AMERICAN UNIVERSITY OF BEIRUT

STOCHASTIC OPTIMIZATION OF STRUCTURAL HEALTH MONITORING TECHNIQUES

by
DANA ELIE NASR

A dissertation
submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
to the Department of Civil and Environmental Engineering
of the Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
November 2016

AMERICAN UNIVERSITY OF BEIRUT

STOCHASTIC OPTIMIZATION OF STRUCTURAL HEALTH MONITORING TECHNIQUES
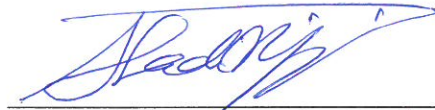
by
DANA ELIE NASR

Approved by:

_____

Dr. George Saad, Assistant Professor   Advisor
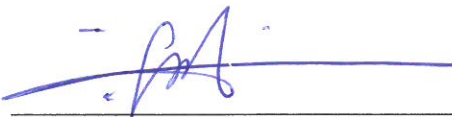Civil and Environmental Engineering

_____

Dr. Mounir Mabsout, Professor   Committee Chair
Civil and Environmental Engineering

_____

Dr. Shadi Najjar, Associate Professor   Member of Committee
Civil and Environmental Engineering

_____
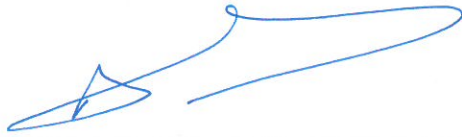
Dr. Fadi Karameh, Associate Professor   Member of Committee
Electrical and Computer Engineering

_____

Dr. Camille Issa, Professor   Member of Committee
Civil Engineering, Lebanese American University

Dr. Antoine Gergess, Professor   Member of Committee
Civil and Environmental Engineering, University of Balamand

Dr. Najib Gerges, Associate Professor   Member of Committee
Civil and Environmental Engineering, University of Balamand

Date of dissertation defense: November 29, 2016

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name:

_____
Last   First   Middle

◯ Master's Thesis   ◯ Master's Project   ◯ Doctoral
Dissertation

☐ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes
after : **One ----   year from the date of submission of my thesis, dissertation, or project.**
**   Two ----   years from the date of submission of my thesis, dissertation, or project.**
**   Three ---- years from the date of submission of my thesis, dissertation, or project.**

_____

Signature   Date

ACKNOWLEDGMENTS

I would like first to express my deepest gratitude and appreciation to my advisor, Professor George Saad for his invaluable support, guidance and encouragement through this long journey. Without his help and advice, the research would have not been achieved.

Special thanks are also extended to my thesis committee members: Professor Mounir Mabsout, Professor Shadi Najjar, Professor Fadi Karameh, Professor Najib Gergess, Professor Antoine Gergess, and Professor Camille Issa, for their time, reviews and helpful feedbacks on the thesis.

I would like to thank the civil and environmental department faculty members and staff for their assistance and cooperation. I also would like to thank my colleagues, especially Wael Slika for his assistance.

My recognition and appreciation are addressed to all who have contributed to the accomplishment of this research, namely Mr. Zakhem for funding and financially supporting this research study.

Last but not least, I owe many thanks and appreciation to my loving and supportive family. This dissertation is dedicated to my late father, Elie, who has been my main source of inspiration and motivation. Words are not enough to express my gratitude to my mother, Marie-Rose, for her prayers, unconditional love and support, and to my sister and best friend, Zeina, for always being there for me. Without them, this thesis would have never been accomplished.

AN ABSTRACT OF THE DISSERTATION OF

Dana Elie Nasr   for   Doctor of Philosophy
                       Major: Civil Engineering

Title: Stochastic Optimization of Structural Health Monitoring Techniques

The modeling of complex nonlinear structures is always accompanied with different sources of uncertainties. These uncertainties become significant when the structural system is subjected to regular aging factors or to some extreme events that could alter its behavior unexpectedly. Thus to reduce failure risks, and improve the knowledge of the system state and parameters, Structural Health Monitoring (SHM) techniques are employed for early detecting the damage of such structural systems. This process relies on analyzing collected real time measurements using Data Assimilation techniques. The Kalman Filter (KF) method and its different variations fall in the class of Sequential Data Assimilation techniques; these techniques start by calibrating the model parameters and then update them in response to any change in the material's behavior, based on an optimal probabilistic framework that minimizes the mismatch between the predicted values and actual measurements. In this dissertation, a highlight on the importance of uncertainty quantification is presented through a comparative study between intrusive and non-intrusive ways in quantifying uncertainties. A comparison between two different variations of the Kalman filter technique, the Ensemble Kalman Filter (EnKF) and the Polynomial Chaos based Ensemble Kalman Filter (PCKF), is performed for this purpose. The comparison is based on the ability and efficiency of each technique in quantifying the present uncertainties and in properly identifying the state and parameters of the system under consideration.

A four-degrees of freedom (DOFs) system subjected to El-Centro earthquake ground excitation is used to compare the EnKF to the PCKF in representing the uncertainties for SHM purposes. A preset damage of the first degree-of-freedom of the system is imposed. The Bouc-Wen model is used for the forecast and analysis steps of both KF variations as well as for synthetically generating the measurements of displacements and velocities at each DOF for parametric calibration. The comparison of the EnKF and the PCKF techniques is based on the accuracy of the results when compared to the actual data, the computational burden, and the ability of each technique in representing the uncertainty of the system for structural health monitoring purposes.

SHM used to be an impractical and time consuming technique when it was based on visual inspections, but as technology advances, the deployment of sensors and

monitoring devices in structures made it easier and more abundant. One may assume that the ideal scenario for placing these sensors is on every corner of the structure to collect as much observations as possible, but the problem of dealing with the humongous online data generated arises. Thus to extract the most informative data with the least time and cost, the optimal sensors configuration of the structure should be determined. This study combines the Genetic Algorithm (GA) technique with the ensemble Kalman filter method to form a robust optimal sensor placement (OSP) methodology for identifying the optimal sensor locations in a structure for the purpose of damage detection and system identification. The GA approach is first used to generate a random initial set of sensor locations, then through a minimization procedure, the best locations of the sensors are determined. In this study, the fitness function to be minimized is taken to be the difference between the actual measurement data and their respective predicted values, where these predicted values are calculated using the EnKF method through estimating and updating the state and model parameters of the system.

The proposed OSP methodology based on combining the GA with the EnKF techniques is applied to determine the best sensor locations of a fixed and pre-defined number of available sensors, of a ten-story shear building subjected to El-Centro earthquake excitation at its base. Synthetic measurements of displacements and velocities of the different floors are generated and their respective estimated values are calculated using the EnKF, to evaluate the required objective function to be minimized.

CONTENTS

Chapter

ILLUSTRATIONS

Figure

TABLES

DEDICATION

TO MY BELOVED FAMILY

CHAPTER 1

INTRODUCTION AND RESEARCH OBJECTIVES

1.1 Introduction

Structural Heath Monitoring is a multi-disciplinary field used for damage identification of structures based on periodically spaced real time measurements. Traditionally, this process used to be very impractical and time consuming as it was based on visual inspections, but with the recent development of numerous monitoring devices, it became more practical and easily implementable.

Although SHM is nowadays a fast and accurate method for the detection of damage in different structures, it still faces some major challenges mainly exhibited when applied to structural systems with significant physical complexities. Such systems exhibit strong nonlinear dynamical behavior with uncertain and complex governing laws. Therefore, the challenge lies in developing a robust system identification technique that can be used for characterizing the mathematical model of the structure to enhance SHM for damage detection.

Moreover, with the recent developments in monitoring technologies such as high performance sensors, optical or wireless networks, and the global positioning system, SHM measurement data became very abundant which leads to the problem of dealing with the large flow of data. Data assimilation (DA) techniques are commonly adopted to characterize the state and parameters of unknown systems using observed measurements. The data assimilation techniques are classified into two main categories: variational data assimilation and sequential data assimilation. While variational data

assimilation techniques solve optimization problems through the minimization of an objective function representing the difference between predicted and exact data, sequential data assimilation techniques, including the Bayesian probabilistic framework and the Kalman filtering (KF) techniques, propagate the system state forward in time to approximate the unknown model parameters. The methods falling in the second category are more used in the literature for damage detection and system identification purposes, than the methods falling in the first class of data assimilation techniques, because of their accuracy and computational efficacy.

To avoid having an unnecessary humongous flow of measurements and observations and to extract the most informative measurement data with the least cost and time, the sensors used for SHM purposes should be placed on their optimal locations. There are many methods used in the literature to determine these optimal sensor locations. These methods can be classified into two main categories: optimization-based methods, mainly including the Genetic Algorithm, the Particle Swarm Optimization (PSO), the Tabu Search (TS) and the Simulated Annealing (SA) methods, and selection-based methods primarily relying on information theory measures.

In this PhD thesis, a review of the existing SHM mathematical models and a comparative study between two variations of the Kalman filter technique, the ensemble Kalman filter and the polynomial chaos Kalman filter, are presented. The comparison of these KF approaches is based on the ability of each method in quantifying the different sources of uncertainties present, on the accuracy of the results when compared to the actual data, and on the computational burden. In addition, a robust methodology is proposed in this work based on combining the ensemble Kalman filter method and the

2

genetic algorithm technique for optimal sensors placement identification. The ensemble Kalman filter is used as a history matching method to predict and update the state and parameters of the system, whereas the Genetic Algorithm approach is applied to determine the optimal sensor locations in the structure through the minimization of the mismatch between the predicted and measured values.

1.2 Research Objectives

There are two main objectives for this proposed dissertation. The first objective is to perform a comparative study between intrusive and non-intrusive ways in quantifying uncertainties through two different variations of the KF technique, the ensemble Kalman filter and the polynomial chaos Kalman filter. The comparison is chosen to be between these two specific techniques because in this study the system under consideration is a complex highly nonlinear system. Consequently, the standard KF cannot be applied since it is only used for the case of linear systems perturbed by Gaussian white noise. Similarly, the extended Kalman filter is not applied in this work because of its high computational expediency and low accuracy especially when dealing with high nonlinear systems and significant non-Gaussian noise (Ghanem & Ferro, 2006). Likewise, the unscented Kalman filter was shown to require higher computational burden than other KF extensions, in many works in the literature, although it provides good enough estimates of the unknown system state and parameters (St-Pierre & Gingras, 2004; Chowdhary & Jategaonkar, 2010). The EnKF and the PCKF techniques played the role of good estimators in previous works dealing with complex nonlinear systems subjected to many sources of uncertainties (Ghanem &

Ferro, 2006; Saad, Ghanem, & Masri, 2007). The EnKF, based on Monte Carlo

simulation, is a non-intrusive technique in quantifying uncertainties. Accordingly, the

forward problem under consideration is solved using a black-box model. For complex

problems, the EnKF unfortunately requires a high number of ensemble, and

consequently a high computational burden, to properly approximate the model state and

parameters. However, the PCKF requires lower computational cost for propagating

uncertainties in the system, but being an intrusive method arises the need to go through

the black-box. The PCKF method relies on representing the system state and parameters

by their respective polynomial chaos decompositions and propagating the uncertainty in

the system using the Galerkin projection method.

      The second objective is to devise a robust mathematical tool to optimize the

sensor locations for data management purposes and therefore get the necessary

measurements and observations needed to assess the state and parameters of the

structure with the least computational burden. This tool is formulated based on

combining the ensemble Kalman filter, which is used to estimate the system state and

parameters and update them each time observation data is available, and the Genetic

Algorithm approach that is a search technique used to solve the problem of sensor

placement optimization through a minimization procedure of the mismatch between the

calculated and exact results.

1.3. Dissertation Organization

      This dissertation is composed of six chapters. Chapter 2 gives a general

background on Structural Health Monitoring (SHM) techniques and shows how these

methods evolved over the past years. This chapter also summarizes the major obstacles faced by the Structural Health Monitoring techniques presented in the literature, mainly related to the uncertainty quantification issue and the optimal sensor placement problem. Separate literature reviews along with the different methods respectively used for uncertainty quantification and optimal sensor placement purposes are also presented in chapter 2.

In Chapter 3, the focus is mainly on the mathematical formulation details of the Kalman filter (KF) technique, used with linear systems subjected to Gaussian noise, and its different extensions, used with nonlinear systems or with systems subjected to non-Gaussian noise.

Chapter 4 presents a comparative analysis between intrusive and non-intrusive methods in quantifying uncertainty through two different extensions of the Kalman filter method, the ensemble Kalman filter (EnKF) and the polynomial chaos Kalman filter (PCKF). The system under consideration in this chapter is a complex nonlinear four-DOF system subjected to El-Centro earthquake excitation. The comparison between these two methods is based on the ability of each method in quantifying the uncertainty for system identification and SHM purposes.

Chapter 5 presents a novel methodology based on combining the ensemble Kalman filter technique with the genetic algorithm (GA) for the purpose of determining optimal sensor locations in structures. The robustness and efficiency of this proposed methodology are illustrated through a numerical example consisting of a ten-story building and tested through a sensitivity analysis on the initial positions of the sensors, where the percentages of convergence of the proposed scheme to the optimal solutions, calculated using the brute-force method, are analyzed.

Chapter 6 summarizes and concludes the research outcomes and presents some suggestions for future work and research.

CHAPTER 2

STRUCTURAL HEALTH MONITORING

2.1 Structural Health Monitoring Background

Structural Health Monitoring is a method used in many engineering fields to estimate the health conditions of structures based on periodically spaced observations. Traditionally, this process was very impractical and time consuming as it was based on visual inspections, but recently it became more abundant and easily implementable with the development of numerous monitoring devices. Many works presenting reviews of the existing SHM techniques are available in the literature. Doebling, et al., worked on a report summarizing the SHM literature and the damage detection techniques (Doebling, Farrar, Prime, & Shevitz, 1996). The authors first categorized the available SHM methods based on the type of measured data used and the techniques used to detect the damage from the observed data, then classified the literature based on the type of the structure under analysis. Sohn, et al., presented an updated review of the SHM literature produced in the previously mentioned report, covering the years 1996 till 2001 (Sohn, Farrar, Hemez, & Czarnechi, 2002). The authors suggested a statistical pattern recognition paradigm summarizing the different damage detection studies available in the literature. Chang, et al., presented a global literature review on the techniques used for SHM and damage detection purposes, focusing on the methods used for the special case of civil infrastructures (Chang, Faltau, & Liu, 2003). In this review paper, the authors also categorized the different types of sensors used for the purpose of damage detection. Farrar and Worden presented a summarized overview of the different SHM techniques used and discussed the statistical pattern recognition paradigm introduced in

7

previous works (Farrar & Worden, 2007). The major challenges associated with the SHM process are also presented in this aforementioned study. Goyal and Pabla recently presented a state-of-the-art review on the newest vibration monitoring and signal processing methods used to analyze the collected measured data for structural health monitoring and damage detection purposes (Goyal & Pabla, 2015).

