodology.

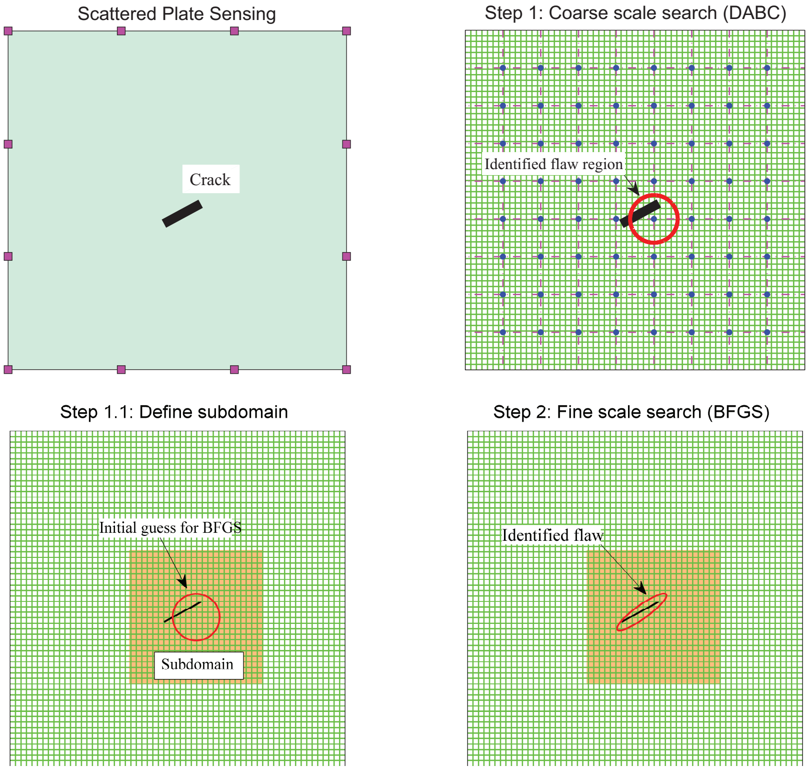


Figure 6.5: A schematic representation of the multiscale flaw detection algorithm.

inclusions, boundary layer solutions or other). In this work, we consider flaws to be in the form of voids, hence the Heaviside function $V(\mathbf{x}, \boldsymbol{\theta})$ is adopted to describe the global enrichment function as illustrated in Equation (5.21).

The function $V(\mathbf{x}, \boldsymbol{\theta})$ contains the information of the number of voids as well as their sizes and shapes. It is noteworthy that the nodes that lie inside the void and whose nodal support is not intersected by the void are dismissed in the calculations. This is done usually by removing the DOFs associated with those nodes from the system of equations and solving for the remaining DOFs. Therefore, the displacement field for void discontinuities can be approximated using the scheme as in [115]:

$$u^h(\mathbf{x}) = \sum_{I \in \mathbb{N}} V(\mathbf{x}, \boldsymbol{\theta}) \mathbf{N}_I(\mathbf{x}) \mathbf{u}_I \tag{6.13}$$

In numerical implementations, the integration points that lie inside the void are directly omitted. Hence, the stiffness matrix derived from the weak form of the system equation can be written as:

$$\mathbf{K}(\boldsymbol{\theta}) = \int_{\Omega} \mathbf{B}^T \mathbf{D} \mathbf{B} V(\mathbf{x}, \boldsymbol{\theta}) d\mathbf{x} \tag{6.14}$$

where $\mathbf{B}$ is the strain-displacement matrix and $\mathbf{D}$ is the stress-strain constitutive matrix.

The minimum signed distance function is used to construct the level set function to capture the shape of the discontinuities and quickly find enriched nodes. For multiple circular voids, the level set function is expressed in Equation (5.23). The parameters $\boldsymbol{\theta}$ describing circular void boundaries can be written as:

$$\boldsymbol{\theta} = \bigcup_{i=1}^{n_c} \{x_i^c, y_i^c, r_i^c\} \tag{6.15}$$

where the subscript i represents the i-th circular void while n_c denotes the total number of circular voids. When the voids are ellipses, the corresponding level set function is [120]:

$$\phi(\mathbf{x}, \boldsymbol{\theta}) = \min \left\{ \left\| \frac{\bar{x}_i^2}{a_i^2} + \frac{\bar{y}_i^2}{b_i^2} \right\| - 1 \right\}_{i=1,2,\ldots,n_e} \tag{6.16}$$

where a_i and b_i are the major and minor axes of the i-th ellipse, respectively. Here, $\bar{\mathbf{x}}_i = (\bar{x}_i, \bar{y}_i)$ denotes the transformed coordinates with the origin at the i-th ellipse center (x_i^c, y_i^c) with orientation β_i along the ellipse major radius. The transformation with respect to the global coordinate system can be written as follows:

$$\bar{x}_i = (x - x_i^c) \cos \beta_i + (y - y_i^c) \sin \beta_i \tag{6.17}$$

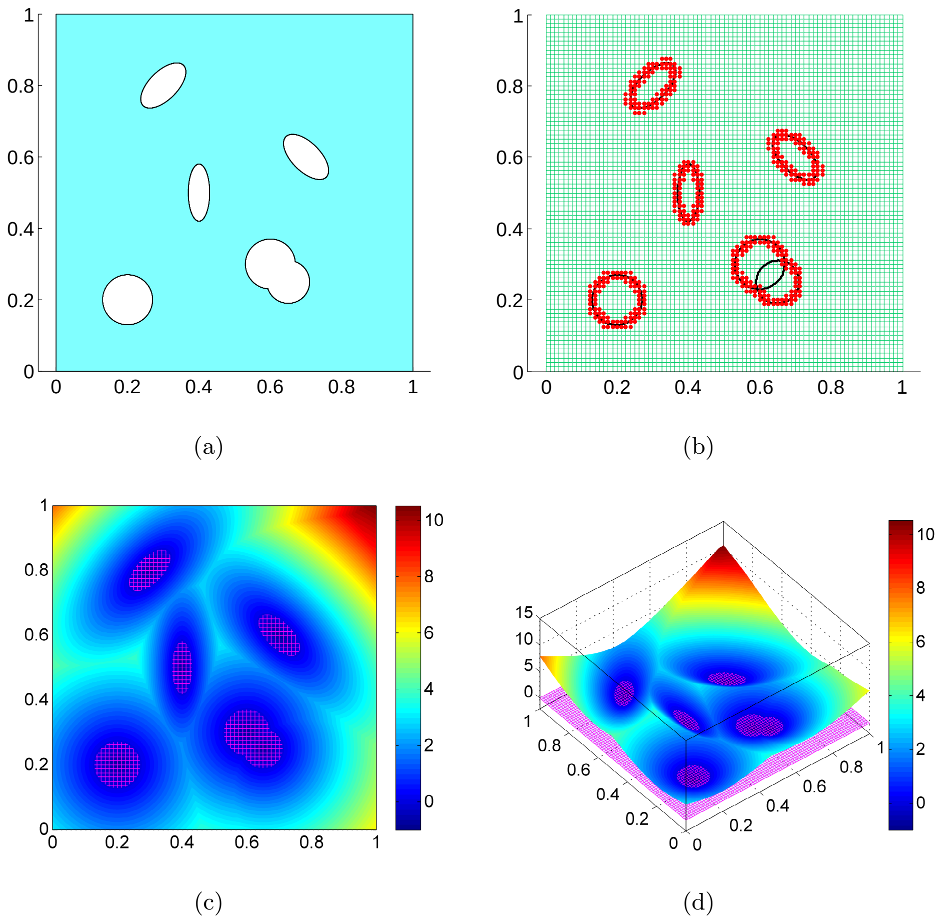
and

$$\bar{y}_i = (x_i^c - x)\sin\beta_i + (y - y_i^c)\cos\beta_i \tag{6.18}$$

The parameters $\boldsymbol{\theta}$ describing the elliptical void boundaries can be written as Equation (6.3).

It is noteworthy that the above level set function for ellipses is not, but similar to, a signed distance function (e.g., the normalized signed distance). As shown in Equation (5.21), since we are only interested in the sign of ϕ and not in its specific value, we adopt the same linear basis functions (shape functions) as those used in regular finite elements.

Figure 6.6 shows the level set and enrichment visualization of a finite element domain with multiple circular/elliptical voids.



(a)

(b)

(c)

(d)

Figure 6.6: Level set and enrichment visualization: (a) flaw boundaries; (b) enriched nodes; (c) contour plot of nodal level set values; and (d) level set function.

6.5 The two-step optimization algorithm

The multiscale flaw detection algorithm described in Section 6.3 consists of hierarchical optimizers such as discrete and continuous optimization strategies. In what follows we summarize the two optimization methods we developed specifically for this multiscale approach, namely, the discrete ABC (DABC) algorithm, and a gradient-based optimizer (the BFGS method).

6.5.1 The discrete ABC (DABC) algorithm

When one deals with discrete optimization problems like integer optimization and topological or binary variable optimization, a discrete optimization algorithm should be employed. In this chapter, we present a discrete ABC algorithm which modifies the continuous ABC algorithm to solve a discrete type optimization problem. In this regard, the solution is no longer free to vary in the computational domain but is limited to specific points in the domain. Thus it can only jump from one discrete point to another. To this end, we modify the solution updating procedure in the standard ABC algorithm:

$$\Theta_{ij}^{up} = \Theta_{ij} + \lfloor 2(\text{rand} - 0.5) \cdot (\Theta_{ij} - \Theta_{kj}) \rceil \tag{6.19}$$

where $i = 1, 2, \ldots, N_{pop}$; $j = 1, 2, \ldots, n$; $\boldsymbol{\Theta}$ is the parameter population; $\boldsymbol{\Theta}^{up}$ is the updated population; k denotes the solution index in the population (k is an integer randomly selected in $[1\ N_{pop}]$ and $k \neq i$); 'rand' denotes a uniformly distributed random number in the range of $[-1,\ 1]$; $\lfloor \cdot \rceil$ represents the nearest integer function, e.g., $\lfloor 1.4 \rceil = 1$ and $\lfloor 1.6 \rceil = 2$.

Nevertheless, as suggested in Section 4.3.1.2, multi-point mutation leads to much more efficient solution updating compared to single-point mutation. Therefore, we propose a DABC algorithm using the new solution updating strategy obtained by modifying the continuous solution updating strategy in Equation 4.19 into an "integer" version:

$$\Theta_{ki}^{up} = \begin{cases} \Theta_{ij}, & \text{if} \quad \text{rand} \geq \rho \\ \Theta_{ij} + \left\lfloor 2(\text{rand} - 0.5) \cdot (\Theta_{ij} - \Theta_{kj}) + \text{rand} \cdot (\theta_i^b - \Theta_{ij}) \right\rceil, & \text{if} \quad \text{rand} < \rho \end{cases} \tag{6.20}$$

The meaning of the indices in Equation (6.20) is identical to those illustrated in Section 4.3.1.2. The purpose of employing the proposed DABC algorithm in this work is to converge to a number of subdomains where the flaws and/or damage regions are located.

6.5.2 The BFGS method

In this work, we employ the well-known gradient-based Quasi-Newton approach, *v.i.z.*, the BFGS method, as our fine scale search operator. The starting point of this algorithm is the initial guess provided by DABC, the coarse scale detection operator. Then BFGS is carried out zooming in to the fine scale features of the flaw.

The description as well as the pseudo code of the BFGS method is given in Section 4.3.2. It is noteworthy that the function gradients used in BFGS are determined by semi-analytical approach proposed in Section 6.6.

Remark. In general, the BFGS method cannot be directly applied to identification of multiple flaws if the number of flaws is unknown beforehand, because of two major reasons: (i) the initial guess of the number of flaws is usually an integer which is larger than the real value; thus the surface of the objective function becomes much more complicated (e.g., highly multi-modal with more local minima compared to the case of known number of flaws) since the optimization search space is augmented. A local search operator such as BFGS generally fails to solve such a complex highly multi-modal optimization problem unless a quite good initial guess is given. (ii) It is difficult to set a good initial guess of the flaw parameters especially when multiple flaws exist. Nonetheless, the powerful global search algorithm is a good choice to provide an admissible initial guess for BFGS.

6.5.3 Criteria for algorithm switching from DABC and BFGS

In order to terminate the DABC algorithm, accept the current best individual as the admissible initial guess for BFGS and thus switch the optimization process from DABC to BFGS, several criteria have been established. To wit, the DABC algorithm is terminated while BFGS is activated if one or more of the following criteria are fulfilled.

(i) The current iteration number exceeds the maximum iteration number: $N_{\text{iter}} > N_{\text{max}}$;

(ii) The objective function hasn't been improved for a number of iterations: $\tilde{N}_{\text{iter}} > N_{\text{fail}}$;

(iii) The objective function value reaches a small threshold value: $G_{\text{iter}} \leq \varepsilon_G$;

(iv) The objective function drops sufficiently, e.g., the ratio of the current objective function value to the initial one reaches a small threshold value: $G_{\text{iter}}/G_0 \leq \varepsilon_d$.

where ε_G and ε_d are the threshold tolerances for criteria (iii) and (iv). It is noteworthy that criterion (i) is the most common one which is widely adopted as the basic stopping criterion in heuristic algorithms. Though criteria (ii)-(iv) are also commonly used, they, in general, play a role as the joint stopping criteria based on criterion (i) rather than being employed alone, because it is quite difficult to set the threshold values. More specifically, the threshold values are problem-dependent. For example, if our demand on the objective function value is beyond that the optimization process could reach, the optimization will never find an end (e.g., in the case that measurement noise and modeling error uncertainties exist). Nonetheless, the combination of these criteria has proven to be successful in heuristic optimization problems [31]. Herein, we take the following "safe" parameter settings as used in the numerical examples: $N_{\text{fail}} = N_{\max}/2$, $\varepsilon_G = 1 \times 10^{-10}$ and $\varepsilon_G = 1 \times 10^{-8}$.

6.6 Sensitivity analysis of flaw parameters in XFEM

When one applies gradient-based optimization techniques such as BFGS to minimize the objective function in Equation (6.4), sensitivities of such a function w.r.t. the parameters should be computed so as to obtain the function gradients, namely,

$$\nabla_{\boldsymbol{\theta}} G(\boldsymbol{\theta}) = \begin{bmatrix} \dfrac{\partial G}{\partial \theta_1} & \dfrac{\partial G}{\partial \theta_2} & \cdots & \dfrac{\partial G}{\partial \theta_i} & \cdots & \dfrac{\partial G}{\partial \theta_{n-1}} & \dfrac{\partial G}{\partial \theta_n} \end{bmatrix}^T \tag{6.21}$$

For a single flaw, Jung *et. al* [111] presented the central finite difference scheme to approximate the objective function gradient with each component computed by

$$\frac{\partial G}{\partial \theta_i} \approx \frac{G(\theta_1, \ldots, \theta_{i-1}, \theta_i + \delta\theta_i, \theta_{i+1}, \ldots, \theta_n) - G(\theta_1, \ldots, \theta_{i-1}, \theta_i - \delta\theta_i, \theta_{i+1}, \ldots, \theta_n)}{\delta\theta_i} \tag{6.22}$$

where $\delta\theta_i$ is an infinitesimal perturbation (e.g., 1×10^{-6}) and n is the number of parameters in $\boldsymbol{\theta}$. The finite difference approach is a simple and straight forward way to compute the stiffness derivative. However, it may introduce errors due to the non-unique way to determine the increment $\delta\theta_i$, since the approximated stiffness derivative depends on a difference between two perturbed matrices [121]. This numerical differentiation also leads to more severe accuracy errors when high order derivatives are computed. In addition, in the process of solving inverse problems, the finite difference remains a significant drain on computational resources to calculate the stiffness derivative (stiffness perturbation and assembling).

Hence, in the current work an analytical scheme is proposed to determine $\partial G/\partial\theta_i$, that is

$$\frac{\partial G}{\partial\theta_i} = \frac{\partial g}{\partial\theta_i} + 2c\left[\boldsymbol{\kappa}^T(\boldsymbol{\theta})\frac{\partial\boldsymbol{\kappa}(\boldsymbol{\theta})}{\partial\theta_i} + \boldsymbol{\rho}^T(\boldsymbol{\theta})\frac{\partial\boldsymbol{\rho}(\boldsymbol{\theta})}{\partial\theta_i}\right] \tag{6.23}$$

where $\partial g/\partial\theta_i$ can be determined by

$$\frac{\partial g}{\partial\theta_i} = \frac{2\left[\mathbf{S}\hat{\mathbf{u}}(\boldsymbol{\theta}) - \tilde{\mathbf{u}}\right]^T \mathbf{S}}{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}}}\frac{\partial\hat{\mathbf{u}}(\boldsymbol{\theta})}{\partial\theta_i} \tag{6.24}$$

and the second and the third parts in Equation (6.23) can be analytically computed, *v.i.z.*,

$$\frac{\partial\boldsymbol{\kappa}(\boldsymbol{\theta})}{\partial\theta_i} = \left[0,\ldots,\frac{\partial\boldsymbol{\kappa}(\theta_i)}{\partial\theta_i},\ldots,0\right]^T \tag{6.25}$$

$$\frac{\partial\boldsymbol{\rho}(\boldsymbol{\theta})}{\partial\theta_i} = \left[0,\ldots,\frac{\partial\boldsymbol{\rho}(\theta_i)}{\partial\theta_i},\ldots,0\right]^T \tag{6.26}$$

where the derivatives $\partial\boldsymbol{\kappa}(\theta_i)/\partial\theta_i$ and $\partial\boldsymbol{\rho}(\theta_i)/\partial\theta_i$ can be expressed as:

$$\frac{\partial\boldsymbol{\kappa}(\theta_i)}{\partial\theta_i} = \frac{1 + \left[1 - 2\alpha\left(\theta_i - \theta_i^{\min}\right)\right]\exp\left[2\alpha(\theta_i - \theta_i^{\min})\right]}{\left\{1 + \exp\left[2\alpha(\theta_i - \theta_i^{\min})\right]\right\}^2} \tag{6.27}$$

and

$$\frac{\partial\boldsymbol{\rho}(\theta_i)}{\partial\theta_i} = \frac{1 + [1 + 2\alpha\left(\theta_i - \theta_i^{\max}\right)]\exp\left[-2\alpha(\theta_i - \theta_i^{\max})\right]}{\{1 + \exp\left[-2\alpha(\theta_i - \theta_i^{\max})\right]\}^2} \tag{6.28}$$

In order to further derive Equation (6.24), we need to first compute $\partial\hat{\mathbf{u}}(\boldsymbol{\theta})/\partial\theta_i$. To this end, consider the general linear finite element equation $\mathbf{K}(\boldsymbol{\theta})\hat{\mathbf{u}}(\boldsymbol{\theta}) = \mathbf{f}$ with $\mathbf{f}$ being the constant force vector, and take the chain rule w.r.t. θ_i so that

$$\frac{\partial\mathbf{K}(\boldsymbol{\theta})}{\partial\theta_i}\hat{\mathbf{u}}(\boldsymbol{\theta}) + \mathbf{K}(\boldsymbol{\theta})\frac{\partial\hat{\mathbf{u}}(\boldsymbol{\theta})}{\partial\theta_i} = \mathbf{0} \tag{6.29}$$

The substitution of Equation (6.29) into Equation (6.24) yields:

$$\frac{\partial g}{\partial\theta_i} = \frac{2\left[\tilde{\mathbf{u}} - \mathbf{S}\hat{\mathbf{u}}(\boldsymbol{\theta})\right]^T \mathbf{S}}{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}}}\left[\mathbf{K}^{-1}(\boldsymbol{\theta})\frac{\partial\mathbf{K}(\boldsymbol{\theta})}{\partial\theta_i}\hat{\mathbf{u}}(\boldsymbol{\theta})\right] \tag{6.30}$$

where $\partial\mathbf{K}(\boldsymbol{\theta})/\partial\theta_i$ is the stiffness derivative w.r.t. flaw parameters. Herein, we follow the explicit derivation introduced in [121] and compute the stiffness derivative by a semi-analytical form as follows.

$$\frac{\partial\mathbf{K}(\boldsymbol{\theta})}{\partial\theta_i} = \mathop{\mathcal{A}}_{e\in\eta}\int_{\Omega^e}\mathbf{B}^{eT}\mathbf{D}^e\mathbf{B}^e\frac{\partial V(\mathbf{x}^e,\boldsymbol{\theta})}{\partial\theta_i}d\mathbf{x}^e \tag{6.31}$$

where e denotes individual elements and $\mathcal{A}$ is the assembly operator over a set of elements η. Similar to Equations (6.9) and (6.10), $V(\mathbf{x}, \boldsymbol{\theta})$ can be also approximated by the smooth logistic function, expressed as:

$$\bar{V}(\mathbf{x}, \boldsymbol{\theta}) := \frac{1}{1 + \exp\left[-2\alpha\phi(\mathbf{x}, \boldsymbol{\theta})\right]} \tag{6.32}$$

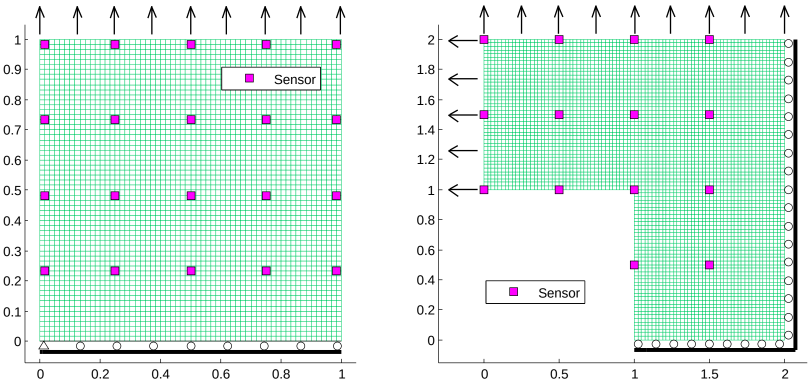
Therefore, we have

$$\frac{\partial \bar{V}(\mathbf{x}, \boldsymbol{\theta})}{\partial \theta_i} = \frac{-2\alpha \exp\left[-2\alpha\phi(\mathbf{x}, \boldsymbol{\theta})\right]}{\left\{1 + \exp\left[-2\alpha\phi(\mathbf{x}, \boldsymbol{\theta})\right]\right\}^2} \frac{\partial \phi(\mathbf{x}, \boldsymbol{\theta})}{\partial \theta_i} \tag{6.33}$$

where the derivative $\partial\phi(\mathbf{x}, \boldsymbol{\theta})/\partial\theta_i$ can be determined using the central difference (see Equations (6.21) and (6.22)). The reason why the central difference is used is that it is quite difficult to analytically obtain the close form of the level set derivative due to the "min" function (see Equations (5.23) and (6.16)) when multiple flaws exist. Moreover, since $\phi(\mathbf{x}, \boldsymbol{\theta})$ is a $\mathcal{C}^0$ continuous function, the central difference would be a good estimator of its derivative. In addition, the computational time is negligible. The semi-analytical method requires only one time stiffness assembly to compute the stiffness gradient and doesn't necessarily require solving the entire system compared to the central difference method used by Jung *et. al* [111]. Thus, the explicit form of ∇G can be determined following the aforementioned semi-analytical approach.