Although SHM techniques became precise and fast methods used for system identification and damage detection purposes in different structures, they still encounter major challenges when used with complex nonlinear structural systems. Hence, the challenge lies in devising a robust system identification technique that can be used for characterizing the mathematical model of the structure to enhance SHM for damage detection. Many researchers have investigated this topic in a number of publications in the past years. Masri, et al., generated a procedure for state equation identification for complex nonlinear systems as a generalization of the Restoring Force Method and tested its efficiency on three single-degree-of-freedom nonlinear oscillators' examples: the Duffing oscillator, the Noisy-Duffing-Van der Pol oscillator and the Bouc-Wen hysteretic oscillator (Masri, Caffrey, Caughey, Smyth, & Chassiakos, 2004). The implementation of the proposed procedure only requires the knowledge of the applied excitation and the resulting acceleration. Kerschen, et al., presented a detailed literature review on the different methods used for the purpose of nonlinear system identification in structural dynamics (Kerschen, Worden, Vakakis, & Golinval, 2006). The paper classifies these methods into seven categories: by-passing nonlinearity linearization, time-domain methods, frequency-domain methods, modal methods, time-frequency analysis, black-box modeling, and structural model updating. The authors also presented literature reviews on the methods used for nonlinearity detection and on the

characterization of the nonlinearity, including the location, the type and the functional form of the nonlinearity present. Ghanem and Ferro proposed a method based on combining the ensemble Kalman filter method with a non-parametric modeling technique for the purpose of Structural Health Monitoring for strongly nonlinear systems (Ghanem & Ferro, 2006). The methodology is applied on a four-story shear building subjected to El Centro seismic excitation at its base. A comparison between the ensemble Kalman filter and the extended Kalman filter (ExKF) techniques is presented, resulting in an outperformance of the proposed methodology over the ExKF method in parameters and state estimations. Masri, et al., suggested a procedure based on Monte Carlo simulations and Restoring Force Method to analyze the response of uncertain nonlinear systems (Masri, Ghanem, Arrate, & Caffrey, 2009). The proposed approach is investigated on a single-degree-of-freedom system with bilinear hysteretic characteristics. Saad and Ghanem combined the polynomial chaos Kalman filter method with a non-parametric representation of the nonlinearities for state and parameters estimation and structural health monitoring purposes (Saad & Ghanem, 2011). The method relies on expressing the parameters by their respective polynomial chaos representations. It was applied on a four story shear building subjected to seismic excitation and was successfully able to locate the damage in time and space. The advantage of this proposed methodology is not only limited to the low computational cost needed, but also in providing the user with all the statistical information of the uncertain parameters and variables.

Furthermore, SHM measurement data became abundantly available due to the recent developments in the monitoring devices and optical networks, leading to the problem of dealing with the large flow of observational data. As previously mentioned,

9

the data assimilation techniques are therefore adopted to estimate the system state and parameters using these measurement data. They were first developed for weather forecasting and ocean state estimation (Daley, 1997; Kalnay, 2003), then started to be used for many other applications including the system identification and SHM fields. The data assimilation techniques are classified into two main categories: variational data assimilation and sequential data assimilation. The first class aims at minimizing a certain cost function that describes the misfit between the model and actual data to find a solution to a numerical forecast model, using gradient-based optimization and adjoint methods (Navon, Zou, Derber, & Sela, 1992; LeDimet & Talagrand, 1986). The main drawback of this class of data assimilation methods is that it is computationally expensive. Whereas the second class, Sequential Data Assimilation, is based on predicting information forward in time to estimate the state of the system using its probabilistic framework and therefore overcoming the need to derive an inverse model and saving computational burden (Evensen, 1994; Bertino, Evensen, & Wackernagel, 2003).

The sequential data assimilation techniques, consisting mainly of the Bayesian probabilistic approach and the Kalman filter technique and its different variations, are used more abundantly in the literature for damage detection and system identification purposes than the variational data assimilation methods, since they are computationally more efficient. Figure 1 below summarizes the most common sequential data assimilation techniques used in the literature for SHM and damage detection purposes.
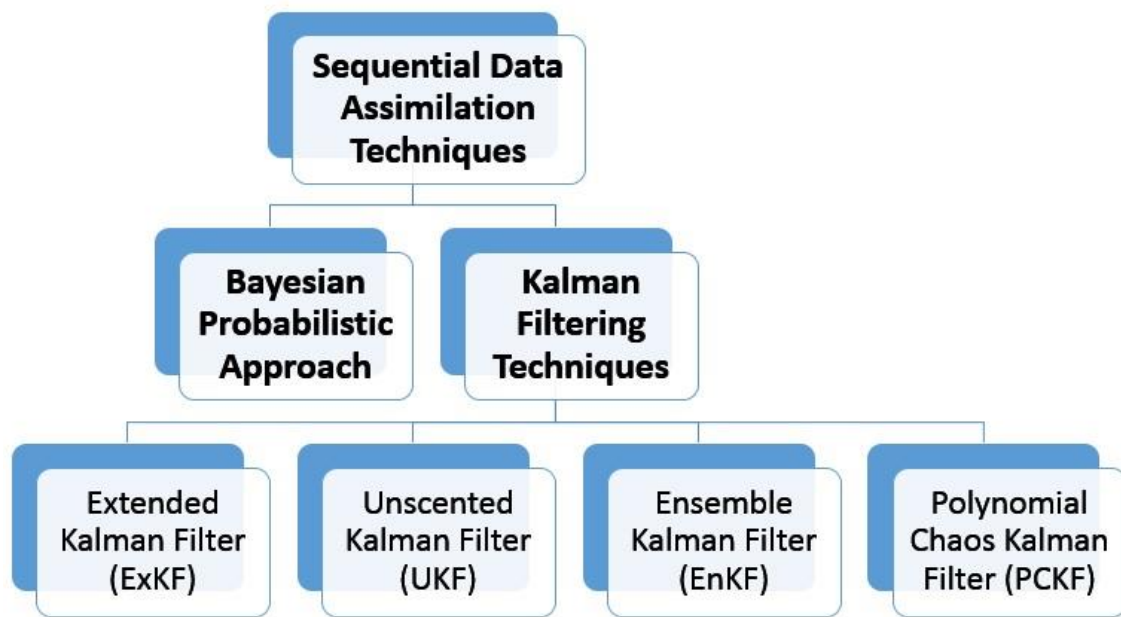
Figure 1. Sequential Data Assimilation Techniques

Vanik, et al., introduced a global continual on-line structural health monitoring method based on the Bayesian probabilistic approach to solve the problem of ill-conditioning present in the process of identifying the model parameters from the modal data (Vanik, Beck, & Au, 2000). The proposed methodology is applied on two illustrative examples, a 2-degrees of freedom (DOF) and a 10-DOF structure models. The authors proved that their approach allows the detection of small levels of damage through the observation of the structure for a long period of time. Yuen, et al., presented a structural health monitoring method consisting of two stages (Yuen, Au, & Beck, 2004). In the first stage, modal identification is performed using measured structural response from the undamaged and damaged system. Using the modal parameters identified in the first stage, updated probability density functions (PDF) for the stiffness parameters are constructed in stage II by employing a Bayesian statistical approach. The proposed methodology was applied on Phase I International Association for Structural

Control-American Society of Civil Engineers (IASC-ASCE) benchmark structure and

was able to identify the location and severity of the damage of a 4-DOF linear shear

building and a 3 dimensional (3D) 12-DOF linear shear building models. Ching and

Beck then used the previously mentioned two-step Bayesian probabilistic structural

health monitoring methodology in (Ching & Beck, 2004) and applied it on Phase II

IASC-ASCE experimental benchmark studies. The experimental Phase II benchmark

problem consists of 6 damage configurations involving the damage detection of a 3D

12-DOF shear building model, and 3 damage configurations based on testing the ability

of the proposed method in detecting the damage of a 3D 36-DOF model. The authors

also introduced a robust expectation-maximization algorithm in their work to determine

the optimal values of the parameters. Beck applied probability logic with Bayesian

updating methodology, for the purpose of system identification and uncertainty

quantification, on IASC-ASCE structural health monitoring benchmark structure model

consisting of a 4-story steel frame structure (Beck, 2010). Zhang, et al., used a recently

developed fast Bayesian method to identify the response and modal parameters, mainly

the natural frequencies, damping ratios and mode shapes, of an operational super tall

building, the Shanghai Tower, located in Shanghai, China (Zhang, Xiong, Shi, & Ou,

2016). The identified results are compared with modal parameters obtained from finite

element models. After the completion of the main structure, the mode shapes are

investigated through a field test and the variation of modal properties with changing

environment is inspected through temperature and humidity records.

The ordinary Kalman filter is a recursive Bayesian filter used for the limited

case of linear systems subjected to Gaussian errors. Extensions of the original KF to the

extended Kalman filter (ExKF) based on some linearization techniques are suggested in

the literature to handle more general cases (Chui & Chen, Kalman Filtering with Real

Time Applications, 1991). Ljung in (Ljung, 1979) suggested a convergence analysis of

the original extended Kalman filter algorithm, for parameter estimation of linear

systems, to overcome the common problem of divergent and biased estimates. Then,

Hoshiya and Saito in (Hoshiya & Saito, 1984) proposed to incorporate a weighted

global iteration procedure into the ExKF to give more stable and convergent solutions

for system identification problems of seismic structural systems. Corigliano and Mariani

used the ExKF in (Corigliano & Mariani, 2004) for state estimation and parameter

identification, and investigated the possible sources of the inaccurate performance of

this filter in case of softening through the analysis of single-degree-of-freedom and

multi-degree-of-freedom systems subjected to dynamic loadings. Yang, et al., proposed

to apply an adaptive tracking technique on the ExKF (Adaptive Extended Kalman

Filter: AEKF) so it becomes more powerful in identifying time-varying parameters

(Yang, Lin, Huang, & Zhou, 2006). This aforementioned technique was applied on

structures with unknown inputs (excitation) in (Yang, Pan, & Huang, 2007) to identify

the parameters and their variations and determine the unknown excitations. Some

experimental studies of the AEKF applied to a small-scale three-story building model

were then presented in (Zhou, Wu, & Yang, 2008) for damage identification purposes.

Ghosh, et al., proposed two new forms of the ExKF for SHM purposes in (Ghosh, Roy,

& Manohar, 2007) by performing transversal linearization. They applied these new

algorithms on single-degree-of freedom and multi-degree-of-freedom problems and

showed that they outperform the conventional ExKF. Ebrahimian, et al., recently

combined a novel nonlinear finite element model updating framework with the ExKF

method to approximate time-invariant parameters used in the finite element model of

the system under consideration (Ebrahimian, Astroza, & Conte, 2015). A direct differentiation method is employed to facilitate the work of the ExKF in estimating the parameters. The efficiency of this proposed framework in material parameters estimation is tested on two numerical examples, a cantilever steel bridge column and a three-story three-bay moment resisting steel frame. For highly nonlinear models and for models subjected to significant non-Gaussian noise, the ExKF does not provide consistent estimations of the state and model parameters of such systems. Other variations of the standard KF were suggested in the literature to overcome the major drawbacks of the ExKF, mainly the unscented Kalman filter (UKF) and the ensemble Kalman filter.

The UKF uses the "unscented transform" sampling technique to select a set of points, called sigma points, and propagates these points through the nonlinear functions, instead of linearizing the functions as in the case of the ExKF. New mean and covariance estimates are then calculated from the propagated set of points. Many works presenting a comparison between the UKF and ExKF techniques in estimating the system state and parameters are available in the literature. St-Pierre and Gingras compared these two techniques (St-Pierre & Gingras, 2004) for estimating the position module of an integrated navigation information system, where the UKF slightly outperformed the ExKF although it required higher computational burden. Kandepu, et al., applied the ExKF and the UKF techniques on four different numerical examples, a Van der Pol oscillator, a state estimation problem in an induction machine, a state estimation on a reversible reaction example and a solid oxide fuel cell (SOFC) gas turbine (GT) hybrid system (Kandepu, Foss, & Imsland, 2008). The UKF outperformed the ExKF in terms of robustness and time of convergence, while maintaining

14

comparable computational burden to the ExKF. Chowdhary and Jategaonkar compared the performance of the UKF and the ExKF methods in estimating aerodynamic parameters from aircraft flight data (Chowdhary & Jategaonkar, 2010). While the performance and accuracy of the two filters was found to be comparable, the UKF needed a greater computational cost and lower time to converge than the ExKF. Chatzi and Smyth compared the UKF to two different particle filter (PF) techniques, the generic PF and the Gaussian mixture sigma point particle filter (GMSPPF), based on the computational expediency and the efficiency in estimating the state and parameters of nonlinear complex systems (Chatzi & Smyth, 2009). The numerical problem consists of a 3-DOF system subjected to seismic excitation and including a Bouc-Wen hysteretic element, where the displacement of the first DOF and the accelerations of the second and third DOFs are assumed to be available. While the UKF was found to be the most computationally efficient method, the GMSPPF was the most accurate and robust technique in estimating the model parameters. These aforementioned authors then proposed a novel method, the mutated PF (MPF), for nonlinear, non-Gaussian online state and parameter estimations for SHM purposes (Chatzi & Smyth, 2013). The proposed algorithm outperformed the standard PF and the UKF in both computational cost and efficiency. Al-Hussein and Haldar proposed a novel concept indicated as unscented Kalman filter with unknown input and weighted global iterations (UKF-UI-WGI) to assess the health conditions of large nonlinear structural systems (Al-Hussein & Haldar, 2016). This framework combines the general steps of the traditional UKF to identify the state of nonlinear system with a substructure concept used to generate the information on the excitation and the unknown initial state vector. A weighted multiple global iterations procedure is also introduced in this proposed methodology to cover

15

large structures subjected to defective states. The outperformance of the UKF-UI-WGI method over the traditional UKF and the ExKF methods, in accurately identifying the nonlinear and large systems' defect-free and defective states, is tested on single and multiple substructures examples.

Another extension of the discrete Kalman filter, the ensemble Kalman filter (EnKF), based on Monte Carlo method was developed in the past to cover the nonlinear models subjected to non-Gaussian noises (Evensen, 1994) and to avoid the inversion of a large error covariance matrix, as in the case of the ExKF. Hommels, et al., applied the EnKF and the UKF on a conceptual nonlinear case study based on the construction of a road embankment (Hommels, Murakami, & Nishimura, 2009). The authors found that while both methods needed the same computational time, the EnKF outperformed the UKF. Ghanem and Ferro proved that the EnKF plays a role of a good estimator of the system state (Ghanem & Ferro, 2006). In their work, they combined the EnKF with a non-parametric modeling technique and applied it to a four-story shear building subjected to seismic loading, then compared it to the ExKF technique to show the benefits of their method. Evensen in (Evensen, 2009) presented a detailed literature on the applications of the EnKF as a sequential Monte Carlo (MC) method and concentrated on its use for state and parameter estimation through a number of numerical examples. Sajeeb, et al., proposed to use a conditionally linearized Monte Carlo filter for state and parameter estimation of nonlinear structural systems and showed that this method, although being computationally expensive, works as a good estimator for large systems (Sajeeb, Manohar, & Roy, 2009). Nasrellah and Manohar then combined the Monte Carlo filter with the finite element method (FEM) (Nasrellah & Manohar, 2011), which increased the range of capabilities of this method when used

16

for system identification. Slika and Saad developed a robust non-destructive SHM method for assessing the health conditions of a structure and predicting its remaining service life without corrosion using a finite element/finite difference scheme (Slika & Saad, 2016). The authors used the EnKF approach to propagate the system forward in time and calibrate the parameters using synthetically generated chloride concentration measurements. The robustness of the proposed methodology is tested on 1-D and 2-D chloride ingress numerical examples. In case of highly nonlinear systems, the EnKF requires a large number of ensemble to properly approximate the model state statistics.