6.7 Numerical experiments

The performance of the proposed algorithm for detection of multiple flaws and/or damage regions is illustrated in this section by four numerical examples. In the first three examples, we consider a 2D square plate of dimension 1 × 1 unit as shown in Figure 6.7(a) where the flaws and damage regions are hosted, while an L-shape plate is tested in the fourth example as illustrated in Figure 6.7(b). We assume the examples are plane-stress problems with the material properties $E = 1 \times 10^{11}$ and $\nu = 0.3$. The square plate is subjected to a static uniform traction load on the top edge and is fixed by rollers on the bottom edge. Sensors are uniformly distributed in the plate domain to record the static displacement response. A consistent set of uniform structured mesh of 60×60 elements is used in the forward XFEM solves. The L-shape plate is subjected to static uniform traction forces on



(a) Square plate (b) L-shape plate

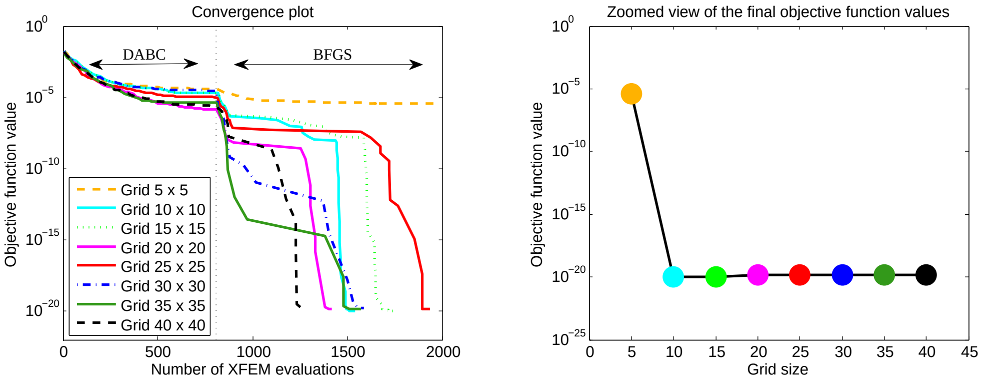
Figure 6.7: Mesh, sensor placement, loading and boundary conditions of the 2D plates.

the top and the left edges. The bottom and the right edges are fixed on rollers. The size of the elements in the mesh are chosen to be 0.05 units. Since the number of flaws is unknown beforehand, an initial guess value of 5 is adopted in all the following numerical analyses. In the identification process, the candidate/identified flaw is generated if its size is larger than the elemental mesh size. Moreover, we take $\alpha = 1 \times 10^3$ in Equations (6.9) and (6.10) and the penalty parameter $c = 1$ to construct the objective functions in all the simulations. To avoid solution uncertainty, each optimization problem is repeated twenty times in the statistical simulations. Either the mean result of the half best solutions or the unique best solution with the minimum objective function value is reported and analyzed.

The measurements used in the simulation are obtained by solving a reference problem through either the XFEM code or the standard FEM code. The numerical analyses are programmed in MATLAB® (The MathWorks, Inc., MA, USA) on a standard Intel (R) Core (TM) i5-3570K 3.40 GHz PC with 16G RAM.

6.7.1 Parametric study on the effect of pseudo grid size

We first study on the effect of pseudo grid size on the flaw detection accuracy in Step 1: the coarse scale search with DABC (the discrete optimization step) so as to determine an optimal value of the grid size. Here, we consider a relatively simple problem, e.g., detection of a single circular void with the center at (0.657, 0.613) units and the radius 0.05 units.



(a) Convergence lines

(b) Final objective function value

Figure 6.8: Parametric convergence studies of the objective function with different pseudo grid sizes using the proposed XFEM-DABC-BFGS algorithm. Note that the gradients in BFGS are computed by using the proposed semi-analytical form.

Nevertheless, the number of flaws is not known a priori. We implement the DABC algorithm to detect such a flaw in Step 1 given different pseudo grids, e.g., with the mesh size of 5×5, 10×10, 15×15, 20×20, 25×25, 30×30, 35×35, 40×40, respectively. The mean result of ten best statistical runs is summarized here for evaluation of grid size effect. Figure 6.8 shows the parametric convergence lines of the objective function for different grid sizes. It can be seen that the algorithm is able to converge to some certain value of the same order (e.g., 1×10^{-20}) if the grid size is much smaller than the flaw size; otherwise, the algorithm fails to accurately detect the flaw (e.g., in the case 5×5 in Figure 6.8, the final objective function value is 6.2×10^{-6} which is much larger).

This study suggest that the pseudo grid size used in Step 1 should not be too large. Nonetheless, an optimal value of the grid size 20×20 is used as a representative size in all the examples, though this optimal grid size is of an empirical value.

6.7.2 Detection of two circular void flaws

The performance of the proposed identification scheme is studied on the detection of two circular voids. As presented in Chapter 5 (Section 5.5.2), two cases regarding the flaw proximity have been considered here to validate the algorithm convergence: (i) two far flaws with $(x_1, y_1, r_1) = (0.3, 0.8, 0.04)$ and $(x_2, y_2, r_2) = (0.7, 0.4, 0.04)$; (ii) two close flaws with $(x_1, y_1, r_1) = (0.45, 0.65, 0.04)$ and $(x_2, y_2, r_2) = (0.55, 0.55, 0.04)$. The parameter bounds used in Step 1 optimization are defined as: $x_i \in [0.05\ 0.95]$, $y_i \in [0.05\ 0.95]$ and $r_i \in [0\ 0.2]$, where $i = 1$, 2. The DABC parameter settings used here are given as follows: $N_{pop} = 10$, $N_{\text{lim}} = 30$, $N_{\text{max}} = 80$ and $n = 15$. The parameters in BFGS are $\varepsilon = 1 \times 10^{-8}$ and L_{max} = 200. The subdomain band is taken $l_{\text{sub}} = 0.3$ units. In these examples, the reference measurements used in identification are generated by XFEM.

The purpose of this example is to study the convergence behavior of the proposed multiscale optimization approach as compared to a single step continuous optimization scheme such as GA and HABC. Circular voids are consistently used to model the flaws. Hence, we have run the GA algorithm, the HABC algorithm, and the hybrid DABC-BFGS algorithm together with the XFEM forward solver to identify these two flaws.

6.7.2.1 Two far flaws.

We consider the detection of two far flaws in this case. Figure 6.9 illustrates the comparison of the statistically mean objective convergence lines using the above mentioned optimization methods. It can be observed that HABC performs better than the classic GA algorithm with higher convergence efficiency, while the multiscale DABC-BFGS approach shows a much superior convergence than all other methods. The combination of coarse and fine scale search (e.g., combined discrete and continuous optimization methods) successfully yields a much more precise identification result than those of continuous GA and HABC. With the initial guess assigned from DABC, the BFGS method is able to converge to a much smaller objective function value (1.34×10^{-21}), meanwhile, yielding a faster convergence rate. Overall, the DABC-BFGS method is superior to any of the continuous approaches such as GA or HABC.

It is also noteworthy from Figure 6.9 that the proposed semi-analytical method for gradient computation improves the convergence efficiency of BFGS, which requires much less XFEM forward analyses than those of the central difference based method, though their final identification accuracies are identical. Moreover, we also test the effect of the penalty parameter c on the algorithm convergence (e.g., by setting $c = 1, 1 \times 10^3$ and 1×10^6,

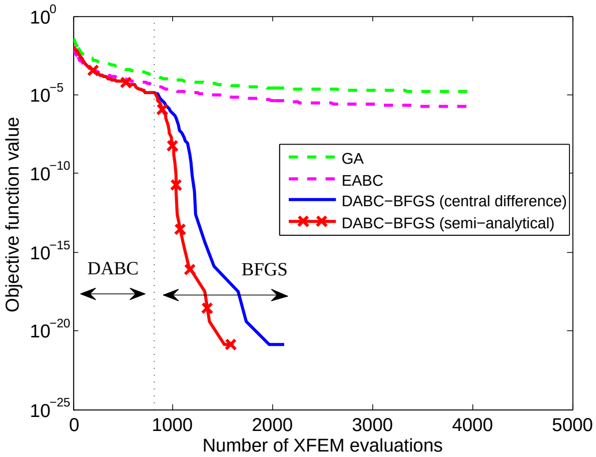


Figure 6.9: Convergence comparison of different optimization methods. Note that the gradients in BFGS are computed by either the central difference method or the proposed semi-analytical form.

Table 6.1: Identified flaw parameters in the case of noise corruption

Noise level (%)	Flaw 1 (Units)			Flaw 2 (Units)		
	x_1	y_1	r_1	x_2	y_2	r_2
0	0.3000	0.8000	0.0400	0.7000	0.4000	0.0400
5	0.3151	0.7999	0.0406	0.7267	0.3889	0.0322
10	0.3235	0.7584	0.0374	0.7293	0.4062	0.0385
True value	0.3	0.8	0.04	0.7	0.4	0.04

respectively). It is noted that identical convergence lines are obtained. Hence, it appears that c doesn't have an explicit impact on the convergence, and it is sufficient to choose $c \geq 1$. Figure 6.10 depicts a number of typical snapshots of the XFEM-DABC-BFGS evolutionary identification process for the target of two far voids. It is obvious that the subdomains, the flaw sizes and positions as well as the number of flaws are identified with quite high accuracy (e.g., the maximum relative error of the flaw parameters is less than 1×10^{-6}).

The effect of measurement noise on the identification result is also analyzed here. Table 6.1 shows the mean XFEM-DABC-BFGS identification results obtained from 100 Monte Carlo simulations, using two separate sets of measurements corrupted with typical levels of noise, e.g., 5% and 10% root-mean-square (RMS) noise ratios to the noise-free signal. The Gaussian zero-mean white noise sequence is employed as the added noise signal. Figure 6.11 shows the identified flaws as well as the probability density functions (PDFs) for the identification error of each flaw. Similar to Equation (5.27), a norm error is defined to evaluate the Monte Carlo simulation results, written as

$$e_i = 1 - \frac{\|\{x_i, y_i, r_i\}^{id}\|}{\|\{x_i, y_i, r_i\}^{exact}\|} \tag{6.34}$$

where i denotes the flaw index, (x_i, y_i) are the circular center coordinate, r_i is the radius, the superscript id represents the identified parameter vector, and the superscript $exact$ represents the exact parameter vector. The operator $\| \cdot \|$ denotes the L_2 norm. Though the identification accuracy is slightly worse than that of the noise-free case, the discrepancy between the identified and the true flaws is quite small which is negligible. The narrow PDFs of the error indicate that the proposed algorithm is fairly robust against measurement noise.

6.7.2.2 Two close flaws.

In this case, we test the proposed approach by identification of two close flaws which is assumed to be more difficult for algorithm convergence than the detection of two far flaws. This might be because the multi-modal surface of the objective function becomes more apparent in the neighborhood of close flaws, which, as a result, leads to much more local minima in this region making the optimization problem more complicated. Another reason is that pronounced non-uniqueness is more likely to occur in the close flaws scenario. For example, if the flaws are close enough, they could be approximated by a single flaw instead of two. This phenomenon is finally evidenced by the identification results (e.g., see Figure 6.13(a)).

Figure 6.12 shows both the local and the global convergence lines in the iterative identification process. It is observed that the proposed XFEM-DABC-BFGS converges to a global minimum (1.46×10^{-20}) provided that the DABC algorithm assigns a good initial guess to BFGS; otherwise, the algorithm converges to a local minimum (2.01×10^{-5}). Figure 6.13 depicts the identified two close flaws obtained from the local convergence and the global convergence, respectively. Furthermore, Figure 6.14 illustrates the snapshots of the XFEM-DABC-BFGS identification process with a global convergence behavior. It is evident from

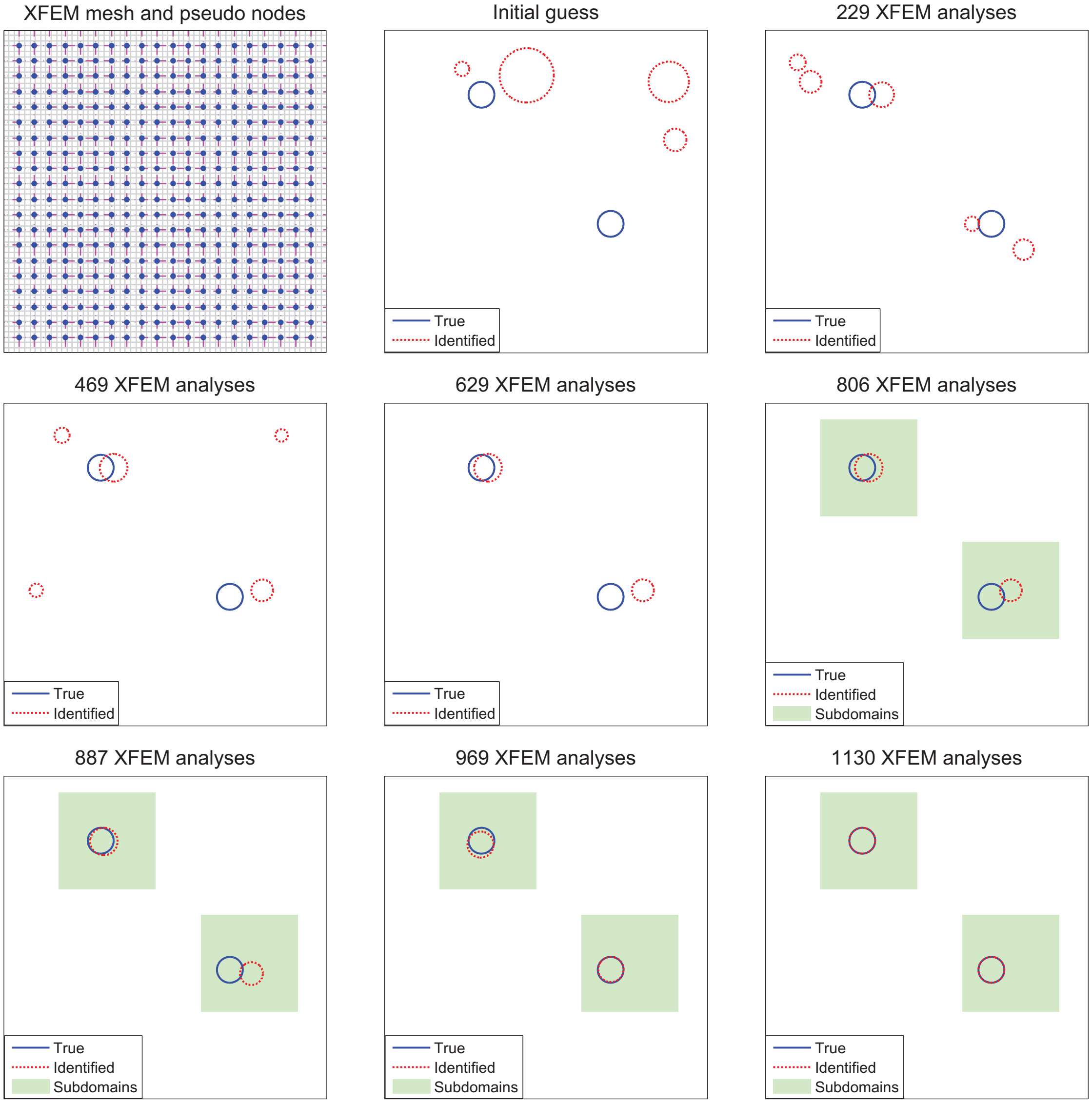
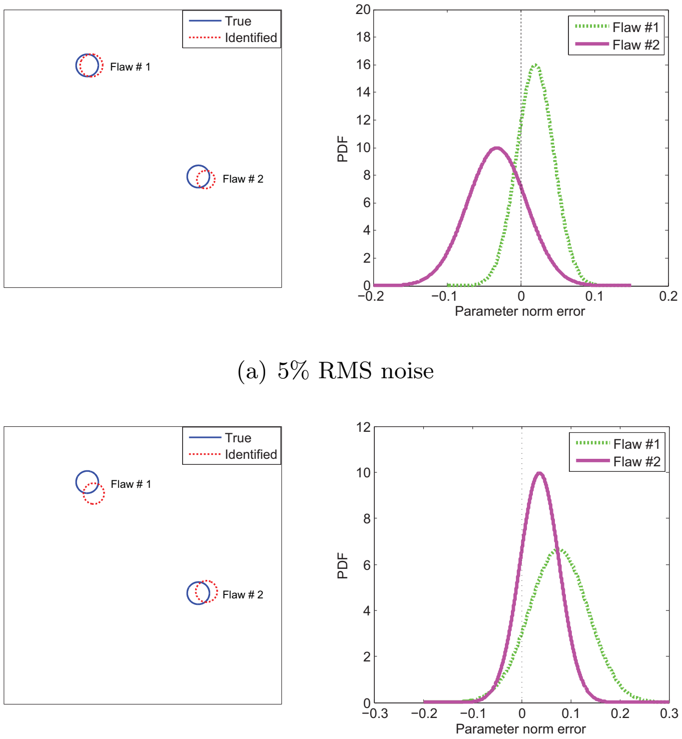


Figure 6.10: Snapshots of the XFEM-DABC-BFGS evolutionary identification process for the target of two far voids. The gradients in BFGS here are computed using the semi-analytical form. Note that the subdomains are determined based on the result of the 494th XFEM analysis. BFGS is then implemented taking such a result as an initial guess.



(a) 5% RMS noise

(b) 10% RMS noise

Figure 6.11: A typical example of the identified two far flaws under noise corruption. The PDFs represent the distribution of the parameter identification errors.

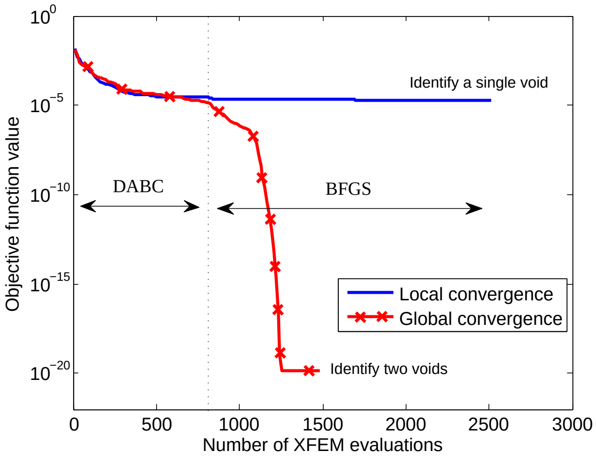
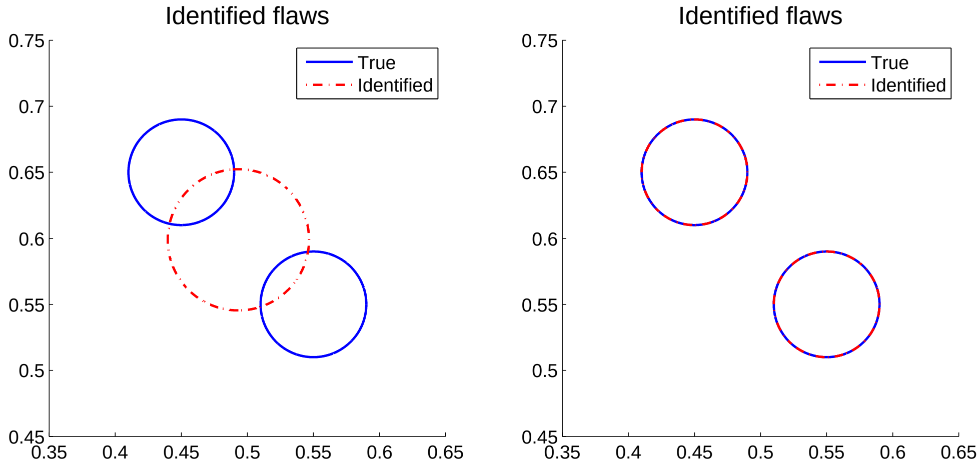


Figure 6.12: XFEM-DABC-BFGS convergence for identification of two close flaws (local vs. global minima).



(a) Local convergence solution (b) Global convergence solution

Figure 6.13: Identified two close flaws using the XFEM-DABC-BFGS scheme (local vs. global convergence solution). Note that the gradients in BFGS here are computed using the semi-analytical form.

this figure that the number of flaws keeps switching on and off until a favorable estimate is reached. The subdomains are defined in the right regions. The two close flaws are finally well detected, whose identified parameters have extraordinarily accurate positions and sizes.

6.7.3 Detection of three damaged regions

To assess the robustness of the proposed XFEM-DABC-BFGS algorithm for detection of irregular-shaped flaws, we study detection of three damaged regions consisting of both irregular-shaped voids and arbitrary shaped cracks as shown in Figure 6.15(a).

Flaw region 1 is an L-shape void, flaw region 2 is the combination of a void and two cracks, and flaw region 3 consists of a group of cracks. The reference measurements are generated by the regular FEM with a fine unstructured mesh using ABAQUS (see Figure 6.15(b)). The contour plot of the von Mises stress field is shown in Figure 6.15(b). The irregular-shaped flaws and the use of FEM with unstructured mesh are equivalent to introducing modeling error which is somewhat similar to “artificial noise” in measurements. We emphasize that due to the difference between the reference model (unstructured FEM

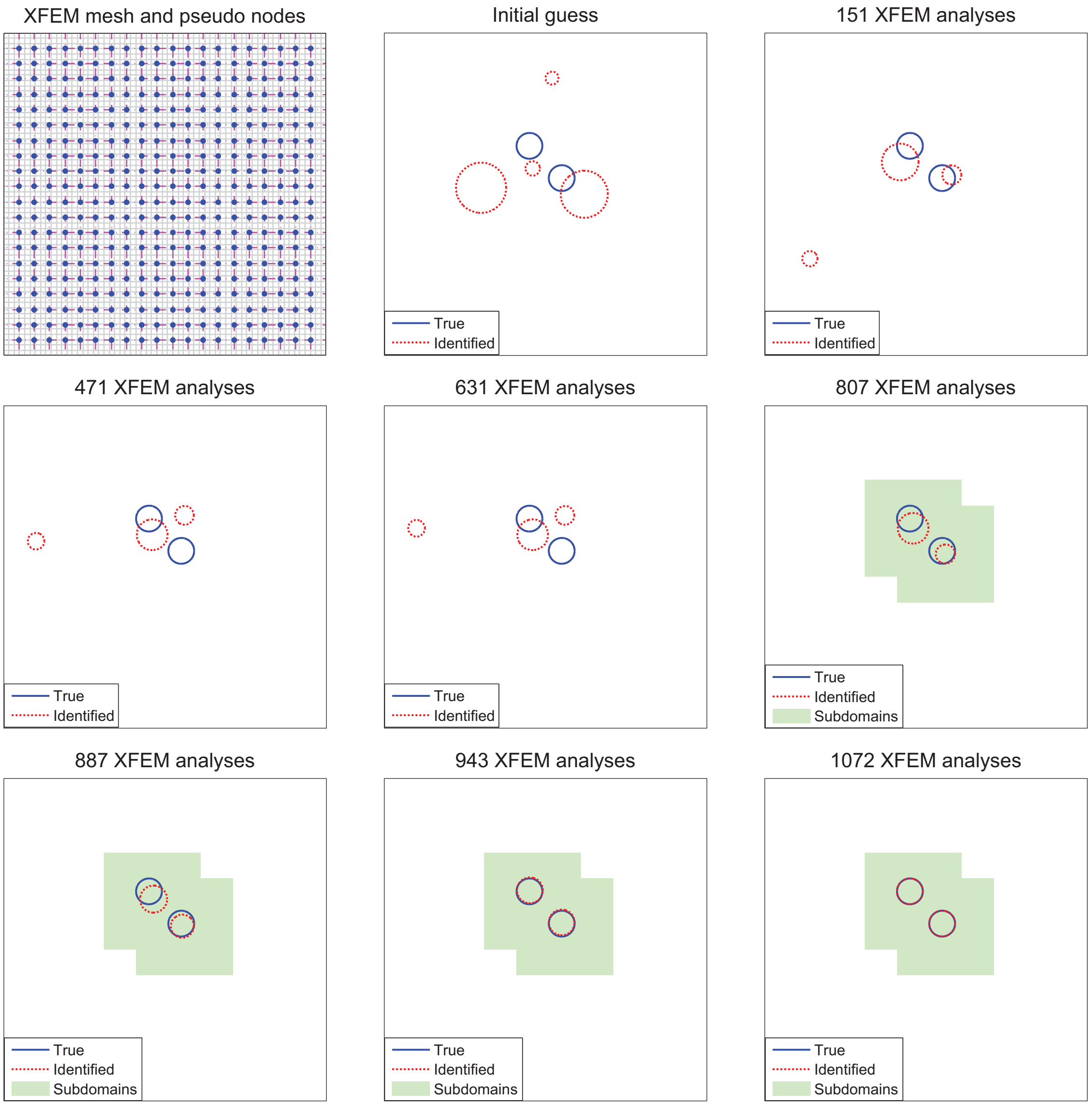
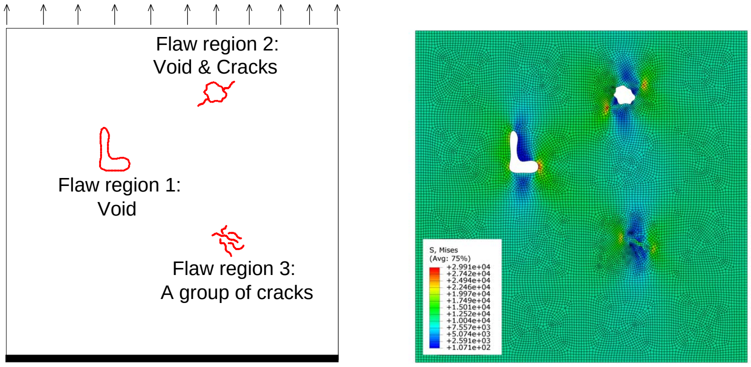


Figure 6.14: Snapshots of the XFEM-DABC-BFGS evolutionary identification process for the target of two close voids. The gradients in BFGS here are computed using the semi-analytical form. Note that the subdomains are determined based on the result of the 491st XFEM analysis. BFGS is then implemented taking such a result as an initial guess.