To solve this problem, many methods were proposed in the literature, one method was presented by Saad and Ghanem (Saad & Ghanem, 2011; Saad, Ghanem, & Masri, 2007; Saad & Ghanem, 2009) that was based on combining the polynomial chaos with the ensemble Kalman filter to introduce the Polynomial Chaos Kalman Filter (PCKF). First, polynomial chaos decompositions of the uncertain parameters are produced. Then, the model is propagated forward in time using the Galerkin projection approach (Saad & Ghanem, 2011; Saad & Ghanem, 2009; Ghanem & Spanos, 2003; Ghanem, 1999). Finally, the uncertain parameters are updated using the polynomial chaos Kalman filter (PCKF) technique every time measurements are available. This approach was applied to a four-story shear building subjected to El-Centro excitation at its base. Li (Li & Xiu, 2009) presented a variation of the PCKF where he proposed to use a set of EnKF algorithms based on generalized polynomial chaos (gPC) expansion. The method is a two-step approach, where the author offered two different routines to solve the system of state equations in the first step (forecast step), the stochastic Galerkin approach and the stochastic Collocation approach. In addition, the gPC expansion was applied to generate arbitrarily large ensemble of realizations to find the

17

state estimates in the second step (analysis step) for both approaches. Li examined the efficiency of his approach through a number of linear and nonlinear problems. A third method was proposed by Blanchard (Blanchard, Sandu, & Sandu, 2007; Blanchard, Sandu, & Sandu, 2010; Blanchard, Sandu, & Sandu, 2010), where he solved parameter estimation problems using the Polynomial Chaos (PC) theory with direct stochastic collocation. This approach was tested on a four degree-of-freedom roll plane model of a vehicle, where the Bayesian theorem was used to obtain the maximum likelihood estimates that are then compared to the exact results. These exact states were created from a proposed experimental setting. Spiridonakos, et al., suggested a new method, based on combining the polynomial chaos expansion (PCE) method with the independent component analysis (ICA) algorithm, to monitor the health conditions of structures subjected to operational variability (Spiridonakos, Chatzi, & Sudret, 2016). The method was found to be efficient and robust when applied to two real and large-scale case studies, the Überführung Bärenbohlstrasse Bridge (Switzerland) subjected to operational conditions and an actual damaged case, the Z24 Bridge (Switzerland).

One main challenge associated with the aforementioned existing sequential data assimilation methods is the presence of various sources of uncertainties when dealing with complex and nonlinear systems. The first objective of this thesis is to show the importance of quantifying these uncertainties in data assimilation techniques for SHM purposes, for that reason a comparison between the Ensemble Kalman Filter (EnKF) and the Polynomial Chaos Kalman Filter (PCKF) is conducted based on the ability of each method in quantifying these uncertainties with the least computational burden and resulting in the most accurate results.

18

Another main challenge associated with SHM techniques is to determine the optimal locations of the monitoring devices on the structures to avoid having an enormous and unnecessary flow of measured data that requires lots of time and cost to be processed and analyzed and get only the most informative data required to estimate the system state and parameters and detect the damage in the structures. Therefore, the second objective of this dissertation is to present a novel methodology used to solve optimal sensor locations problems.

2.2. Overview of the Challenges faced by SHM

2.2.1. Uncertainty Quantification

The modeling of complex nonlinear structures is always accompanied with different forms of uncertainties due to many sources of errors. These errors are magnified due to regular aging factors and deterioration of the structure or due to extreme events, such as earthquakes, that could alter its behavior unexpectedly. The randomness facing the modeling of such complicated systems may be due to many possible sources, mainly the presence of model errors coming from the inadequacy of the mathematical model in describing the true physics of the system under consideration, parameter uncertainty and parametric variability coming respectively from the model parameters of the system and the input variables of the model that have unknown exact values, and experimental or observational uncertainty coming from the variability in the experiments' measured data (Kennedy & O'Hagan, 2001). The different sources of randomness in physical phenomena can be classified into two main categories, aleatoric and epistemic uncertainty (Helton, 2000; Helton & Davis, 2002; Helton & Oberkampf, 2004; Ghanem & Spanos, 2003; Der Kiureghiana & Ditlevsen,

2009). The first category is due to natural randomness inherent in the behavior or environment of the system. The aleatoric uncertainty is also called irreducible uncertainty since it cannot be reduced by performing more experimental testing. The second category, the epistemic uncertainty or reducible uncertainty, is mainly due to the lack of knowledge and available observational data. It can be reduced by conducting additional experimental studies or implementing a new better physical model, leading to an increased amount of knowledge about the behavior of the system under consideration.

A range of different uncertainty representation methods are available in the literature to properly address and accurately quantify this wide variety of uncertainties and sources of randomness present in such complex structural models (Lin, Engel, & Esliner, 2012). These techniques can be categorized into two large groups, statistical methods directly applied to the models, such as Monte Carlo (MC) simulation, random sampling methods, response surface methods and many others, and deterministic methods used to solve stochastic partial differential equations representing the mathematical model of the physical process, mainly the Karhunen-Loeve expansion (KLE) and the polynomial chaos expansion (PCE) (Lin, Engel, & Esliner, 2012). It should be noted that a huge number of methods used for the purpose of uncertainty representation is available in the literature, but this subsection only discusses, in what follows, the most popular techniques.

2.2.1.1. Monte Carlo Simulation

Monte Carlo method was first introduced by Stanislway Ulam in the late 1940s and then developed by John Von Neumann and Nicholas Metropolis (Metropolis &

Ulam, 1949). It is named after Monte Carlo Casino in Monaco that is famous for its gambling games that are based on random outputs and luck, as much as any modeling method.

Monte Carlo simulation is the most common technique used in the literature for quantifying uncertainties by propagating the input parameters uncertainties through the model into uncertainties in the output. The method starts by generating a large number of realizations of the system parameters based on their probability distributions. Deterministic computations on the stochastic inputs are then performed, resulting in a large number of independent possible future paths for the system. Finally, these output results, coming from different sets of stochastic values, are assembled to produce probability distributions of possible outcome values and determine the system response (Christian & Casella, 2013; Cunha, Nasser, Sampaio, Lopes, & Breitman, 2014).

Although the Monte Carlo method is general and simple to implement, which makes it the most popular method used for uncertainty quantification purposes, it faces some major challenges especially that it requires a large number of realizations to adequately describe the statistical behavior of the random system under consideration and consequently demanding a high computational effort.

2.2.1.2. Random Sampling Methods

- Simple Random Sampling (SRS): is the most widely used probability sampling method, where each member of a subset of individuals of an entire statistical population is randomly chosen and has an equal chance to be selected. In addition, each subset of a fixed number k of individuals also has an equal probability to be chosen among all other subsets of k members from the total population. Although the SRS method is

extensively used in the literature because of its ease of use, implementation and analysis, the robustness and practicality of this method are only guaranteed when dealing with small and homogenous populations (Yates, Moore, & Starnes, 2008).

• Systematic Random Sampling (Interval Sampling): systematic sampling consists of randomly selecting a first sample element from the population as a starting point of the sampling process. A fixed periodic sampling interval k is then calculated by dividing the total population size by the chosen sample size, and consequently the method continues by selecting every k^{th} member of the population (Explorable.com, 2009; Black, 2004). As in the case of SRS, systematic sampling must only be used with homogeneous populations. Although systematic sampling is easy to implement when compared to other probability sampling methods, the researcher applying this method must take into consideration any hidden pattern or any periodicity related to the sampling interval used causing threats to the randomness of the sample and resulting in an unrepresentative sample of the population.

• Stratified Sampling: in stratified sampling, the researcher divides the entire population into smaller groups, known as "strata", each containing elements sharing specific characteristics and attributes (i.e. age, gender, religion …). A simple random sampling or a systematic sampling are usually applied within each stratum and the resulting samples are then combined into an overall random sample (Explorable.com, 2009; Shahrokh Esfahani & Dougherty, 2014). Stratified sampling method results in more precise and accurate estimates of heterogeneous populations, partitioned into homogeneous subgroups, than the simple random sampling method applied on the entire population, because the variability within each stratum is much lower when compared to the variability of the whole population. Due to the precision and accuracy accompanied

22

with the stratified sampling method, the sample size needed can be reduced, resulting in cheaper sampling cost and computational burden. This is true when the stratified sampling is applied on populations characterized by a diversity of traits, but this method becomes inefficient and impractical if the strata are difficult to be formed disjointedly.

- Cluster Sampling: the researcher using cluster sampling, first divides the population into subgroups, called "clusters". A simple random sampling or a systematic random sampling method is then applied on the available clusters to select a simple random sample. Two types of cluster samples are available in the literature, the one-stage cluster sample, including all elements in each sampling cluster and the two-stage cluster sample, including only a finite number of elements from each cluster using SRS or systematic sampling. The main advantage of cluster sampling over all other sampling methods lies in the computational cost reduction for preparing sampling frames. On the other side, this method may suffer from high sampling errors and results in unrepresentative samples of the entire population since it abandons a significant part of the clusters and leave them outside the sample (Explorable.com, 2009).

- Latin Hypercube Sampling (LHS): LHS is a widely used statistical model, implemented for the purpose of generating controlled random samples of points for a probability distribution, ensuring that each distribution in the model under consideration is evenly sampled. One main advantage of this sampling method lies in the gain of accuracy and precision, especially when used with Monte Carlo method for numerical integration problems. Unfortunately, this improvement in the level of accuracy is mostly modest and comes at the cost of additional memory space required to achieve the sampling process (McKay, Beckman, & Conover, 1979; Iman, Davenport, & Zeigler, 1980).

• Importance Sampling: The main idea behind importance sampling method is to appropriately select an alternative distribution from which samples are generated to estimate the targeted distribution. This method can be used for Monte Carlo integration and can result in a reduction in the computational burden in case the researcher chooses a good importance sampling function. On the other side, this sampling technique requires longer sampling process time than the standard Monte Carlo method in case of a bad choice of the proposal distribution. Therefore, the choice of the distribution from which the samples are generated to estimate the properties of the distribution of interest is a key issue in importance sampling. In other words, sampling x from the distribution $f(x)$ is similar to sampling $x \times \omega(x)$ from the distribution $g(x)$, where $\omega(x) = \frac{f(x)}{g(x)}$ is the importance sampling weight (Geweke, 1989; Veach & Guibas, 1995; Neal, 2001).

2.2.1.3. Karhunen-Loeve Expansion

It is difficult to deal with random processes; therefore special techniques should be used to solve stochastic differential equations. One of the most practical tools is the Karhunen-Loeve (KL) expansion that is a continuous representation technique for stochastic processes using a superposition of orthonormal random variables weighted by the eigenfunctions of the covariance of these random processes (Ghanem & Spanos, 2003; Sudret & Der Kiureghian, 2000).

The KL expansion of a random field $\omega(x, \theta)$ is based on the spectral decomposition of its covariance function $C(x_1, x_2) = \sum_{n=0}^{\infty} \lambda_n f_n(x_1) f_n(x_2)$, which is bounded, symmetric and positive definite. The eigenvalue (λ_n) and the eigenvector (f_n)

24

of the covariance $C(x_1, x_2)$ are the solution to the Fredholm integral equation (eigenvalue problem) (Ghanem & Spanos, 2003):

$$\int_D C(x_1, x_2) f_n(x_2) dx_2 = \lambda_n f_n(x_1) \tag{1}$$

Having $\bar{\omega}(x)$ being the expected value of the random field $\omega(x, \theta)$ and $\{\xi_n(\theta)\}$ denoting the coordinates of the realization of the random field with respect to the set $\{f_n\}$, $\omega(x, \theta)$ can be written as:

$$\omega(x, \theta) = \bar{\omega}(x) + \sum_{n=0}^{\infty} \xi(\theta) \sqrt{\lambda_n}\, f_n(x) \tag{2}$$

Truncating the series in the previous equation at the Mth term results in the following equation:

$$\omega(x, \theta) = \bar{\omega}(x) + \sum_{n=0}^{M} \xi(\theta) \sqrt{\lambda_n}\, f_n(x) \tag{3}$$

EigenValue Problem

Taking the one-Dimensional autocorrelation function as follow:

$C(x_1, x_2) = e^{\frac{-|x_1 - x_2|}{b}}$, where b is the correlation length and the domain D is [-a, a], the eigenvalues (λ_n) and eigenfunctions $(f_n(x))$ are solutions of the following integral equation:

$$\int_{-a}^{a} e^{-c|x_1 - x_2|} f(x_2) dx_2 = \lambda f(x_1) \tag{4}$$

$$\Rightarrow \int_{-a}^{x_1} e^{-c(x_1 - x_2)} f(x_2) dx_2 + \int_{x_1}^{a} e^{-c(x_1 - x_2)} f(x_2) dx_2 = \lambda f(x_1) \tag{5}$$

Differentiating with respect to x_1:

$$\Rightarrow \lambda f'(x_1) = -c \int_{-a}^{x_1} e^{-c(x_1-x_2)} f(x_2)dx_2 + f(x_1) + c \int_{x_1}^{a} e^{-c(x_1-x_2)} f(x_2)dx_2 \tag{6}$$

$$-f(x_1)$$

Differentiating a second time with respect to x_1:

$$\Rightarrow \lambda f''(x_1) = -c^2 \int_{-a}^{x_1} e^{-c(x_1-x_2)} f(x_2)dx_2 - cf(x_1) + c^2 \int_{x_1}^{a} e^{c(x_1-x_2)} f(x_2)dx_2 \tag{7}$$

$$-cf(x_1)$$

$$\Rightarrow \lambda f''(x_1) = c^2 \lambda f(x_1) - 2cf(x_1) \tag{8}$$

$$\Rightarrow \lambda f'(x) = (-2c + c^2\lambda)f(x) \tag{9}$$

$$\Rightarrow f''(x) + \omega^2 f(x) = 0, -a \le x \le +a \tag{10}$$

The boundary conditions are the following: $\begin{cases} cf(a) + f'(a) = 0 \\ cf(-a) - f'(-a) = 0 \end{cases}$

$$\Rightarrow f(x) = a_1 cos(\omega x) + a_2 sin(\omega x) \tag{11}$$

$$\Rightarrow \begin{cases} a_1(c - \omega tan(\omega a)) + a_2(\omega + ctan(\omega a)) = 0 \\ a_1(c - \omega tan(\omega a)) - a_2(\omega + ctan(\omega a)) = 0 \end{cases} \tag{12}$$

$$\Rightarrow \begin{cases} c - \omega tan(\omega a) = 0 \\ \omega + ctan(\omega a) = 0 \end{cases} \tag{13}$$

For n odd: ($n \ge 1$)

- ω is the solution of: $\frac{1}{b} - \omega tan(\omega a) = 0$ in the range $\left[(n-1)\frac{\pi}{a}, \left(n - \frac{1}{2}\right)\frac{\pi}{a}\right]$

- $\lambda_n = \frac{2b}{1+\omega_n^2 b^2}$

- $f_n(x) = \frac{cos(\omega_n x)}{\sqrt{a + \frac{sin(2\omega_n a)}{2\omega_n}}}$

For n even: ($n \ge 2$)

- ω^* is the solution of: $\frac{1}{b} - \omega^* tan(\omega^* a) + \omega^* = 0$ in the range $\left[\left(n - \frac{1}{2}\right)\frac{\pi}{a}, n\frac{\pi}{a}\right]$

26

- $$\lambda_n^* = \frac{2b}{1+\omega_n^{*2}b^2}$$

- $$f_n^*(x) = \frac{\sin(\omega_n^* x)}{\sqrt{a - \frac{\sin(2\omega_n^* a)}{2\omega_n^*}}}$$

Numerical Solution

The Fredholm equation of the second kind: $\int_D C(x_1, x_2) f_n(x_2) dx_2 = \lambda_n f_n(x_1)$, is solved using the Galerkin method.