(a) Sketch of the flaws

(b) FEM mesh and von Mises

Figure 6.15: The damage has three regions including both cracks and voids. A set of fine unstructured mesh is used to generate the reference measurements by standard FEM.

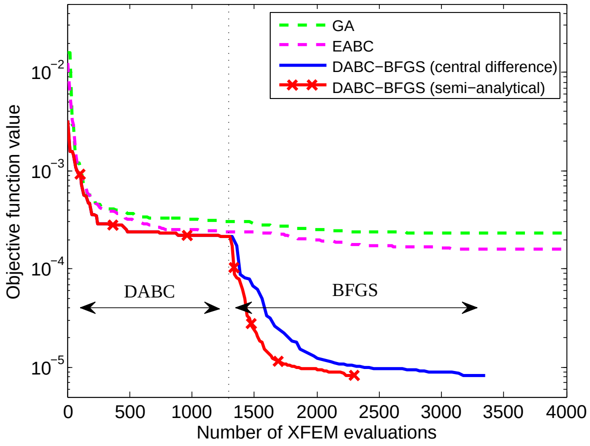


Figure 6.16: XFEM-DABC-BFGS convergence for identification of three damage regions. Note that The gradients in BFGS here are computed using the semi-analytical form.

modeled with ABAQUS) and the forward model (structured XFEM), we avoid the so called “inverse-crime”, which is defined when the same model is used to synthesize as well as invert data in an inverse problem [1].

The DABC parameter settings used for simulation in this example are $N_{pop} = 10$, $N_{\lim} = 30$, $N_{\max} = 120$ and $n = 15$. The BFGS parameters, the subdomain band and the parameter bounds are identical to those used in Example 6.7.2. Both GA and HABC algorithms are also tested for comparison purpose. Figure 6.16 presents the iterative convergence lines for

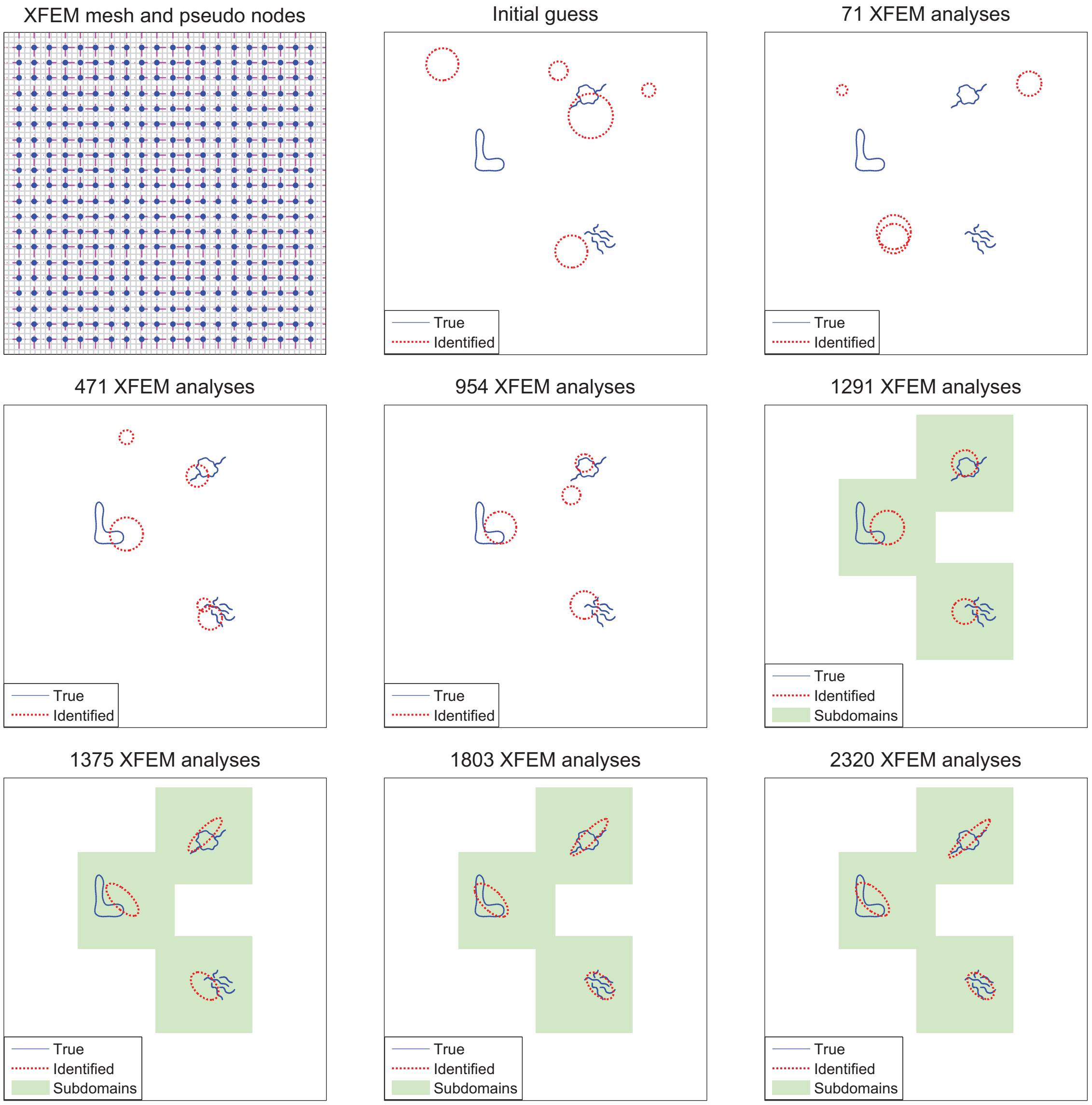


Figure 6.17: Snapshots of the XFEM-DABC-BFGS evolutionary identification process for the target of three damage regions. The gradients in BFGS here are computed using the semi-analytical form. Note that the subdomains are determined based on the result of the 1007th XFEM analysis. BFGS is then implemented taking such a result as an initial guess. Elliptical voids are used in BFGS to approximate the damage regions.

identification of three damage regions. It can be observed that the XFEM-DABC-BFGS algorithm based on semi-analytical gradients has the best convergence behavior. Figure 6.17 depicts a set of snapshots of the XFEM-DABC-BFGS evolutionary identification process.

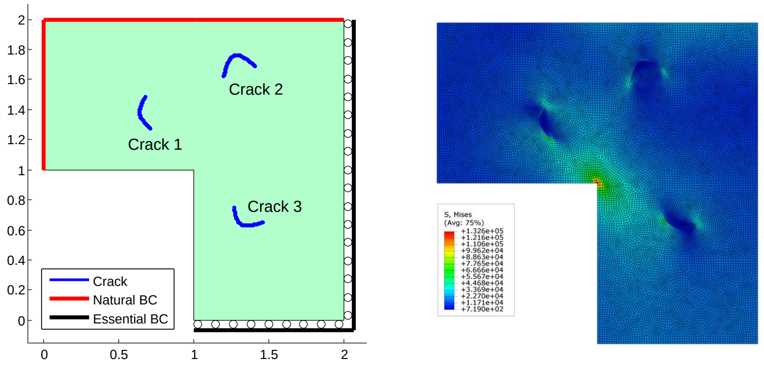
It is noteworthy in the XFEM-DABC-BFGS algorithm that the circular enrichment based XFEM is used in Step 1 to model the flaws as the DABC optimization proceeds. Once the search space reduction is applied and the subdomains are well determined in Step 1.1, enrichments of the XFEM solver are switched to ellipses in Step 2 and semi-analytical gradient-based BFGS is employed to solve the optimization problem within the subdomains.

It is found that the proposed algorithm is able to converge to the correct number of damage regions as well as approximate the flaws and damage regions quite well. Especially, in the identification of flaw region 3, the group of cracks can be accurately estimated using a single ellipse. It is thus concluded that the proposed algorithm is robust and insensitive to artificial noise due to uncertainties associated with flaw shape. Each XFEM analysis takes about 0.2 seconds. The total time of a single semi-analytical XFEM-DABC-BFGS run for this example is less than 15 minutes.

6.7.4 Detection of three curved cracks in an L-shape plate

To further understand the modeling error effects on the convergence of XFEM-DABC-BFGS, an L-shape plate with three "banana-shaped" cracks, as shown in Figure 6.18(a), is investigated. Similar to the example in Section 6.7.3, the reference measurements are obtained by standard FEM with unstructured mesh using ABAQUS so as to consider real scenarios with modeling error uncertainties. Therefore, an "inverse crime" is also avoided in this example. The FEM mesh as well as the von Mises contour plot is given in Figures 6.18(b).

The DABC parameters used in this example are $N_p = 10$, $N_{\text{lim}} = 30$, $N_{\text{max}} = 120$ and $n = 15$. The BFGS parameters are same as those in Example 6.7.2. The subdomain band is taken $l_{\text{sub}} = 0.6$ units. Similar to the above examples, the GA, HABC and DABC-BFGS algorithms are tested in this example. In the multiscale flaw detection process, circular enrichments are first used to approximate the cracks in Step 1 (DABC) and then elliptical enrichments are employed in Step 2 (BFGS).



(a) Structural geometry

(b) FEM mesh and von Mises

Figure 6.18: The damage in the L-shape plate includes three curved cracks. A set of fine unstructured mesh is used to generate the reference measurements by regular FEM, while structured mesh is employed in the XFEM forward analysis.

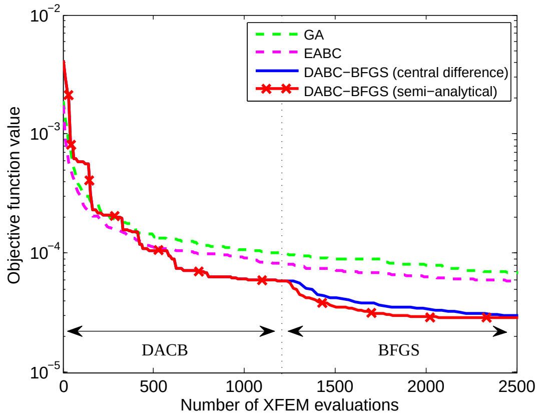


Figure 6.19: XFEM-DABC-BFGS convergence for identification of three curved cracks. Note that The gradients in BFGS here are computed using the semi-analytical form.