First, each Eigenfunction of $C(x_1, x_2)$ is written as follows: $f_k(x) = \sum_{i=1}^{N} d_i^{(k)} h_i(x)$, where $h_i(x)$ represents a complete set of functions in the Hilbert space H.

Second, since the previous summation is truncated at the Nth term, this will result in an error that can be represented as:

$$\in_N = \sum_{i=1}^{N} d_i^{(k)} \left[\int_D C(x_1, x_2) h_i(x_2) dx_2 - \lambda_n h_i(x_1) \right] \tag{14}$$

This error should be minimized $\Rightarrow \left(\in_N, h_j(x) \right) = 0; j = 1, ..., N$

$$\Rightarrow \sum_{i=1}^{N} d_i^{(k)} \left[\int_D \left[\int_D C(x_1, x_2) h_i(x_2) dx_2 \right] h_j(x_1) dx_1 - \lambda_n \int_D h_i(x) h_j(x) dx_1 \right] = 0 \tag{15}$$

The preceding procedure will result in the following generalized algebraic Eigenvalue Problem that should be solved for the eigenvalues (λ) and eigenvectors (columns of matrix D) (Ghanem & Spanos, 2003; Sudret & Der Kiureghian, 2000):

$$CD = \Lambda BD \tag{16}$$

where: $C_{ij} = \iint_{D\,D} C(x_1, x_2) h_i(x_2) dx_2 h_j(x_1) dx_1$

$$B_{ij} = \int_D h_i(x)h_j(x)dx$$

$$D_{ij} = d_i^{(j)}$$

$$\Lambda_{ij} = \delta_{ij}\lambda_i$$

2.2.1.4. Polynomial Chaos Expansion

N. Wiener (Wiener, 1938) was the first one to introduce the Polynomial chaos theory in the form of Homogeneous Chaos Expansion that uses Hermite polynomials to model stochastic processes with Gaussian random variables. Every source of uncertainty in the system under consideration is represented by a vector of random variables ξ. All these independent random variables are then correlated with an individual random event θ.

Any random process $u(x,\theta)$ with prescribed probability density function, can be expanded as a polynomial function of Multi-dimensional Hermite polynomials in Gaussian random variables (Ghanem & Spanos, 2003), as

$$u(x,\theta) = \sum_{i=0}^{\infty} u_i(x)\psi_i\big(\xi(\theta)\big) \tag{17}$$

where $\{u_i, i = 0, \dots, \infty\}$ are deterministic expansion coefficients that can be evaluated using different methods (Projection Method, Collocation Method,…), $\psi_n(\xi_{i1}, \dots, \xi_{in})$ is the nth order Polynomial Chaos in the Gaussian variables $(\xi_{i1}, \dots, \xi_{in})$ and $\{\psi_i, i = 0, \dots, \infty\}$ are the orthogonal multidimensional Hermite polynomials.

After truncating the polynomial chaos expansion at the Pth term, the above relation becomes

$$u(x, \theta) = \sum_{i=0}^{P} u_i(x) \psi_i\big(\xi(\theta)\big) \tag{18}$$

where *P+1* is the total number of terms in a polynomial chaos expansion. For an order less than or equal to *p* and a dimension equals to *M*, *P+1* is equal to

$$P + 1 = \frac{(p + M)!}{p!\, M!} \tag{19}$$

2.2.2. Optimal Sensor Placement

Another challenge faced by SHM techniques is to place the monitoring devices on their optimal locations on the structure to get the most necessary and informative data for damage detection and avoid having an unnecessary humongous flow of measurement data.

The problem of optimally placing sensors on a structure was rigorously investigated in the literature. Many optimal sensor placement (OSP) methods were suggested to determine the best sensor locations for damage detection and parameter estimation purposes. These optimal sensor placement methods can be classified into two main categories, as shown in Figure 2 below: Optimization-based and Selection-based procedures (Staszewski & Worden, 2001). The most used optimization-based methods available in the literature include the Genetic Algorithm (GA), the Particle Swarm Optimization (PSO), the Simulated Annealing (SA), the Tabu Search (TS) approaches and other machine learning algorithms. On the other hand, selection-based methods rely mainly on information theory measures.
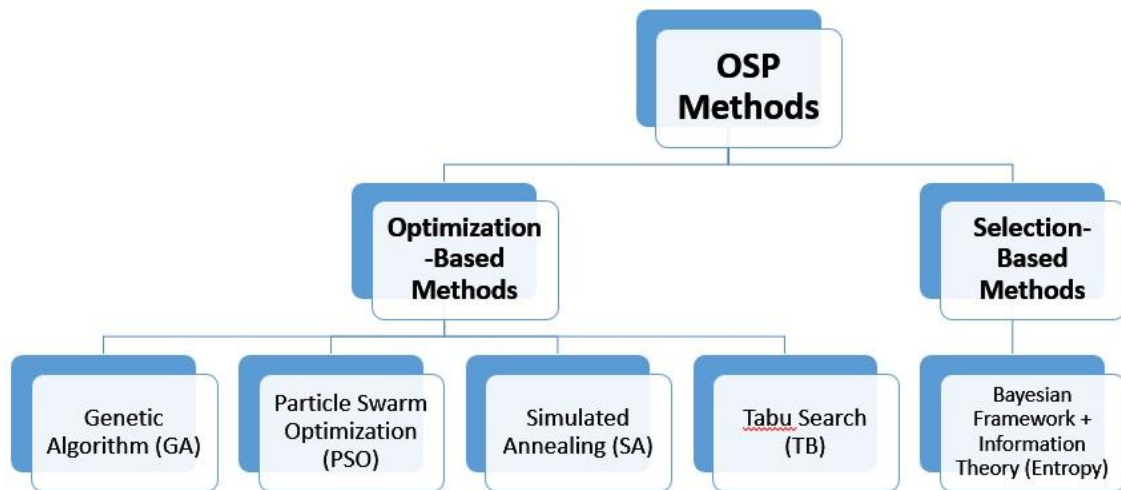
Figure 2. Optimal Sensor Placement Methods

The GA is a search technique used to solve optimization problems using the theory of natural evolution that includes some random operations such as: selection, crossover, and mutation. The algorithm starts by generating a random set of solutions, called "population of chromosomes", and by determining a certain fitness function to be optimized. The fitness of each solution is evaluated and the best ones are selected as "parents". A series of natural operations is performed on the "parents" and a new better population is created. The algorithm is repeated, each time starting with the new created population, until some predefined termination condition is satisfied. Many researchers used the GA to determine the best sensor locations for the purpose of system identification. Yao, et al., used the GA to determine the optimal sensor placement for a space structure and a photo-voltaic array, and defined the fitness function to be the determinant of the Fisher information matrix (Yao, Sethares, & Kammer, 1993). Guo, et al., proposed an improved version of the GA by improving the crossover and mutation operations and applied the new proposed methodology on a SHM system consisting of a two-dimensional (2D) truss structure to determine the optimal sensor configuration

(Guo, Zhang, Zhang, & Zhou, 2004). A recent unsupervised and nonparametric genetic algorithm for decision boundary analysis (GADBA) approach, combined with a novel concentric hypersphere (CH) algorithm, was proposed by Silva, et al., for damage detection in bridges, in the presence of environmental and operational variability (Silva, et al., 2016).

The particle swarm optimization is a stochastic optimization method sharing many similarities with the genetic algorithm technique. The algorithm starts by randomly generating a population of solutions (called 'particles'), then updates the generations to get the optimum particles. Rao and Anandakumar proposed a new hybrid PSO method based on combining a self-configurable PSO with the Nelder-Mead algorithm for the purpose of optimal sensor placement identification (Rao & Anandakumar, 2007). The main objective of the proposed algorithm is to identify the modal frequencies as well as the mode shapes. The authors used two distinct fitness functions in their work, the total Mean Square Error (MSE) and the determinant of the Fisher Information Matrix (FIM). The effectiveness of the presented hybrid PSO algorithm in determining the optimal numbers and placement of sensors is tested on three numerical examples, a cantilever beam, a slab bridge, and a girder bridge. These examples show the outperformance of the proposed algorithm over some iterative information-based approaches and other PSO implementations in generating optimal solutions as well as faster convergence. Begambre and Laier also used a combination between the basic PSO and the Nelder-Mead algorithm for damage identification purposes (Begambre & Laier, 2009). The fitness function used in this work is based on the Frequency Response Functions (FRF) of the system. According to the study results, the hybrid PSO algorithm outperformed the basic PSO and the simulated annealing

31

algorithms in locating the global optimum using a lower number of function evaluations, when applied on three different numerical problems, a 10-bar truss, a free-ends steel beam, and a nonlinear oscillator. Seyedpoor generated a two stage method for detecting damage in structures (Seyedpoor, 2012). A model strain energy based index (MSEBI), calculated using the modal analysis information extracted from a finite element modeling, is used in the first stage to locate the eventual structural damage. Then, a particle swarm optimization technique is used in the second stage of the proposed method to determine the extent of the actual damage, using the results of the first stage. The efficiency of this two stage method in identifying the multiple damage in structural systems is tested on two numerical examples, a 15-element cantilevered beam and a 31-bar planar truss.

The simulated annealing is another optimization based method; it is named after the physical process of annealing in thermodynamics and used for finding a global optimum for problems having multiple local minima and maxima. Chiu and Lin used the simulated annealing (SA) algorithm to optimize the sensor locations for small and large sensor fields (Chiu & Lin, 2004).

The fourth optimization-based method is the tabu search, which is a "meta-heuristic" optimization approach used to avoid being trapped at a local minimum or maximum in neighborhood search problems. The main step in the TS method is to create a "tabu list" of previously considered solutions to prohibit visiting them again. There are limited studies on the use of TS for OSP purposes in the literature.

In the past couple of years, many researchers resorted to machine learning algorithms for developing robust system identification techniques. While the direct objective of the studies was for damage detection only, some of the techniques

presented in them may be useful for OSP purposes. Salehi, et al., proposed a study aiming at developing global interpretation algorithms using discrete binary data for damage identification purposes (Salehi, Das, Chakrabartty, Biswas, & Burgueno, 2015). Figueiredo, et al., presented a comparative study between four machine learning algorithms: auto-associative neural network (AANN), factor analysis (FA), Mahalanobis squared distance (MSD) and singular value decomposition (SVD), where the MSD was found to be the best method for damage detection in the presence of operational and environmental variability (Figueiredo, Park, Farrar, Worden , & Figueiras, 2011). Zhou, et al., presented an approach based on combining the posterior probability support vector machines (PPSVM) based method, and Dempster-Shafer (DS) evidence theory for damage identification of structures (Zhou, et al., 2015).

One common challenge facing optimization-based methods is selecting the appropriate fitness function for minimization purposes. This is typically problem dependent, and the efficacy of the used optimization approach hinges on the suitability of the selected fitness function.

The second category of solvers used for OSP methods, selection-based procedures, is based on selecting the best sensor locations using information theory measures (i.e. entropy or mutual information) which quantify the uncertainty associated with random variables. The optimal locations of the sensors are the ones having good information content and the bad locations are where sensors have low information content. Yuen, et al., applied the Bayesian method to compute the uncertainty in model parameters and the information entropy to get the optimal sensor configuration. The proposed approach was applied on two numerical examples with uncertain excitations:

33

an 8 DOF chain mass-spring model and a 40 DOF truss model (Yuen, Katafygiotis, Papadimitriou, & Mickleborough, 2001). While these selection-based methods generally yield good results, they suffer in most of the cases from high computational burden when compared to the optimization-based methods.

Many methods based on a combination between the GA approach and the Bayesian framework are extensively used in the literature for optimally placing sensors on a structure. The Bayesian approach is first used to compute the uncertainty in the parameters, then the GA is applied to minimize the information entropy over the set of possible sensor configurations. Papadimitriou, et al., combined these two methods to determine the best sensor locations of a 9-story building and a 29 DOF truss structure (Papadimitriou, Beck, & Au, 2000). Flynn and Todd applied this same approach to determine the best locations of sensors in the field of active sensing (Flynn & Todd, 2010). Chow, et al., and Lam, et al., combined the GA and the Bayesian approaches to determine the optimal sensor configuration of a 3D finite element model of a transmission tower modeled in ANSYS (Chow H. , Lam, Yin, & Au, 2011; Lam, Yang, & Hu, 2011). Papadimitriou used the information entropy to determine the optimal sensor configuration in (Papadimitriou , 2004), where he proposed an approach, sequential sensor placement (SSP), that is used to sequentially determine the best sensor locations by placing one sensor at a time, and compared it with the GA. The first sensor is placed in a location that results in the highest reduction in the information entropy for the case of one sensor only. The location of the second sensor is next determined that gives the highest reduction in the information entropy for the case of two sensors, given that the first sensor is fixed in its optimal location found in the first step. The approach continues in a similar fashion until the optimal configuration of the fixed number of

available sensors is reached. The comparison is based on the computational burden

needed to determine the optimal sensor locations of two numerical examples, a 10 DOF

chain-like spring-mass model and a 240 DOF three-dimensional (3D) truss structure.

This dissertation presents a methodology based on combining the GA and the

EnKF methods to tackle the problem of optimal sensor placement. In the presented

framework, the GA is used to randomly generate possible sensor locations, and the

EnKF is employed to identify the system state and parameters based on the selected

sensor locations. Within the GA framework, the fitness function to be minimized is

taken to be the difference between synthetically generated actual measurement data and

their respective predicted values, calculated using the state and model parameters

estimated via EnKF. In some recent works, the GA and the EnKF methods were

combined together for purposes other than the optimal sensor placement. Li, et al.,

combined these two techniques to determine the optimal permeability distribution of a

3D model of landfills (Li, Qin, Tsotsis, & Sahimi, 2012), where the GA was first used

to generate initial ensembles of the permeability distribution of the landfill model and

the EnKF was then used to approximate this distribution and update it based on

synthetic real-time measurements. Lyons and Nasrabadi used the EnKF for history

matching and updating a reservoir model, then combined it with the GA method for

well placement optimization (Lyons & Nasrabadi, 2013).

CHAPTER 3

SEQUENTIAL DATA ASSIMILATION

The main challenge encountered with the existing sequential data assimilation methods is the presence of various sources of uncertainties and errors when dealing with complex and nonlinear systems. This dissertation presents a general overview of the existing Kalman filtering techniques and shows the importance of quantifying the uncertainties in these data assimilation techniques when used to estimate the unknown system state and parameters for damage identification and SHM purposes. The concentration of this dissertation is mainly on the ensemble Kalman filter and the polynomial chaos Kalman filter to compare intrusive versus non-intrusive techniques for uncertainty quantification in a KF setting of complex and highly nonlinear systems.

3.1. Kalman Filter

The Kalman Filter (Kalman, 1960; Burgers, Leeuwen, & Evensen, 1998; Welch & Bishop, 2006; Grewal & Andrews, 2008) is an optimal recursive data processing estimator that approximates the state of linear dynamical systems perturbed by Gaussian white noise, using observations that are subjected to Gaussian errors. This process involves two stages, the first stage is the forecast or predictive stage, where the model state at time k is propagated forward in time, and the second stage is the update or corrective stage, where the variables describing the state of the system are adjusted based on the actual measurements at time $k+1$.

In the forecast step, a dynamical linear system is considered and characterized by a model state vector x_k at time k, the predicted state is given by the following equation,

$$x_{k+1} = A_k x_k + \varepsilon_k \quad x_k, \varepsilon_k \epsilon R^n, \quad A_k \in R^{n \times n} \tag{20}$$

where n is the dimension of the model state vector, A_k is a constant matrix and ε_k is a Gaussian white noise vector due to the modeling uncertainties. The error covariance and the forecasted error covariance matrices are defined, respectively, as

$$P_k = E[(x_k^t - x_k)(x_k^t - x_k)^T] \quad P_k \in R^{n \times n} \tag{21}$$

$$P_{k+1} = A P_k A^T + Q \quad P_{k+1}, Q \in R^{n \times n} \tag{22}$$

where x_k^t is the true state vector at time instant k, and Q is the covariance matrix of the model noise ε_k .

In the update step, the predictions are corrected at each time measurements are recorded. The state model analysis, x^a, and its respective covariance matrix, P^a, are given as functions of the state model forecast, x^f, and its respective covariance matrix, P^f, as

$$x^a = x^f + KG(d - Hx^f) \tag{23}$$

$$P^a = (I - KG \times H)P^f \quad I, KG, H \in R^{n \times n} \tag{24}$$

where I is the identity matrix, $d \in R^m$ is the observation vector, H is the measurement operator connecting the true state model x^t and the measurements vector d as follows

$$d = Hx^t + \epsilon \tag{25}$$

ϵ is the measurement noise vector,

KG is the Kalman gain given by the following equation,

$$KG = P^f H^T (HP^f H^T + R)^{-1} \tag{26}$$

where R is the error covariance matrix of the measurement noise ϵ.

3.2. Extended Kalman Filter

As mentioned in the previous subsection, the standard Kalman Filter is only used for the case of linear dynamic systems subjected to Gaussian white noise. For the case of nonlinear systems (nonlinear state and observations) or for the case of systems subjected to non-Gaussian noise, many variations of the Kalman Filter can be used instead of the traditional KF approach. Such nonlinear dynamical models have the following system state and measurement equations:

$$x_{k+1} = f_k(x_k) + \varepsilon_k \quad x_k, \varepsilon_k \in R^n, \quad f_k \in R^{n \times n} \tag{27}$$

$$y_k = h_k(x_k) + \epsilon \quad x_k, \epsilon \in R^n, \quad h_k \in R^{n \times n} \tag{28}$$

where ε_k and ϵ are Gaussian white noise vectors due to the modeling and measurement uncertainties, respectively (Ribeiro, 2004).

The most widely used approximate technique is the Extended Kalman Filter (ExKF) that was first introduced by Chui and Chen (Chui & Chen, 1991) and clearly discussed in (Ghanem, Masri, Pellissetti, & Wolfe, 2005). ExKF is based on calculating an approximation of the optimal estimate using some linearization processes. A real time linear Taylor series expansion method can be first used to recursively linearize the nonlinear system dynamics under consideration, $x_{k+1} = f(x_k) + \varepsilon_k$ around the last filtered state estimate $\hat{x}_{k,k}$.

In the forecast step, the prediction step of the traditional KF method is applied to the linearized system yielding the following predicted system state and error covariance equations:

$$\hat{x}_{k+1,k} = f_k(\hat{x}_{k,k}) \tag{29}$$

$$P_{k+1,k} = F_k P_{k,k} F_k^T + Q_k \tag{30}$$

where Q is the covariance matrix of the model noise ε_k and F_k is the Jacobian matrix of $f(.)$:

$$F_k = \nabla f_k(\hat{x}_{k,k}) \tag{31}$$

In the update step, the observation matrix y_k is linearized around $\hat{x}_{k+1,k}$ and the predicted state dynamics are corrected each time measurements are available yielding:

$$\hat{x}_{k+1,k+1} = \hat{x}_{k+1,k} + KG[y_{k+1} - h_{k+1}(\hat{x}_{k+1,k})] \tag{32}$$

$$P_{k+1,k+1} = [I - KG(H_{k+1})]P_{k+1,k} \tag{33}$$

where I is the identity matrix, KG is the same Kalman gain of the traditional KF and it is given by the following equation:

$$KG = P_{k+1,k} H_{k+1}^T [H_{k+1} P_{k+1,k} H_{k+1}^T + R_{k+1}]^{-1} \tag{34}$$

H_k is the Jacobian matrix of $h(.)$, given by:

$$H_{k+1} = \nabla h(\hat{x}_{k+1,k}) \tag{35}$$

The ExKF showed some drawbacks, especially regarding its high computational cost and its failure to be a good estimator in the case of highly nonlinear systems and/or in the case of significant non-Gaussian noise. Therefore, other variations of the KF were suggested in the literature to overcome the major drawbacks of the ExKF, mainly the unscented Kalman filter and the Ensemble Kalman filter.

3.3. Unscented Kalman Filter

The Unscented Kalman filter uses the "unscented transform" sampling technique to select a set of points, called sigma points, around the mean and propagates these points through the nonlinear functions, instead of linearizing the functions as in the case of the ExKF. The number of sigma points is equal to 2M+1, with M being the dimension of the state vector x. New mean and covariance estimates are then calculated from the propagated set of points (St-Pierre & Gingras, 2004; Kamdepu, Foss , & Imsland, 2008; Chatzi & Smyth, 2009; Chowdhary & Jategaonkar, 2010; Laviola, 2003).

The columns of the sigma points matrix, corresponding to the random variable x with mean $\hat{x}$ and covariance P_{xx} , are calculated as follows (Laviola, 2003; Chatzi & Smyth, 2009)

$$\mathcal{X}_0 = \hat{x}; \; W_0 = \frac{\lambda}{M+\lambda}$$

$$\mathcal{X}_i = \hat{x} + \left(\sqrt{(M+\lambda)P_{xx}}\right)_i \; ; W_i = \frac{1}{2(M+\lambda)}; \; i = 1 \dots M \qquad (36)$$

$$\mathcal{X}_i = \hat{x} - \left(\sqrt{(M+\lambda)P_{xx}}\right)_{i-M} ; W_i = \frac{1}{2(M+\lambda)}; \; i = M+1 \dots 2M$$

where λ is a scaling parameter and W_i is the weight corresponding to the ith point. These selected sigma points are next propagated through the nonlinear process model function as follows

$$\mathcal{X}_{i(k+1,k)} = f[\mathcal{X}_{i(k,k)}] \qquad (37)$$

In the prediction step, the predicted mean and error covariance matrix are computed as follows (Laviola, 2003; Chatzi & Smyth, 2009)

$$\hat{x}_{(k+1,k)} = \sum_{i=0}^{2M} W_i \, \mathcal{X}_{i(k+1,k)} \tag{38}$$

$$P_{(k+1,k)} = \sum_{i=0}^{2M} W_i \left[\mathcal{X}_{i(k+1,k)} - \hat{x}_{(k+1,k)}\right]\left[\mathcal{X}_{i(k+1,k)} - \hat{x}_{(k+1,k)}\right]^T + Q \tag{39}$$

where Q is the process noise covariance matrix.

The observation model and the predicted observation are given as

$$\mathcal{Y}_{i(k+1,k)} = h\left[\mathcal{X}_{i(k+1,k)}\right] \tag{40}$$

$$\hat{\mathcal{Y}}_{(k+1,k)} = \sum_{i=0}^{2M} W_i \, \mathcal{Y}_{i(k+1,k)} \tag{41}$$

In the update step, the corrected state estimate and the corresponding error

covariance matrix are represented by (Laviola, 2003; Chatzi & Smyth, 2009)

$$\hat{x}_{(k+1,k+1)} = \hat{x}_{(k+1,k)} + KG\left(\mathcal{Y}_{(k+1)} - \hat{\mathcal{Y}}_{(k+1,k)}\right) \tag{42}$$

$$P_{(k+1,k+1)} = P_{(k+1,k)} - (KG)P_{yy(k+1,k)}(KG)^T \tag{43}$$

$$P_{yy(k+1,k)} = \sum_{i=0}^{2M} W_i \left[\mathcal{Y}_{i(k+1,k)} - \hat{\mathcal{Y}}_{(k+1,k)}\right]\left[\mathcal{Y}_{i(k+1,k)} - \hat{\mathcal{Y}}_{(k+1,k)}\right]^T + R \tag{44}$$

$$P_{xy(k+1,k)} = \sum_{i=0}^{2M} W_i \left[\mathcal{X}_{i(k+1,k)} - \hat{x}_{(k+1,k)}\right]\left[\mathcal{Y}_{i(k+1,k)} - \hat{\mathcal{Y}}_{(k+1,k)}\right]^T \tag{45}$$

where R is the measurement noise covariance matrix and KG is the Kalman gain given

by

$$KG = \left(P_{xy(k+1,k)}\right)\left(P_{yy(k+1,k)}\right)^{-1} \tag{46}$$

3.4. Particle Filter

In this subsection, a brief overview of the particle filter (PF) method is presented for completeness purposes. The particle filter methodology is similar to the sequential importance sampling approach with a re-sampling step, and is also known as sequential Monte Carlo method (SMC). It consists of approximating the posterior probability distributions of the system state, given a weighted set of random samples and noisy measurements (Chatzi & Smyth, 2009; Chatzi & Smyth, 2013). This Monte Carlo-based methodology can be used instead of some Kalman filter variations methods, such as the ensemble Kalman filter and the unscented Kalman filter, in estimating the state and parameters of the dynamical system given some noisy observations, in case the number of samples used is large enough (Roth, Fritsche, Hendeby, & Gustafison, 2015). In such a case, the particle filter is able to represent the actual posterior distributions and the solution reaches the optimal Bayesian estimate. Unfortunately, this increase in the number of samples will definitely lead to an increase in the computational burden. Another drawback of the particle filter is dealing with the "degeneracy" problem, where particles with negligible weights are taken into account to approximate the PDF. This deficiency in the particle filter can be solved using the resampling technique that throws out the particles with negligible weights and preserves the ones with larger contributions (Arulampalam, Maskell, Gordon, & Clapp, 2002).

3.5. Ensemble Kalman Filter

The Ensemble Kalman Filter (EnKF) was introduced by Evensen (Evensen, 1994) to overcome some of the limitations of the standard Kalman Filter and the Extended Kalman Filter, and then improved and developed in many works (Burgers,

Leeuwen, & Evensen, 1998; Evensen, 2003; Welch & Bishop, 2006; Gillijns S. , et al., 2006; Evensen, 2009). The EnKF is based on Monte Carlo sampling; it propagates an ensemble of realizations forward in time and then corrects them whenever measurements are available.

The EnKF consists of first evaluating the ensemble matrix A, holding the ensemble members x_i, (Chatzi & Smyth, 2009)

$$A = (x_1, x_2, \dots, x_N) \quad A \in R^{n \times N}, x_i \in R^n \tag{47}$$

where n is the size of the model state vector and N is the number of ensemble members. The ensemble mean and ensemble perturbation matrices are respectively evaluated as follows,

$$\bar{A} = A1_N \quad \bar{A} \in R^{n \times N} \tag{48}$$

$$A' = A - \bar{A} = A(I - 1_N) \quad A' \in R^{n \times N} \tag{49}$$

where $1_N \in R^{N \times N}$ is a matrix having its elements equal to $1/N$. The ensemble covariance matrix is next calculated by

$$P = \frac{1}{N-1} A' A'^T \quad P \in R^{n \times n} \tag{50}$$

The analysis equation is the following

$$A^a = A^f + KG(D - HA^f) \tag{51}$$

where KG is the same Kalman Gain used in the standard KF

$$KG = P^f H^T (HP^f H^T + R)^{-1} \tag{52}$$

D is the ensemble of observation matrix holding the measurement vectors $d \in R^m$

$$D = (d_1, d_2, \dots, d_N) \quad D \in R^{m \times N} \tag{53}$$

43

where m is the number of measurements, and

$$d_j = d + \epsilon_j \quad j = 1, \dots, N \tag{54}$$

H is the measurement operator connecting the true state to the observations.

R is the measurement error covariance matrix defined by

$$R = \frac{1}{N-1} \gamma \gamma^T \quad R \in R^{m \times m} \tag{55}$$

where γ is the ensemble of perturbations expressed as

$$\gamma = (\epsilon_1, \epsilon_2, \dots, \epsilon_N) \quad \gamma \in R^{m \times N} \tag{56}$$

3.6. Polynomial Chaos Kalman Filter

Although the EnKF solves the major limitations of the standard KF and the ExKF, it faces some challenges in accurately approximating the model state when a small ensemble size is used. So for highly nonlinear problems, the EnKF requires a large ensemble size which increases the computational cost, that's why the Polynomial Chaos Kalman Filter (PCKF) was proposed to be used instead (Saad, Ghanem, & Masri, 2007; Saad & Ghanem, 2009; Saad & Ghanem, 2011). It should be noted here that a good number of works used the polynomial chaos theory for different purposes in the past years. A stochastic finite element procedure (SFEP) has been proposed by R. Ghanem and P.D. Spanos, then investigated by B. Sudret *et al.* (Sudret & Der Kiureghian, 2000; Ghanem & Spanos, 2003; Sudret, Berveiller, & Lemaire, 2006). The method represents the input random variables by Hermite series expansion and the response by the polynomial chaos expansion. It is proven that the proposed methodology is a convenient non-sampling method giving acceptable results while

reducing the computational burden needed. R. Ghanem *et al.* proposed a robust

methodology to predict the dynamical system evolution in the presence of stochastic

uncertainty by representing the random quantities in the dynamical system by their

polynomial chaos representations (Ghanem, Masri, Pellissetti, & Wolfe, 2005). L.