Figure 6.19 depicts the evolutionary convergence lines of different algorithms. It can be seen that the semi-analytical XFEM-DABC-BFGS algorithm gives the best performance. Figure 6.20 presents a number of typical snapshots of the XFEM-DABC-BFGS evolutionary process for identification of three curved cracks. It is also clear from the figure that the cracks can be detected quite well using void approximations. We conclude that in both Examples 6.7.3 and 6.7.4 that the proposed XFEM-DABC-BFGS algorithm using semi-analytical gradients is robust and efficient under uncertainties caused by modeling error.

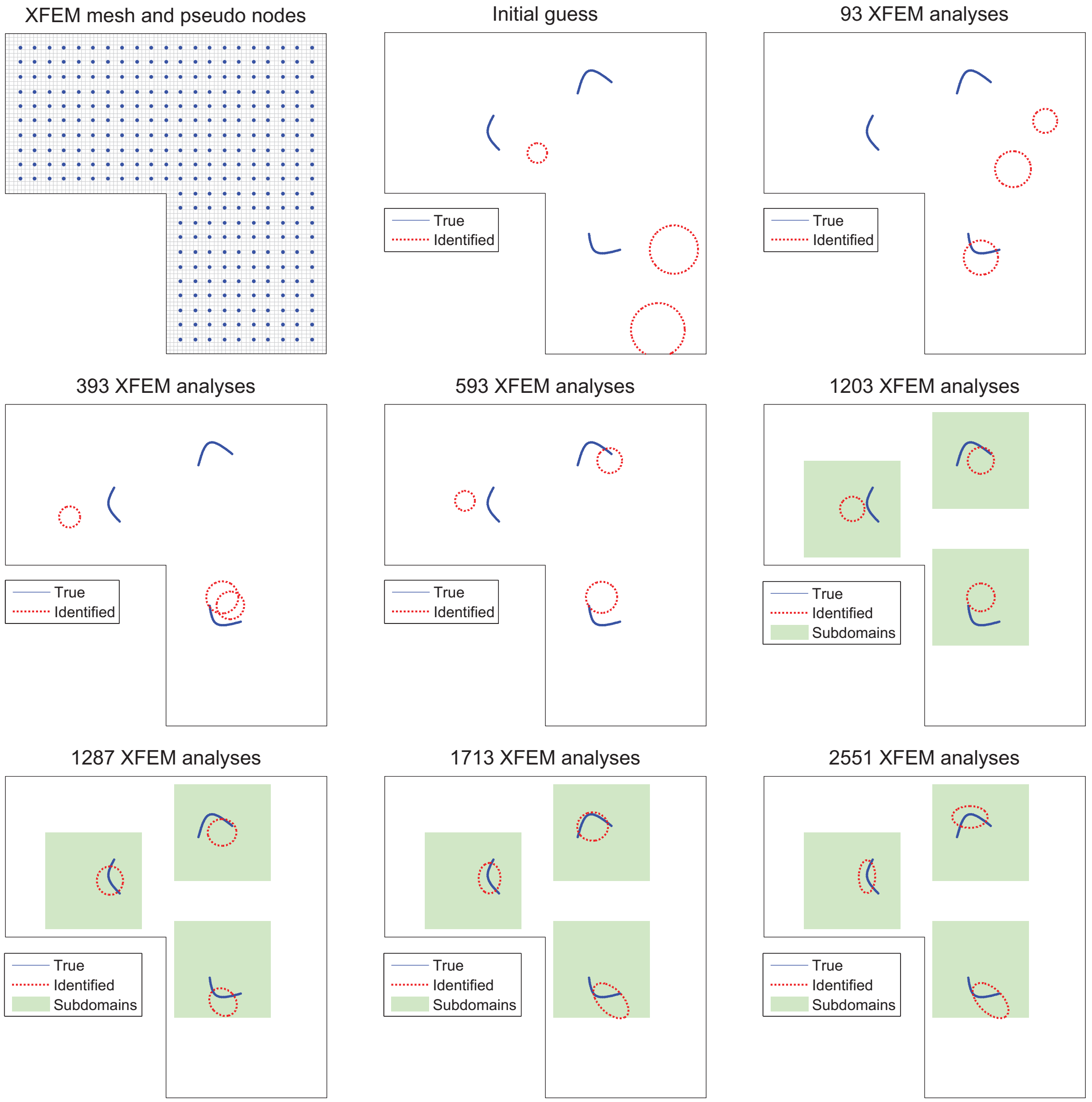


Figure 6.20: Snapshots of the XFEM-DABC-BFGS evolutionary identification process for the target of three curved cracks. The gradients in BFGS here are computed using the semi-analytical form. Note that the subdomains are determined based on the result of the 1203rd XFEM analysis. BFGS is then implemented taking such a result as an initial guess. Elliptical voids are used in BFGS to approximate the damage regions.

6.8 Conclusions

This chapter presents a novel multiscale algorithm based on XFEM for detection of multiple flaws and/or damage regions in structures using hierarchical optimizers (e.g., the DABC algorithm and the BFGS method). The proposed algorithm can be applied to quantify any number and type of flaw with arbitrary shape and size (e.g., cracks, voids, and their combination) without knowing the number of flaws beforehand.

The present work further improves the previous work on XFEM−HABC algorithm for multi-void flaws detection (see Chapter 5). The key idea of the proposed algorithm is to apply a two step optimization scheme, where first rough flaw locations and sizes are quickly determined and then fine tuning is applied in smaller subdomains to obtain global convergence to the true flaws. An XFEM model with both circular and elliptical enrichments is used to solve the forward problem. In general, the circular enrichments are used in the coarse scale search phase while the elliptical enrichments are employed in the fine scale search.

The first phase employs a discrete type optimization in which the optimizer is limited to specific flaw locations and shapes, thus converting a continuous optimization problem in the entire domain into a coarse discrete optimization problem with limited number of choices. To this end we develop a special algorithm called discrete ABC algorithm. The second phase employs a gradient-based optimization of the BFGS type on local well defined and bounded subdomains. A semi-analytical approach is developed to compute the stiffness derivative associated with the evaluation of objective function gradients.

Four benchmark examples are studied considering effects of modeling error and measurement noise and the algorithm is compared with a single step continuous optimization schemes. It is evident that the proposed methodology is robust, efficient and yields quite accurate flaw detection results under these uncertainties. Moreover, the multiscale approach converges much faster than the single step methods proposed in the literature. The simulation results demonstrate the potential application of the proposed algorithm in the field of nondestructive structural damage detection.

Part III

Conclusions

Chapter 7

Conclusions

The concluding chapter presents the main contributions and some concluding remarks of this dissertation. Directions of future research are also introduced.

7.1 Main contributions and concluding remarks

This dissertation investigated state-of-the-art hierarchical optimization-based computational algorithms for multiscale model-based system identification and damage detection. The model is typically characterized by a set of unknown parameters governing the actual system's behavior. These parameters can be identified by employing optimization tools such as the improved artificial bee colony algorithms and local search operators like the simplex method and gradient-based optimization techniques. The dissertation is presented in two parts depending on the scale of the system. The first three chapters (2–4) concentrates on identification and damage detection of macro-scale systems like buildings and bridges through time domain input-output and output-only measurements. The second two chapters (5–6) aim at multiple flaws/defects detection of meso-scale systems such as critical structural components like plates.

The author's main contributions in the field of damage detection of multiple length scale systems presented in this dissertation are summarized here:

- **Improved ABC algorithms**: Several improved versions of the artificial bee colony (ABC) algorithm in both serial and parallel schemes have been proposed in the context of parameter identification. Local search operators such as the simplex method and BFGS-type method were added to improve the algorithm local search capability. The performances of the proposed algorithms were illustrated by either numerical or experimental data settings under different conditions, addressing issues such as limited instrumentation, noise in measured signals, modeling errors, *a priori* knowledge of the system, etc. Comparisons with other existing identification approaches have been carried out for demonstration of the superior performance of the proposed methodology. Robustness and effective applicability to identification problems have been observed for the proposed family of improved ABC algorithms.

- **Output-only identification**: A framework for output-only identification of building and bridge systems has been proposed. This can be done by inserting a modified Newmark integration method into the optimization-based identification process, which is used for predicting the time history of input excitations. If the time histories of the structural response (e.g., accelerations) and information about the structural mass are available, then the proposed method is able to simultaneously identify the structural parameters and input forces.

- **Bayesian inference**: An optimization-based Bayesian inference methodology has been presented for probabilistic finite element model updating of macro-scale structural systems. The model updating process is first formulated as an inverse problem analyzed by Bayesian inference. Then a hybrid optimization approach is presented to solve such a high-dimensional inverse problem. System parameters as well as the prediction error covariance are updated iteratively in the optimization process. Posterior PDFs of uncertain system parameters can be determined using a weighted sum of Gaussian distributions. The effectiveness of the proposed approach was illustrated by both numerical and experimental data sets of building-type systems.

- **XFEM for multi-flaw detection using static data**: A novel algorithm based on the eXtended Finite Element Method (XFEM) and the improved ABC algorithms has been proposed to detect and quantify multiple flaws in structures. In this work, an adaptive algorithm that can detect multiple flaws without any knowledge on the number of flaws beforehand is proposed. While the previous XFEM-GA work only considered quantification of a single flaw [1–4], we propose an adaptive algorithm that can identify multiple flaws by introducing a topological variable into the search space which turns on and off flaws during run time. Hence the number of flaws to be detected becomes part of the optimization algorithm. Each flaw is approximated by a circular or an elliptical void. Either a continuous optimization algorithm (e.g., the improved ABC algorithms) or a two-step optimization scheme (e.g., discrete ABC algorithm *vs.* BFGS-type optimization) can be used to determine the flaw parameters. The identification is based on a limited number of static strain or displacement sensors uniformly placed in the problem domain or attached to the structure surface boundaries. The proposed methodology was tested and illustrated by several benchmark examples, showing a successful identification of various numbers and types of flaws with arbitrary shapes and sizes (e.g., cracks, voids, and their combination), without knowing the number of flaws beforehand.

7.2 Future research directions

Based on the observations of the present work, future research can be directed as follows:

- When dealing with identification of a macro-scale system such as buildings and bridges, matching the whole time history of measurements is a challenging task and its computational cost is expensive, since the optimization algorithm requires to solve the forward model many times. Even though efficient optimization tools reduce the number of forward analyses, the computational issue still exists. Possible solutions could be the synergy of the data-driven approach and the model-based approach. For instance, a data-driven approach is first employed to identify modal parameters (e.g., frequencies and mode shapes) and an optimization model-based approach is then applied to match the “measured” modal parameters. This process doesn’t necessarily require solving the whole system and only needs eigen-analyses of matrices. Therefore, one possible future research could be concentrated on proposing a framework combining data-driven and optimization model-based approaches, in which mode matching issues will be addressed.

- The present work deal with identification problems given a single type of measurement (e.g., acceleration response of a building). Nevertheless, heterogeneous sensing is widely used in nowadays SHM, e.g., to record accelerations, displacements, strains, etc. Therefore, possible future research could be focused on optimization model-based identification taking into account heterogeneous data fusion.

- Though the XFEM-based flaw detection approaches were numerically proved to be promising, a lab and filed testing is required in the future to validate the proposed methodologies. Since the current flaw detection framework only deals with homogeneous materials., future study could be concentrated on detecting multiple flaws embedded in composite materials through developing a laminate theory-based XFEM. In addition, the present work only considers 2D plane-stress problems with excitations and measurements along plane directions. Future work will extend the flaw identification to 3D models by developing an XFEM with shell- or plate- elements and using dynamic measurement of waves [122].

Part IV

Bibliography

Bibliography

[1] D. Rabinovich, D. Givoli, and S. Vigdergauz, “XFEM-based crack detection scheme using a genetic algorithm,” *International Journal for Numerical Methods in Engineering*, vol. 71, no. 9, pp. 1051–1080, 2007.

[2] D. Rabinovich, D. Givoli, and S. Vigdergauz, “Crack identification by ’arrival time’ using XFEM and a genetic algorithm,” *International Journal for Numerical Methods in Engineering*, vol. 77, no. 3, pp. 337–359, 2009.