Nechak *et al.* used both intrusive and non-intrusive methods based on the polynomial

chaos theory to study the dynamical behavior of friction systems (Nechak, Berger, &

Aubry, 2011). V. Keshavarzzadeh *et al.* suggested a method based on nonlinear

sequence transformations to improve the accuracy and convergence of polynomial

chaos solutions when dealing with high-dimensional problems (Keshavarzzadeh,

Ghanem, Masri, & Aldraihem, 2014).

In the PCKF (Saad, Ghanem, & Masri, 2007; Saad & Ghanem, 2009; Saad &

Ghanem, 2011), the forecast step is based on Galerkin projection method.

The matrix holding the chaos coefficients is as follow,

$$A = (x_0, x_1, \dots, x_P) \quad A \in R^{n \times (P+1)} \tag{57}$$

where *x₀* is the mean of the model state *x*, stored in the first column of *A*, $\{x_1, \dots, x_P\}$ are the

model state perturbations, stored in the remaining columns of *A*, *P+1* is the total number of

terms in the polynomial chaos representation of the model state, n is the size of the model

state vector *x* that is represented as

$$x = \sum_{i=0}^{P} x_i(x) \psi_i\big(\xi(\theta)\big) \quad x \in R^n \tag{58}$$

where $\{\psi_i\}$ is the set of Hermite polynomial functions of the Gaussian random variables ξ.

The model state error covariance matrix is given by

$$P = \sum_{i=1}^{P} x_i x_i^T \langle \psi_i^2 \rangle \quad P \in R^{n \times n} \tag{59}$$

where the operator $\langle . \rangle$ represents the expected value.

Given a measurements vector d, its polynomial chaos representation is as follows

$$d = \sum_{i=0}^{P} d_i \psi_i(\xi(\theta)) \quad d \in R^m \tag{60}$$

where m is the total number of measurements, d_0 is the mean, given from the actual measurement vector, and $\{d_1, ..., d_P\}$ are the measurement uncertainties. The polynomial chaos representation of d can be stored in matrix B

$$B = (d_0, d_1, ..., d_P) \quad B \in R^{m \times (P+1)} \tag{61}$$

The measurement error covariance matrix can then be represented as

$$R = \sum_{i=1}^{P} d_i d_i^T \langle \psi_i^2 \rangle \quad R \in R^{m \times m} \tag{62}$$

The analysis/corrector step is stated as follows

$$A^a = A^f + KG(B - HA^f) \tag{63}$$

where H is the observation matrix and KG is the Kalman gain, having the same formulation as the one used in the standard KF,

$$KG = P^f H^T (H P^f H^T + R)^{-1} \tag{64}$$

CHAPTER 4

COMPARISON BETWEEN THE ENSEMBLE AND THE POLYNOMIAL CHAOS KALMAN FILTERS FOR UNCERTAINTY QUANTIFICATION

The difficulty in modeling complex and nonlinear structural systems lies in the presence of different independent sources of uncertainties. Emphasis on the importance of uncertainty quantification is illustrated in this chapter through a numerical problem presenting a comparison between intrusive techniques, represented by the Polynomial Chaos Kalman filter, and non-intrusive techniques, represented by the Ensemble Kalman filter, in quantifying uncertainties due to parametric, model and measurement errors. While the EnKF method is based on Monte Carlo simulation and uses a black-box model to propagate an ensemble of realizations forward in time, the PCKF approach propagates the polynomial chaos representations of the unknown states and parameters to identify the system responses and detect the damage.

4.1. Numerical Example

The numerical problem in this chapter consists of a four-degrees of freedom system as shown in Figure 3 below. The performance and robustness of both the ensemble Kalman filter and the polynomial chaos Kalman filter methods are tested on this numerical problem through the estimation of the displacement and velocity of each DOF as well as the system's unknown parameters. The displacements and velocities of the different DOFs of the system are assumed to be all monitored. A pre-defined damage of

the first DOF is imposed 10 seconds after the excitation hits the system. Additionally, all DOFs are assumed to undergo hysteretic behaviors characterized by the Bouc-Wen model. The mass for each DOF is assumed to be equal such that $M_1 = M_2 = M_3 = M_4 = 2\ Kg$ and the system is assumed to be subjected to El-Centro earthquake ground excitation (Ghanem & Ferro, 2006; Chatzi & Smyth, 2009; Saad & Ghanem, 2011).
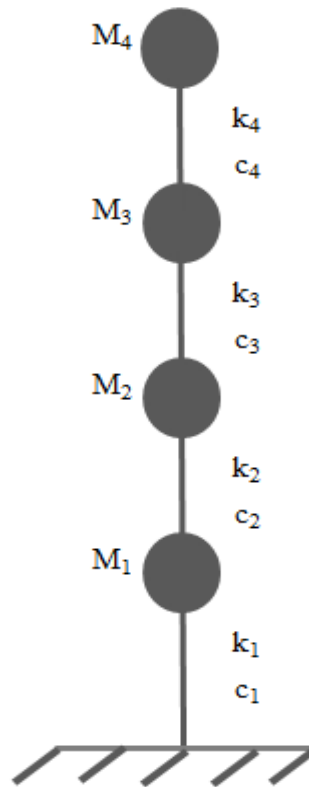

Figure 3. Four-DOF System

This numerical problem is composed of two main models. For both filters, the EnKF and the PCKF, the first model consists of the forward model, which is represented by the Bouc-Wen model that is used to synthetically generate the measurements data. Therefore the equation of motion is expressed as

$$M\ddot{u} + C\dot{u}(t) + \propto K_{el}u(t) + (1-\propto)K_{in}z(x,t) = -M\tau\ddot{u}_g(t) \qquad (65)$$

$$
\begin{bmatrix} M_1 & 0 & 0 & 0 \\ 0 & M_2 & 0 & 0 \\ 0 & 0 & M_3 & 0 \\ 0 & 0 & 0 & M_4 \end{bmatrix}
\begin{bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \\ \ddot{u}_3 \\ \ddot{u}_4 \end{bmatrix}
+
\begin{bmatrix} c_1+c_2 & -c_2 & 0 & 0 \\ -c_2 & c_2+c_3 & -c_3 & 0 \\ 0 & -c_3 & c_3+c_4 & -c_4 \\ 0 & 0 & -c_4 & c_4 \end{bmatrix}
\begin{bmatrix} \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \\ \dot{u}_4 \end{bmatrix}
$$

$$
+\alpha
\begin{bmatrix} k_1+k_2 & -k_2 & 0 & 0 \\ -k_2 & k_2+k_3 & -k_3 & 0 \\ 0 & -k_3 & k_3+k_4 & -k_4 \\ 0 & 0 & -k_4 & k_4 \end{bmatrix}
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}
\qquad (66)
$$

$$
+(1-\alpha)
\begin{bmatrix} k_1+k_2 & -k_2 & 0 & 0 \\ -k_2 & k_2+k_3 & -k_3 & 0 \\ 0 & -k_3 & k_3+k_4 & -k_4 \\ 0 & 0 & -k_4 & k_4 \end{bmatrix}
\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}
=
\begin{bmatrix} F_1(t) \\ F_2(t) \\ F_3(t) \\ F_4(t) \end{bmatrix}
$$

where M is the mass matrix, C is the damping matrix, K_{el} and K_{in} are respectively the elastic and inelastic stiffness matrices and are both assumed to be equal to the ordinary stiffness matrix of the system, $\propto$ is the ratio of the post yielding stiffness to the elastic stiffness and is taken to be equal to 0.15 for this special numerical problem, τ is an influence vector, u is the displacement vector and z is the evolutionary hysteretic vector of dimension n and whose i^{th} component is expressed by the Bouc-Wen model by

$$\dot{z}_i = A_i\dot{x}_i - \beta_i|\dot{x}_i||z_i|^{n_i-1}z_i - \gamma_i\dot{x}_i|z_i|^{n_i} \quad i = 1,\dots,n \qquad (67)$$

where $A = 1$, x is the inter-story drift vector and $\beta, n \ and \ \gamma$ are the Bouc-Wen hysteretic model parameters. Parameter n is taken to be equal to 1 for simplicity reasons to avoid using Taylor series approximations for the power of a non-polynomial for the PCKF method. The purpose of this numerical problem is to identify the states of the system as

well as the parameters $\beta_i, \gamma_i, k \ and \ c$, where $i = 1, \dots, 4$ is the number of degrees of freedom.

To synthetically generate the measured data, the stiffness is assumed to be constant and equal to $k = 8.5 \ N/m$ on all the degrees of freedom, the damping is also assumed to be constant for all DOFs such that $c = 0.27$, and the values of the Bouc-Wen hysteretic parameters, before the damage occurs at the first DOF of the system, are assumed to be $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 2 \ and \ \gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1$ (Chatzi & Smyth, 2009). An increase of 50% to the values of hysteretic parameters of the first DOF is added once the damage is imposed to create a softening character (Shen, Golnaraghi, & Heppler, 2005). Therefore, the values of $\beta \ and \ \gamma$ of the first DOF ten seconds after the excitation hits the system under consideration become $\beta_1 = 3 \ and \ \gamma_1 = 1.5$. Measurements are assumed to available every 20 time steps with a fixed time step $dt = 0.01$ seconds. It should be noted that a sensitivity analysis was performed to select this time step value. An additive Gaussian white noise perturbation with a standard deviation equals to 3% of the exact data is added to the simulated displacements and velocities of the system's DOFs to represent the measurement errors. The fourth-order Runge-Kutta integration is used to solve the differential equation (66) to determine the system responses of the displacements and velocities.

The state variable vector and its first order derivative are respectively represented in the following two equations as

$$X = [u_1 \ u_2 \ u_3 \ u_4 \ \dot{u}_1 \ \dot{u}_2 \ \dot{u}_3 \ \dot{u}_4 \ z_1 \ z_2 \ z_3 \ z_4 \ \beta_1 \ \beta_2 \ \beta_3 \ \beta_4 \ \gamma_1 \ \gamma_2 \ \gamma_3 \ \gamma_4 \ k \ c]^T \quad (68)$$

$$\dot{X}$$

$$= \begin{bmatrix} \dot{X}_1 & \dot{X}_2 & \dot{X}_3 & \dot{X}_4 & \dot{X}_5 & \dot{X}_6 & \dot{X}_7 & \dot{X}_8 \dot{X}_9 & \dot{X}_{10} & \dot{X}_{11} & \dot{X}_{12} & \dot{X}_{13} & \dot{X}_{14} & \dot{X}_{15} & \dot{X}_{16} & \dot{X}_{17} & \dot{X}_{18} & \dot{X}_{19} & \dot{X}_{20} & \dot{X}_{21} & \dot{X}_{22} \end{bmatrix}^T$$

$$= \begin{bmatrix} \dot{u}_1 & \dot{u}_2 & \dot{u}_3 & \dot{u}_4 & \ddot{u}_1 & \ddot{u}_2 & \ddot{u}_3 & \ddot{u}_4 & \dot{z}_1 & \dot{z}_2 & \dot{z}_3 & \dot{z}_4 & \dot{\beta}_1 & \dot{\beta}_2 & \dot{\beta}_3 & \dot{\beta}_4 & \dot{\gamma}_1 & \dot{\gamma}_2 & \dot{\gamma}_3 & \dot{\gamma}_4 & \dot{k} & \dot{c} \end{bmatrix}^T$$

$$= \begin{bmatrix}
X_5 \\
X_6 \\
X_7 \\
X_8 \\
-\dfrac{1}{M_1}\begin{bmatrix} -\ddot{u}_g + (X_{17}+X_{18})X_5 - X_{18}X_6 \\ +\alpha[(X_{13}+X_{14})X_1 - X_{14}X_2] \\ +(1-\alpha)[(X_{13}+X_{14})X_9 - X_{14}X_{10}] \end{bmatrix} \\
-\dfrac{1}{M_2}\begin{bmatrix} -\ddot{u}_g - X_{18}X_5 + (X_{18}+X_{19})X_6 - X_{19}X_7 \\ +\alpha[-X_{14}X_1 + (X_{14}+X_{15})X_2 - X_{15}X_3] \\ +(1-\alpha)[-X_{14}X_9 + (X_{14}+X_{15})X_{10} - X_{15}X_{11}] \end{bmatrix} \\
-\dfrac{1}{M_3}\begin{bmatrix} -\ddot{u}_g - X_{19}X_6 + (X_{19}+X_{20})X_7 - X_{20}X_8 \\ +\alpha[-X_{15}X_2 + (X_{15}+X_{16})X_3 - X_{16}X_4] \\ +(1-\alpha)[-X_{15}X_{10} + (X_{15}+X_{15})X_{11} - X_{16}X_{12}] \end{bmatrix} \\
-\dfrac{1}{M_4}\begin{bmatrix} -\ddot{u}_g - X_{20}X_7 + X_{20}X_8 \\ +\alpha[-X_{16}X_3 + X_{16}X_4] \\ +(1-\alpha)[-X_{16}X_{11} + X_{16}X_{12}] \end{bmatrix} \\
X_5 - X_{13}|X_5|X_9 - X_{17}X_5|X_9| \\
X_6 - X_{14}|X_6|X_{10} - X_{18}X_6|X_{10}| \\
X_7 - X_{15}|X_7|X_{11} - X_{19}X_7|X_{11}| \\
X_8 - X_{16}|X_8|X_{12} - X_{20}X_8|X_{12}| \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix} \tag{69}$$

where $\ddot{u}_g$ is the El-Centro earthquake ground excitation.

51

For both filters, the EnKF and the PCKF methods, the second model consists of an inverse model that is used to detect the behavior of the system under consideration. The inverse model is also expressed by the Bouc-Wen model.

The initial values of the displacements u_i, velocities $\dot{u}_i$ and evolutionary hysteretic vector z_i $(i = 1, ... , 4)$ are assumed to be the following

$$u_1^0 = u_2^0 = u_3^0 = u_4^0 = 0;$$

$$\dot{u}_1^0 = \dot{u}_2^0 = \dot{u}_3^0 = \dot{u}_4^0 = 0; \tag{70}$$

$$and\ z_1^0 = z_2^0 = z_3^0 = z_4^0 = 0$$

The initial guesses of the unknown parameters of the system are assumed to have the following mean values

$$\beta_1^0 = 2.5;\ \beta_2^0 = 2.5;\ \beta_3^0 = 2.5;\ \beta_4^0 = 2.5;$$

$$\gamma_1^0 = 1.2;\ \gamma_2^0 = 1.2;\ \gamma_3^0 = 1.2;\ \gamma_4^0 = 1.2; \tag{71}$$

$$k^0 = 7\ and\ c^0 = 0.4$$

and a standard deviation of 5% of the initial assumptions.

A fourth order Runge-Kutta time integrating method is implemented once again to propagate the system state forward in time. An additive Gaussian white noise having a standard deviation equals to 0.5% of the forecasted state vector and 1% of the forecasted parameters, is used every 5 time steps to represent the model uncertainty. The perturbed synthetic measurements of the displacements and velocities of the DOFs are used to calibrate the model parameters and estimate the response of the system under consideration.

For the PCKF method, The Galerkin projection method is used to solve the equation (66) to determine the acceleration $\ddot{u}_i$ $(i = 1, \dots, 4)$. On the other hand, to solve equation (67) and determine the evolutionary hysteretic vector z_i $(i = 1, \dots, 4)$, the Galerkin projection cannot be used because of the presence of the absolute value in the equation, and a linear fitting method (i.e. regression method) is used instead although it requires more computational effort than the Galerkin projection approach. It should be noted that this non-intrusive method used to solve for z_i is an approximation method, suffering from errors due to the truncation.

Representation of the Different Sources of Uncertainty

Regarding the PCKF method, the dimension is specified based on the number of independent sources of uncertainty available in the system under consideration. Therefore, one major challenge faced in this numerical problem was the increase in the dimensionality of the PCE due to the presence of different independent sources of uncertainties, i.e. every time a measurement is recorded or a model error is implemented in the system to account for the physical and mathematical model simplifications, an additional increase in the dimensionality of the PCE is incorporated. To solve this problem, the PCE bases are assumed to be limited to finite terms while maintaining a good enough approximate propagation of the covariance matrix and the parameters' means, leading to a relative decrease in the dimensionality of the PCE under consideration (Slika & Saad, 2016; Slika & Saad, 2016).

Therefore, the errors due to model uncertainty are first assumed to be correlated for each state $(u, \dot{u} \text{ and } z)$ and each parameter $(\beta, \gamma, c \text{ and } k)$, resulting in a total of 7 independent sources of model errors. Next, the errors due to measurement uncertainties

are also assumed to be respectively correlated for u and $\dot{u}$, leading to two additional independent sources of errors. Finally, the initial guess errors are taken to be correlated for each parameter $(\beta, \gamma, c \text{ and } k)$ respectively, resulting in 4 additional independent sources of errors. Therefore, the minimum allowed dimension of the PCKF method is 34, i.e. a minimum of 4 finite terms are taken for the model error and 1 finite term is assumed for the measurement uncertainty.

It should be noted that for this case, both filters faced a challenge in accurately estimating parameters $\beta \text{ and } \gamma$ for each DOF. To solve this problem, errors representing the model uncertainties in $\beta \text{ and } \gamma$ are respectively assumed to be independent. Therefore, the total number of independent sources of errors due to model uncertainty becomes 13 instead of 7 and the minimum dimension of the PCKF method turns out to be 45 (i.e. a minimum of 3 finite terms are taken for the model error and 1 finite term for the measurement error). Although this assumption results in an increase in the minimum allowed dimension of the PCKF method from 34 to 45, both filters are now able to accurately approximate the unknown system state and parameters and to adequately locate the imposed prescribed damage in time and space.

4.2. Results and Discussions

Before proceeding with the results, it should be noted that sensitivity analyses on the ensemble size of the EnKF method and on the order and dimension of the PCKF method were performed. The mean of the predicted hysteretic parameter β_1 and its standard deviation, ten seconds after the damage is imposed on the first DOF, are calculated for this purpose, for different sizes of the EnKF approach and different orders and dimensions of the PCKF method. While the PCKF method gives the same

outcomes for different simulation runs having same order and dimension of the PCE, the results of the EnKF method slightly vary between different simulation runs for the same ensemble size. For this reason, five simulation runs were respectively executed for each ensemble size. Figures 4 and 5 respectively represent the values of the mean and standard deviation of the predicted parameter β_1 at ten seconds post-damage, calculated using the EnKF and PCKF approaches.

It can be seen from figure 4 that as the ensemble size increases, the variability between the means of the predicted parameters β_1 of the different simulation runs decreases to attain an acceptable variability for a 10,000 ensemble size. This observation is also valid for figure 5, where the results of the standard deviations of the predicted parameters β_1, at ten seconds after the damage is imposed on the first DOF, attain acceptable errors between different simulation runs for an EnKF method with 10,000 ensemble size.

Similarly, as the order of the PCKF method is increased from 1 to 2, the method results in a closer approximation of the mean of the parameter β_1 to the assumed exact measured value, as shown in figure 4, and in a slightly smaller standard deviation of parameter β_1, as shown in figure 5. In addition, for PCKF order 2, a rough plateau in the values of the mean and standard deviation of parameter β_1 is noticeable for different dimensions of the PCE. Therefore, a PCE with order 2 and dimension 45 plays the role of a good parameter estimator. It should be noted that the results for PCE with order 3 are not presented in this dissertation for practicality reasons, since for this case the computational burden of the simulation runs becomes unreasonably expensive and impractical. As a conclusion, an EnKF with 10,000 ensemble size and a PCE with order 2 and dimension 45 can be used for the comparative study in this numerical problem.
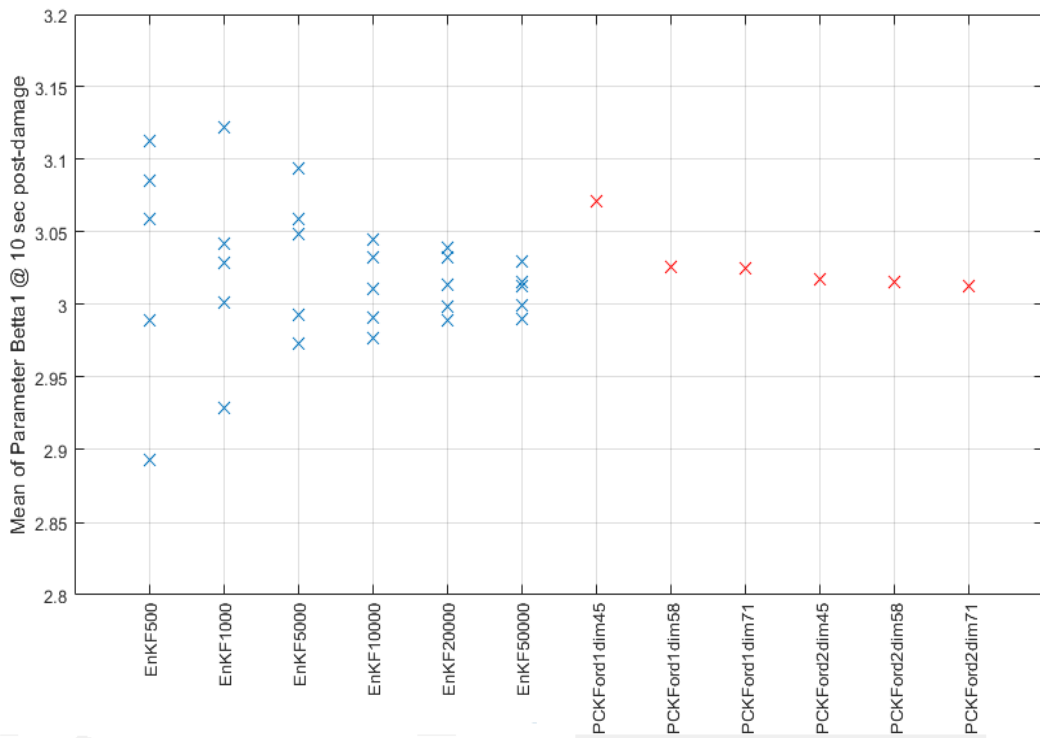
Figure 4. Mean of predicted parameter β_1 vs. Ensemble size and PCKF dimension and order, 10 seconds after the damage
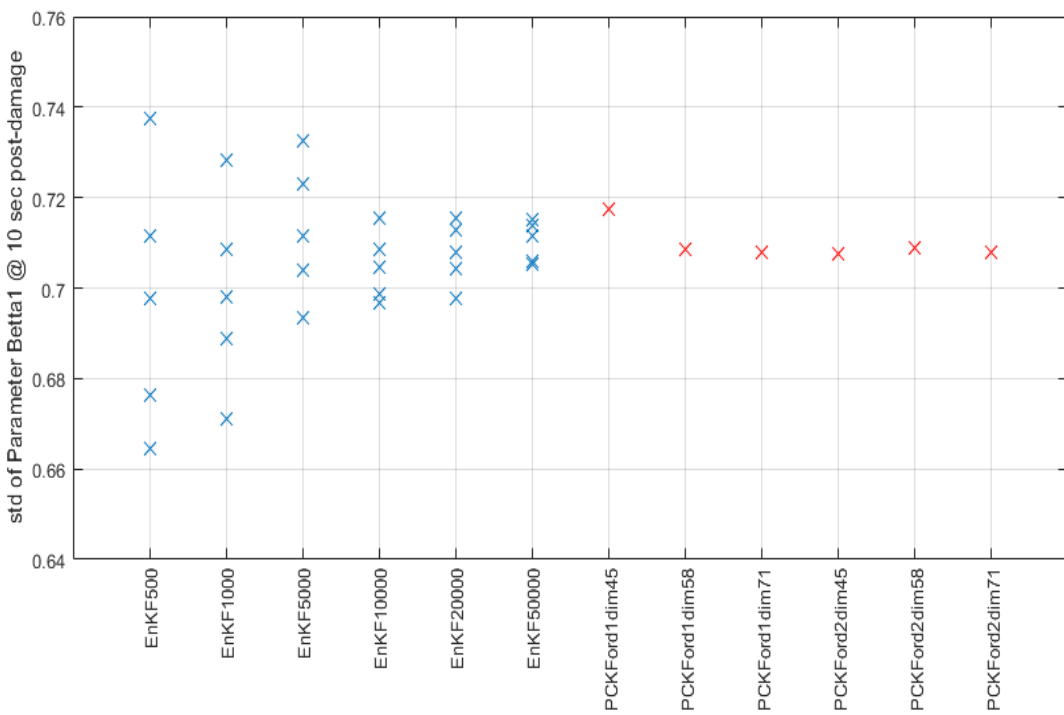

Figure 5. Standard deviation of predicted parameter β_1 vs. Ensemble size and PCKF dimension and order, 10 seconds after the damage

Furthermore, while the duration for the PCKF parameter and state identification with dimension 45 and order 2 was 3.0618×10^3 seconds, the computational time for the EnKF identification with size 10,000 ensemble of realizations was 7.1455×10^3 seconds. As a result, for high accuracy requirements, while both filters are able to approximate the system state and unknown parameters and identify the damage in space and time, the PCKF method outperforms the EnKF approach, that requires high ensemble size to attain high precisions, in terms of computational expenses.

Figures 6 to 9 respectively represent for each DOF, a comparison between the EnKF and PCKF estimates of the displacement (part (a) of each figure), velocity (part (b) of each figure) and evolutionary hysteretic vector (part (c) of each figure), and their respective synthetic actual measured values. It can be clearly seen that there is a very good match between the three plots in each figure, which implies that both variations of the Kalman filter method play the role of very good estimators of the system state.
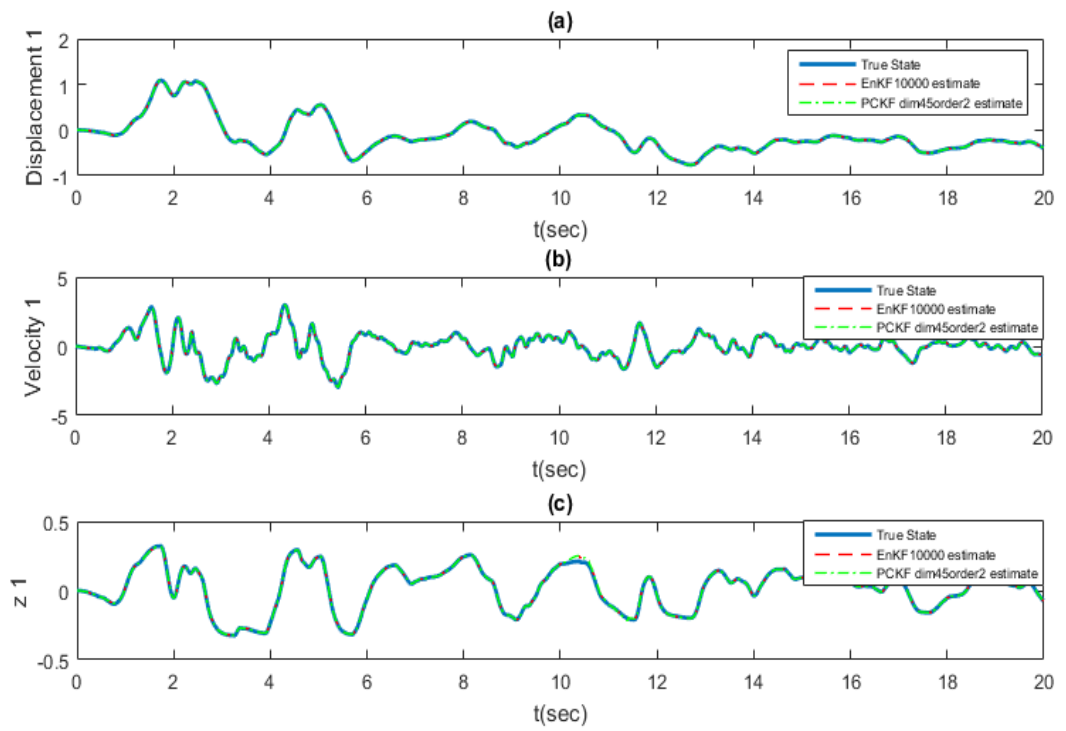
Figure 6. First DOF (a) displacement, (b) velocity and (c) evolutionary hysteretic vector
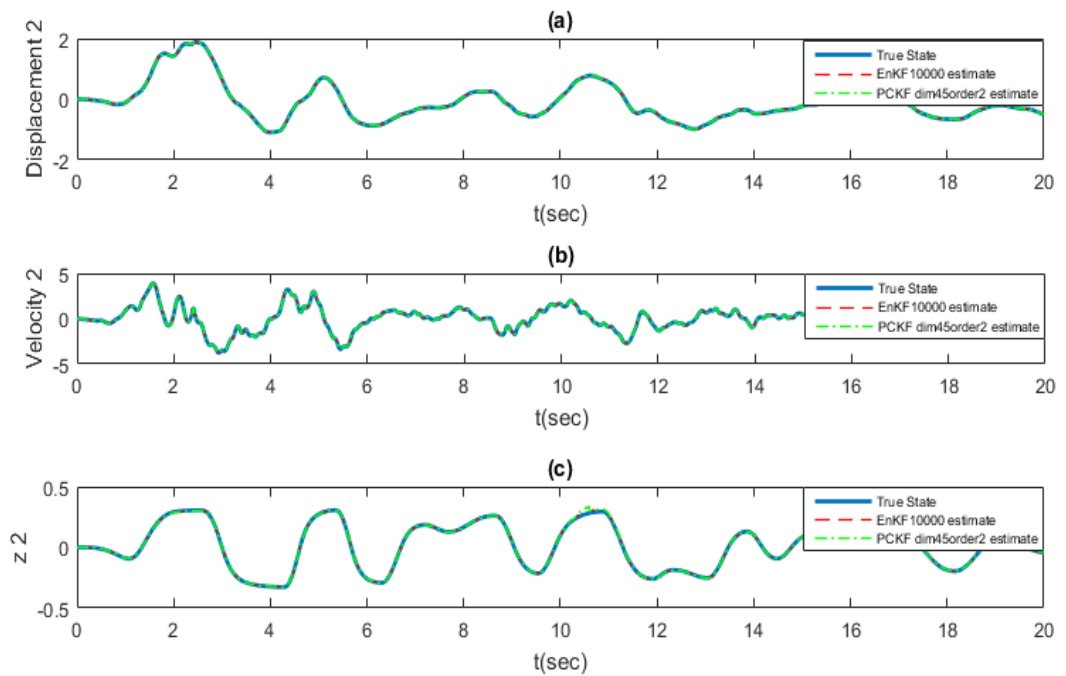