[3] H. Waisman, E. Chatzi, and A. W. Smyth, “Detection and quantification of flaws in structures by the extended finite element method and genetic algorithms,” *International Journal for Numerical Methods in Engineering*, vol. 82, no. 3, pp. 303–328, 2010.

[4] E. N. Chatzi, B. Hiriyur, H. Waisman, and A. W. Smyth, “Experimental application and enhancement of the XFEM–GA algorithm for the detection of flaws in structures,” *Computers & Structures*, vol. 89, no. 7-8, pp. 556–570, 2011.

[5] H. Sun, H. Luş, and R. Betti, “Identification of structural models using a modified artificial bee colony algorithm,” *Computers & Structures*, vol. 116, no. 0, pp. 59–74, 2013.

[6] P. Ibáñez, “Identification of dynamic parameters of linear and non-linear structural models from experimental data,” *Nuclear Engineering and Design*, vol. 25, no. 1, pp. 30–41, 1973.

[7] M. Hoshiya and E. Saito, “Structural identification by extended Kalman filter,” *Journal of Engineering Mechanics*, vol. 110, no. 12, pp. 1757–1770, 1984.

[8] C. Koh and L. See, “Identification and uncertainty estimation of structural parameters,” *Journal of Engineering Mechanics*, vol. 120, no. 6, pp. 1219–1236, 1994.

[9] H. Luş, R. Betti, and R. W. Longman, “Identification of linear structural systems using earthquake-induced vibration data,” *Earthquake Engineering & Structural Dynamics*, vol. 28, no. 11, pp. 1449–1467, 1999.

[10] J.-W. Lin, R. Betti, A. W. Smyth, and R. W. Longman, "On-line identification of non-linear hysteretic structural systems using a variable trace approach," *Earthquake Engineering & Structural Dynamics*, vol. 30, no. 9, pp. 1279–1303, 2001.

[11] A. W. Smyth, S. F. Masri, E. B. Kosmatopoulos, A. G. Chassiakos, and T. K. Caughey, "Development of adaptive modeling techniques for non-linear hysteretic systems," *International Journal of Non-Linear Mechanics*, vol. 37, no. 8, pp. 1435–1451, 2002.

[12] Özge Arslan, M. Aykan, and H. N. Özgüven, "Parametric identification of structural nonlinearities from measured frequency response data," *Mechanical Systems and Signal Processing*, vol. 25, no. 4, pp. 1112–1125, 2011.

[13] G. H. McVerry, "Structural identification in the frequency domain from earthquake records," *Earthquake Engineering & Structural Dynamics*, vol. 8, no. 2, pp. 161–180, 1980.

[14] R. Brincker, L. Zhang, and P. Andersen, "Modal identification of output-only systems using frequency domain decomposition," *Smart Materials and Structures*, vol. 10, no. 3, pp. 441–445, 2001.

[15] U. Lee and J. Shin, "A frequency-domain method of structural damage identification formulated from the dynamic stiffness equation of motion," *Journal of Sound and Vibration*, vol. 257, no. 4, pp. 615–634, 2002.

[16] C. R. Farrar and S. W. Doebling, "Lessons learned from applications of vibration-based damage identification methods to a large bridge structure," in *Proceedings of the International Workshop on Structural Health Monitoring*, (Stanford University, Stanford, CA), pp. 351–370, 1997.

[17] M. J. Perry and C. G. Koh, "Output-only structural identification in time domain: numerical and experimental studies," *Earthquake Engineering & Structural Dynamics*, vol. 37, no. 4, pp. 517–533, 2008.

[18] J. Yang and S. Lin, "Identification of parametric variations of structures based on least squares estimation and adaptive tracking technique," *Journal of Engineering Mechanics*, vol. 131, no. 3, pp. 290–298, 2005.

[19] J. N. Yang and H. Huang, "Sequential non-linear least-square estimation for damage identification of structures with unknown inputs and unknown outputs," *International Journal of Non-Linear Mechanics*, vol. 42, no. 5, pp. 789–801, 2007.

[20] J. N. Yang, S. Lin, H. Huang, and L. Zhou, "An adaptive extended Kalman filter for structural damage identification," *Structural Control and Health Monitoring*, vol. 13, no. 4, pp. 849–867, 2006.

[21] M. Wu and A. W. Smyth, "Application of the unscented Kalman filter for real-time nonlinear structural system identification," *Structural Control and Health Monitoring*, vol. 14, no. 7, pp. 971–990, 2007.

[22] E. N. Chatzi and A. W. Smyth, "The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing," *Structural Control and Health Monitoring*, vol. 16, no. 1, pp. 99–123, 2009.

[23] T. Sato and K. Qi, "Adaptive H_∞ filter: its application to structural identification," *Journal of Engineering Mechanics*, vol. 124, no. 11, pp. 1233–1240, 1998.

[24] J. Ching, J. L. Beck, and K. A. Porter, "Bayesian state and parameter estimation of uncertain dynamical systems," *Probabilistic Engineering Mechanics*, vol. 21, no. 1, pp. 81–96, 2006.

[25] C. Koh, Y. Chen, and C.-Y. Liaw, "A hybrid computational strategy for identification of structural parameters," *Computers & Structures*, vol. 81, no. 2, pp. 107–117, 2003.

[26] Z. Zhang, C. Koh, and W. Duan, "Uniformly sampled genetic algorithm with gradient search for structural identification – part I: global search," *Computers & Structures*, vol. 88, no. 15-16, pp. 949–962, 2010.

[27] Z. Zhang, C. Koh, and W. Duan, "Uniformly sampled genetic algorithm with gradient search for structural identification – part II: local search," *Computers & Structures*, vol. 88, no. 19-20, pp. 1149–1161, 2010.

[28] A. Charalampakis and C. Dimou, "Identification of Bouc-Wen hysteretic systems using particle swarm optimization," *Computers & Structures*, vol. 88, no. 21-22, pp. 1197–1205, 2010.

[29] S. Xue, H. Tang, and J. Zhou, "Identification of structural systems using particle swarm optimization," *Journal of Asian Architecture and Building Engineering*, vol. 8, no. 2, pp. 517–524, 2009.

[30] C.-B. Yun and E. Y. Bahng, "Substructural identification using neural networks," *Computers & Structures*, vol. 77, no. 1, pp. 41–52, 2000.

[31] G. Franco, R. Betti, and H. Luş, "Identification of structural systems using an evolutionary strategy," *Journal of Engineering Mechanics*, vol. 130, no. 10, pp. 1125–1139, 2004.

[32] H. Tang, S. Xue, and C. Fan, "Differential evolution strategy for structural system identification," *Computers & Structures*, vol. 86, no. 21-22, pp. 2004–2012, 2008.

[33] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical Report-TR06, Erciyes University Press, Erciyes, Turkey, 2005.

[34] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.

[35] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing*, vol. 8, no. 1, pp. 687–697, 2008.

[36] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.

[37] D. Karaboga and B. Akay, "A modified artificial bee colony (ABC) algorithm for constrained optimization problems," *Applied Soft Computing*, vol. 11, no. 3, pp. 3021–3031, 2011.

[38] F. Kang, J. Li, and Q. Xu, "Structural inverse analysis by hybrid simplex artificial bee colony algorithms," *Computers & Structures*, vol. 87, no. 13-14, pp. 861–870, 2009.

[39] M. Sonmez, "Artificial bee colony algorithm for optimization of truss structures," *Applied Soft Computing*, vol. 11, no. 2, pp. 2406–2418, 2011.

[40] G. Stavroulakis and H. Antes, "Flaw identification in elastomechanics: BEM simulation with local and genetic optimization," *Structural optimization*, vol. 16, no. 2-3, pp. 162–175, 1998.

[41] M. Perry, C. Koh, and Y. Choo, "Modified genetic algorithm strategy for structural identification," *Computers & Structures*, vol. 84, no. 8-9, pp. 529–540, 2006.

[42] M. Phan, L. G. Horta, R. W. Longman, and J. N. Juang, "Improvement of Observer/Kalman Filter Identification (OKID) by residual whitening," *Journal of Vibration and Acoustics*, vol. 117, no. 2, pp. 232–239, 1995.

[43] R. Hinterding, "Gaussian mutation and self-adaption for numeric genetic algorithms," in *Evolutionary Computation, 1995., IEEE International Conference on*, vol. 1, pp. 384–389, 1995.

[44] H. Luş, M. De Angelis, R. Betti, and R. Longman, "Constructing second-order models of mechanical systems from identified state space realizations. part I: Theoretical discussions," *Journal of Engineering Mechanics*, vol. 129, no. 5, pp. 477–488, 2003.

[45] H. Luş, M. De Angelis, R. Betti, and R. Longman, "Constructing second-order models of mechanical systems from identified state space realizations. part II: Numerical investigations," *Journal of Engineering Mechanics*, vol. 129, no. 5, pp. 489–501, 2003.

[46] Y.-K. Wen, "Method for random vibration of hysteretic systems," *Journal of the Engineering Mechanics Division*, vol. 102, no. 2, pp. 249–263, 1976.

[47] H. Sun and R. Betti, "Simultaneous identification of structural parameters and dynamic input with incomplete output-only measurements," *Structural Control and Health Monitoring*, vol. 21, no. 6, pp. 868–889, 2014.

[48] E. Turco, "A strategy to identify exciting forces acting on structures," *International Journal for Numerical Methods in Engineering*, vol. 64, no. 11, pp. 1483–1508, 2005.

[49] Z. Lu and S. Law, "Force identification based on sensitivity in time domain," *Journal of Engineering Mechanics*, vol. 132, no. 10, pp. 1050–1056, 2006.

[50] D. Wang and A. Haldar, "Element-level system identification with unknown input," *Journal of Engineering Mechanics*, vol. 120, no. 1, pp. 159–176, 1994.

[51] D. Wang and A. Haldar, "System identification with limited observations and without input," *Journal of Engineering Mechanics*, vol. 123, no. 5, pp. 504–511, 1997.

[52] X. Ling and A. Haldar, "Element level system identification with unknown input with Rayleigh damping," *Journal of Engineering Mechanics*, vol. 130, no. 8, pp. 877–885, 2004.

[53] H. Katkhuda, R. Martinez, and A. Haldar, "Health assessment at local level with unknown input excitation," *Journal of Structural Engineering*, vol. 131, no. 6, pp. 956–965, 2005.

[54] J. Chen and J. Li, "Simultaneous identification of structural parameters and input time history from output-only measurements," *Computational Mechanics*, vol. 33, no. 5, pp. 365–374, 2004.

[55] B. Xu, J. He, R. Rovekamp, and S. J. Dyke, "Structural parameters and dynamic loading identification from incomplete measurements: approach and validation," *Mechanical Systems and Signal Processing*, vol. 28, no. 0, pp. 244–257, 2012.

[56] S. Law and D. Yong, "Substructure methods for structural condition assessment," *Journal of Sound and Vibration*, vol. 330, no. 15, pp. 3606–3619, 2011.

[57] T. N. Trinh and C. G. Koh, "An improved substructural identification strategy for large structural systems," *Structural Control and Health Monitoring*, vol. 19, no. 8, pp. 686–700, 2012.

[58] G. Franco, R. Betti, and R. W. Longman, “On the uniqueness of solutions for the identification of linear structural systems,” *Journal of Applied Mechanics*, vol. 73, no. 1, pp. 153–162, 2010.

[59] J. A. Nelder and R. Mead, “A simplex method for function minimization,” *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[60] J. Lagarias, J. Reeds, M. Wright, and P. Wright, “Convergence properties of the nelder–mead simplex method in low dimensions,” *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[61] A. Charalampakis and V. Koumousis, “Identification of bouc–wen hysteretic systems by a hybrid evolutionary algorithm,” *Journal of Sound and Vibration*, vol. 314, no. 3-5, pp. 571–585, 2008.

[62] M. J. Perry, *Modified genetic algorithm approach to system identification with structural and offshore application.* PhD thesis, National University of Singapore, Singapore, 2007.

[63] Z. Wu, N. E. Huang, S. R. Long, and C.-K. Peng, “On the trend, detrending, and variability of nonlinear and nonstationary time series,” *Proceedings of the National Academy of Sciences*, vol. 104, no. 38, pp. 14889–14894, 2007.

[64] H. Sun and R. Betti, “A hybrid optimization algorithm with Bayesian inference for probabilistic model updating,” *Computer-Aided Civil and Infrastructure Engineering*, 2014. [under review].

[65] M. I. Friswell and J. E. Mottershead, *Finite Element Model Updating in Structural Dynamics.* Dordrecht, The Netherlands: Kluwer Academic, 1995.

[66] J. Beck and L. Katafygiotis, “Updating models and their uncertainties. I: Bayesian statistical framework,” *Journal of Engineering Mechanics*, vol. 124, no. 4, pp. 455–461, 1998.

[67] R. Levin and N. Lieven, “Dynamic finite element model updating using neural networks,” *Journal of Sound and Vibration*, vol. 210, no. 5, pp. 593–607, 1998.

[68] B. N. Datta, “Finite-element model updating, eigenstructure assignment and eigenvalue embedding techniques for vibrating systems,” *Mechanical Systems and Signal Processing*, vol. 16, no. 1, pp. 83–96, 2002.

[69] J. Ching, M. Muto, and J. L. Beck, “Structural model updating and health monitoring with incomplete modal data using gibbs sampler,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 21, no. 4, pp. 242–257, 2006.

[70] S. Cheung and J. Beck, “Bayesian model updating using hybrid Monte Carlo simulation with application to structural dynamic models with many uncertain parameters,” *Journal of Engineering Mechanics*, vol. 135, no. 4, pp. 243–255, 2009.

[71] S. H. Cheung and J. L. Beck, “Calculation of posterior probabilities for bayesian model class assessment and averaging from posterior samples based on dynamic system data,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 25, no. 5, pp. 304–321, 2010.

[72] R. Jafarkhani and S. F. Masri, “Finite element model updating using evolutionary strategy for damage detection,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 3, pp. 207–224, 2011.

[73] K.-V. Yuen, *Bayesian methods for structural dynamics and civil engineering.* Singapore: John Wiley & Sons (Asia) Pte Ltd, 2010.

[74] J. Beck and S. Au, “Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation,” *Journal of Engineering Mechanics*, vol. 128, no. 4, pp. 380–391, 2002.

[75] S. Mariani and A. Ghisi, “Unscented Kalman filtering for nonlinear structural dynamics,” *Nonlinear Dynamics*, vol. 49, no. 1-2, pp. 131–150, 2007.

[76] J. Ching and Y. Chen, “Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging,” *Journal of Engineering Mechanics*, vol. 133, no. 7, pp. 816–832, 2007.

[77] K.-V. Yuen, J. L. Beck, and S. K. Au, “Structural damage detection and assessment by adaptive Markov chain,”

[78] R. Zhang and S. Mahadevan, “Model uncertainty and bayesian updating in reliability-based inspection,” *Structural Safety*, vol. 22, no. 2, pp. 145–160, 2000.

[79] X. Jiang and S. Mahadevan, “Bayesian wavelet methodology for structural damage detection,” *Structural Control and Health Monitoring*, vol. 15, no. 7, pp. 974–991, 2008.

[80] S. Sankararaman and S. Mahadevan, “Bayesian methodology for diagnosis uncertainty quantification and health monitoring,” *Structural Control and Health Monitoring*, vol. 20, no. 1, pp. 88–106, 2013.

[81] E. Jaynes, *Probability Theory: The Logic of Science.* London: Cambridge University Press, 2003.

[82] J. L. Beck, “Bayesian system identification based on probability logic,” *Structural Control and Health Monitoring*, vol. 17, no. 7, pp. 825–847, 2010.

[83] B. Jin and J. Zou, "Hierarchical bayesian inference for ill-posed problems via variational method," *Journal of Computational Physics*, vol. 229, no. 19, pp. 7317–7343, 2010.

[84] T. Erik Rabben, H. Tjelmeland, and B. Ursin, "Non-linear bayesian joint inversion of seismic reflection coefficients," *Geophysical Journal International*, vol. 173, no. 1, pp. 265–280, 2008.

[85] S. Lu, S. V. Pereverzev, and R. Ramlau, "An analysis of tikhonov regularization for nonlinear ill-posed problems under a general smoothness assumption," *Inverse Problems*, vol. 23, no. 1, p. 217, 2007.

[86] K.-V. Yuen, J. L. Beck, and L. S. Katafygiotis, "Probabilistic approach for modal identification using non-stationary noisy response measurements only," *Earthquake Engineering & Structural Dynamics*, vol. 31, no. 4, pp. 1007–1023, 2002.

[87] C.-T. Ng, "Bayesian model updating approach for experimental identification of damage in beams using guided waves," *Structural Health Monitoring*, vol. 13, no. 4, pp. 359–373, 2014 [in press].

[88] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, 2010.

[89] C. G. Broyden, "The convergence of a class of double-rank minimization algorithms 1. general considerations," *IMA J. Appl. Math.*, vol. 6, no. 1, pp. 76–90, 1970.

[90] R. Fletcher, "A new approach to variable metric algorithms," *Comput. J.*, vol. 13, no. 3, pp. 317–322, 1970.

[91] D. Goldfarb, "A family of variable-metric methods derived by variational means," *Math. Comput.*, vol. 24, no. 109, pp. 23–26, 1970.

[92] D. F. Shanno, "Conditioning of quasi-newton methods for function minimization," *Math. Comput.*, vol. 24, no. 111, pp. 647–656, 1970.

[93] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Philadelphia: SIAM Publications, 1996.

[94] E. Johnson, H. Lam, L. Katafygiotis, and J. Beck, "Phase I IASC-ASCE structural health monitoring benchmark problem using simulated data," *Journal of Engineering Mechanics*, vol. 130, no. 1, pp. 3–15, 2004.

[95] P. G. F. J. F. C. Figueiredo, E. and K. Worden, "Structural health monitoring algorithm comparisons using standard data sets," Technical Report, Los Alamos National Laboratory, Los Alamos, USA, 2009.

[96] H. Sun, H. Waisman, and R. Betti, "Nondestructive identification of multiple flaws using XFEM and a topologically adapting artificial bee colony algorithm," *International Journal for Numerical Methods in Engineering*, vol. 95, no. 10, pp. 871–900, 2013.

[97] N. Nishimura and S. Kobayashi, "A boundary integral equation method for an inverse problem related to crack detection," *International Journal for Numerical Methods in Engineering*, vol. 32, no. 7, pp. 1371–1387, 1991.

[98] G. Liu and S. Chen, "Flaw detection in sandwich plates based on time-harmonic response using genetic algorithm," *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 42, pp. 5505–5514, 2001.

[99] C. J. Hellier, *Handbook of Nondestructive Evaluation.* New York: McGraw-Hill, 2001.

[100] L. Marin, L. Elliott, D. B. Ingham, and D. Lesnic, "Identification of material properties and cavities in two-dimensional linear elasticity," *Computational Mechanics*, vol. 31, no. 3-4, pp. 293–300, 2003.

[101] P. Cloetens, M. Pateyron-Salome, J.-Y. Buffiere, G. Peix, J. Baruchel, F. Peyrin, and M. Schlenker, "Observation of microstructure and damage in materials by phase sensitive radiography and tomography," *Journal of Applied Physics*, vol. 81, pp. 5878–5886, May 1997.

[102] J. E. Michaels, "Detection, localization and characterization of damage in plates with an in situ array of spatially distributed ultrasonic sensors," *Smart Materials and Structures*, vol. 17, no. 3, p. 035035, 2008.

[103] V. Giurgiutiu and A. Zagrai, "Damage detection in thin plates and aerospace structures with the electro-mechanical impedance method," *Structural Health Monitoring*, vol. 4, no. 2, pp. 99–118, 2005.

[104] Z. Su, L. Ye, and Y. Lu, "Guided lamb waves for identification of damage in composite structures: A review," *Journal of Sound and Vibration*, vol. 295, no. 3-5, pp. 753–780, 2006.

[105] T. Erhart, W. A. Wall, and E. Ramm, "Robust adaptive remeshing strategy for large deformation, transient impact simulations," *International Journal for Numerical Methods in Engineering*, vol. 65, no. 13, pp. 2139–2166, 2006.

[106] S. C. Mellings and M. H. Aliabadi, "Flaw identification using the boundary element method," *International Journal for Numerical Methods in Engineering*, vol. 38, no. 3, pp. 399–419, 1995.

[107] R. Gallego and J. Suárez, "Numerical solution of the variation boundary integral equation for inverse problems," *International Journal for Numerical Methods in Engineering*, vol. 49, no. 4, pp. 501–518, 2000.

[108] T. Burczynski, W. Kuś, A. Długosz, and P. Orantek, "Optimization and defect identification using distributed evolutionary algorithms," *Engineering Applications of Artificial Intelligence*, vol. 17, no. 4, pp. 337–344, 2004. Selected Problems of Knowledge Representation.

[109] L. Marin, "Detection of cavities in Helmholtz-type equations using the boundary element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 36-38, pp. 4006–4023, 2005.

[110] L. Comino, R. Gallego, and G. Rus, "Combining topological sensitivity and genetic algorithms for identification inverse problems in anisotropic materials," *Computational Mechanics*, vol. 41, no. 2, pp. 231–242, 2008.

[111] J. Jung, C. Jeong, and E. Taciroglu, "Identification of a scatterer embedded in elastic heterogeneous media using dynamic XFEM," *Computer Methods in Applied Mechanics and Engineering*, vol. 259, no. 0, pp. 50–63, 2013.

[112] G. Yan, H. Sun, and H. Waisman, "A guided Bayesian inference approach for detection of multiple flaws in structures using the extended finite element method," *Computers & Structures*, 2014. [under review].

[113] N. Moës, J. Dolbow, and T. Belytschko, "A finite element method for crack growth without remeshing," *International Journal for Numerical Methods in Engineering*, vol. 46, no. 1, pp. 131–150, 1999.

[114] N. Sukumar, N. Moës, B. Moran, and T. Belytschko, "Extended finite element method for three-dimensional crack modelling," *International Journal for Numerical Methods in Engineering*, vol. 48, no. 11, pp. 1549–1570, 2000.

[115] N. Sukumar, D. Chopp, N. Moës, and T. Belytschko, "Modeling holes and inclusions by level sets in the extended finite element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 46-47, pp. 6183–6200, 2001.

[116] H. Waisman and T. Belytschko, "Parametric enrichment adaptivity by the extended finite element method," *International Journal for Numerical Methods in Engineering*, vol. 73, no. 12, pp. 1671–1692, 2008.

[117] Z. L. Yang, G. R. Liu, and K. Y. Lam, "An inverse procedure for crack detection using integral strain measured by optical fibers," *Smart Materials and Structures*, vol. 11, no. 1, p. 72, 2002.

[118] H. Sun, H. Waisman, and R. Betti, "A multiscale flaw detection algorithm based on XFEM," *International Journal for Numerical Methods in Engineering*, 2014. [Accepted & in press, DOI: 10.1002/nme.4741].

[119] H. Sun, H. Waisman, and R. Betti, “A two-scale algorithm for detection of multiple flaws in structures modeled with XFEM,” in *Proc. SPIE, Nondestructive Characterization for Composite Materials, Aerospace Engineering, Civil Infrastructure, and Homeland Security*, vol. 9063, no. 22, pp. 1–14, 2014.

[120] B. Hiriyur, H. Waisman, and G. Deodatis, “Uncertainty quantification in homogenization of heterogeneous microstructures modeled by XFEM,” *International Journal for Numerical Methods in Engineering*, vol. 88, no. 3, pp. 257–278, 2011.

[121] H. Waisman, “An analytical stiffness derivative extended finite element technique for extraction of crack tip strain energy release rates,” *Engineering Fracture Mechanics*, vol. 77, no. 16, pp. 3204–3215, 2010.

[122] H. Sun, H. Waisman, and R. Betti, “Elastic wave measurement-based flaw detection in an unbounded domain using explicit XFEM and absorbing boundary layers,” *International Journal for Numerical Methods in Engineering*, 2014. [to be submitted].