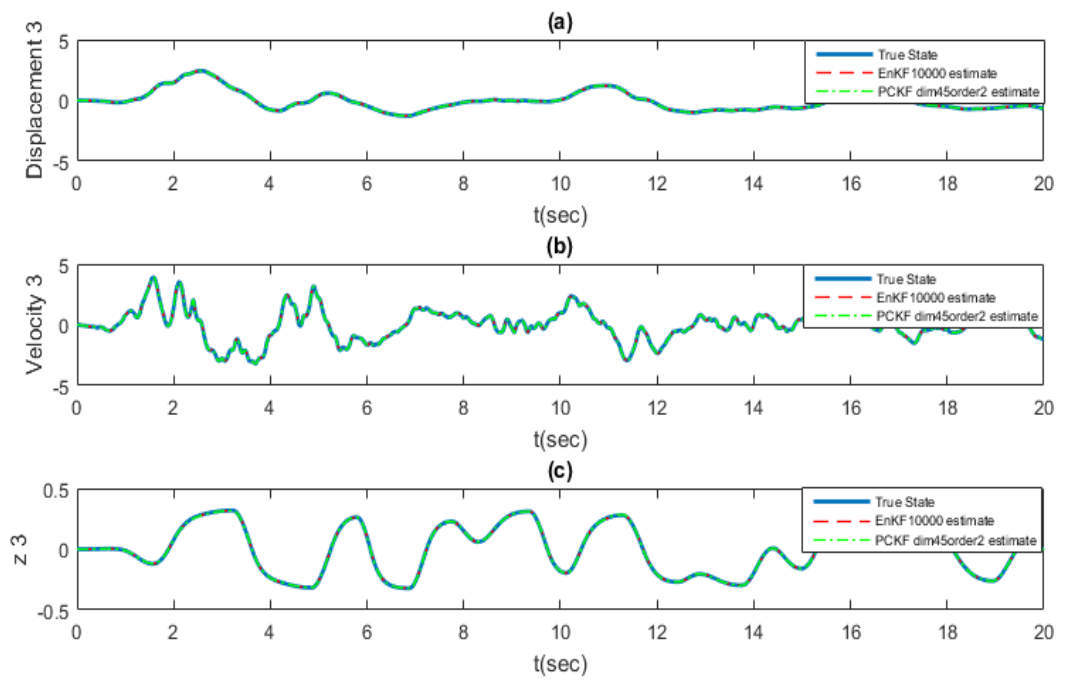**Figure 7.** Second DOF (a) displacement, (b) velocity and (c) evolutionary hysteretic vector

Figure 8. Third DOF (a) displacement, (b) velocity and (c) evolutionary hysteretic vector
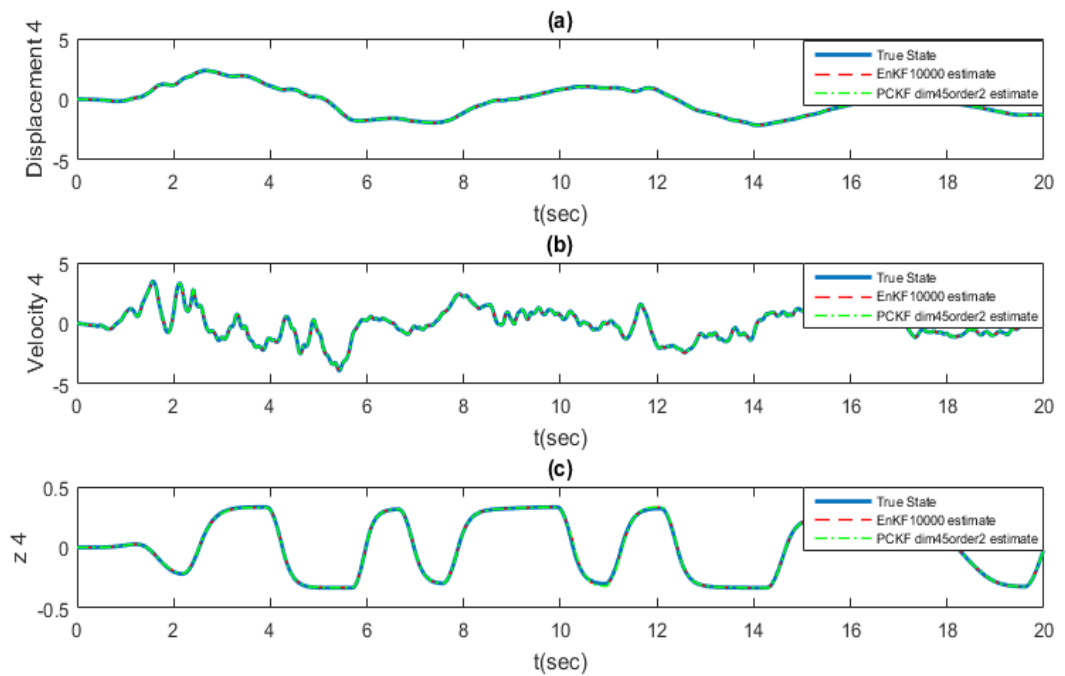

Figure 9. Fourth DOF (a) displacement, (b) velocity and (c) evolutionary hysteretic vector

The EnKF and PCKF estimates of the hysteretic model parameters β and γ of each DOF are respectively presented in figures 10 and 11. A perfect match between the EnKF and PCKF estimates of the different hysteretic model parameters can be obviously noticed in each figure. Furthermore, both KF filter variations were able to locate the imposed damage in time and space, which is represented by the jump of the different parameters at 10 seconds, followed by a correction and matching with the true values after few time steps.

Same conclusions can be drawn for parameter k, represented in figure 12 part (a) and parameter c, represented in Figure 12 part (b). Both filters were able to estimate the true values of parameters k and c, even when starting from relatively far initial guess values, and to locate the damage imposed 10 seconds after the excitation hits the system under consideration.
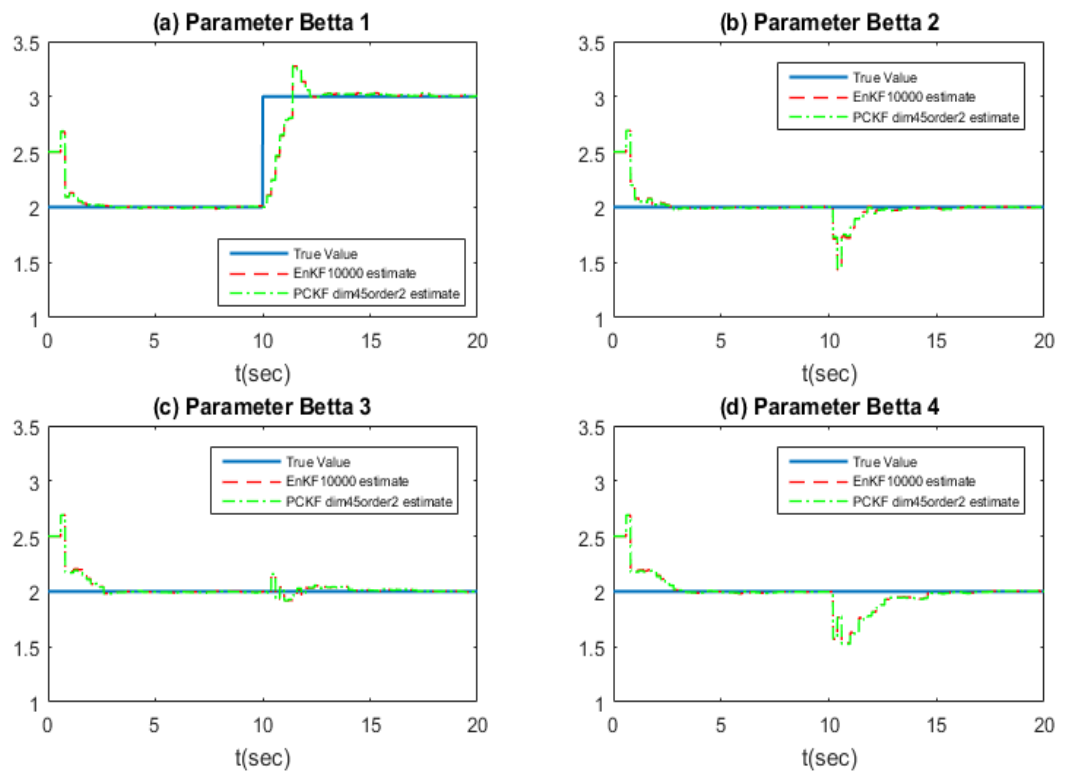
Figure 10. EnKF and PCKF estimates of Parameter β (a) first DOF; (b) second DOF, (c) third DOF and (d) fourth DOF

60
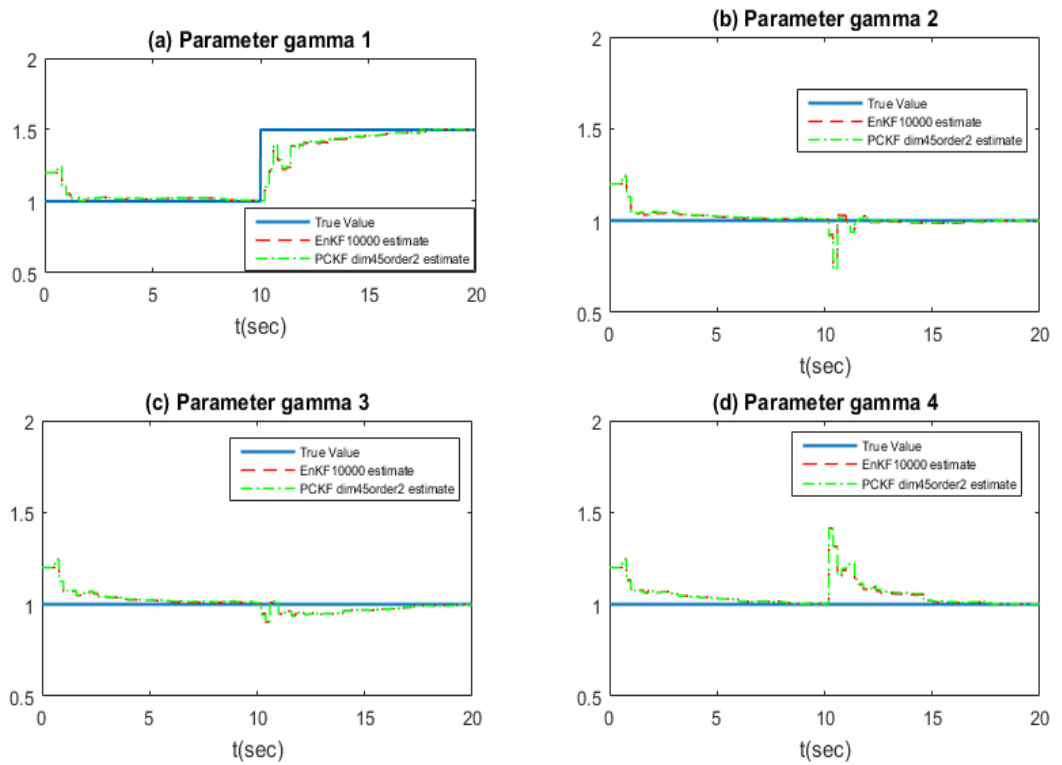
Figure 11. EnKF and PCKF estimates of Parameter γ (a) first DOF; (b) second DOF, (c) third DOF and (d) fourth DOF
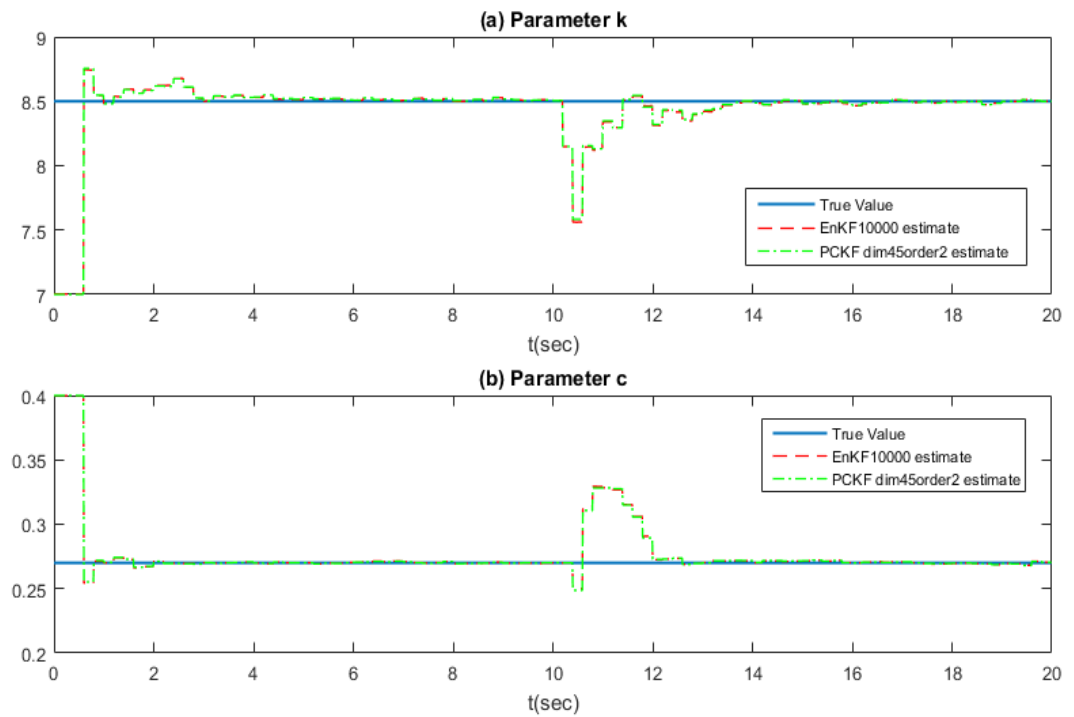
Figure 12. (a) Parameter k, and (b) Parameter c

Figure 13 represents the hysteretic loop corresponding to the first DOF. The nearly perfect match between the actual hysteretic loop and its EnKF and PCKF estimates, authenticates the validity of parameter approximation of both filters.

Figure 13. Hysteresis Loop of the first DOF

CHAPTER 5

OPTIMAL SENSOR PLACEMENT USING A COMBINED GENETIC ALGORITHM – ENSEMBLE KALMAN FILTER FRAMEWORK

With the recent technological advances, the deployment of monitoring devices in structures is becoming more abundant. In an ideal scenario, one would deploy sensors on every corner of a structure, but the problem of dealing with the humongous online data gener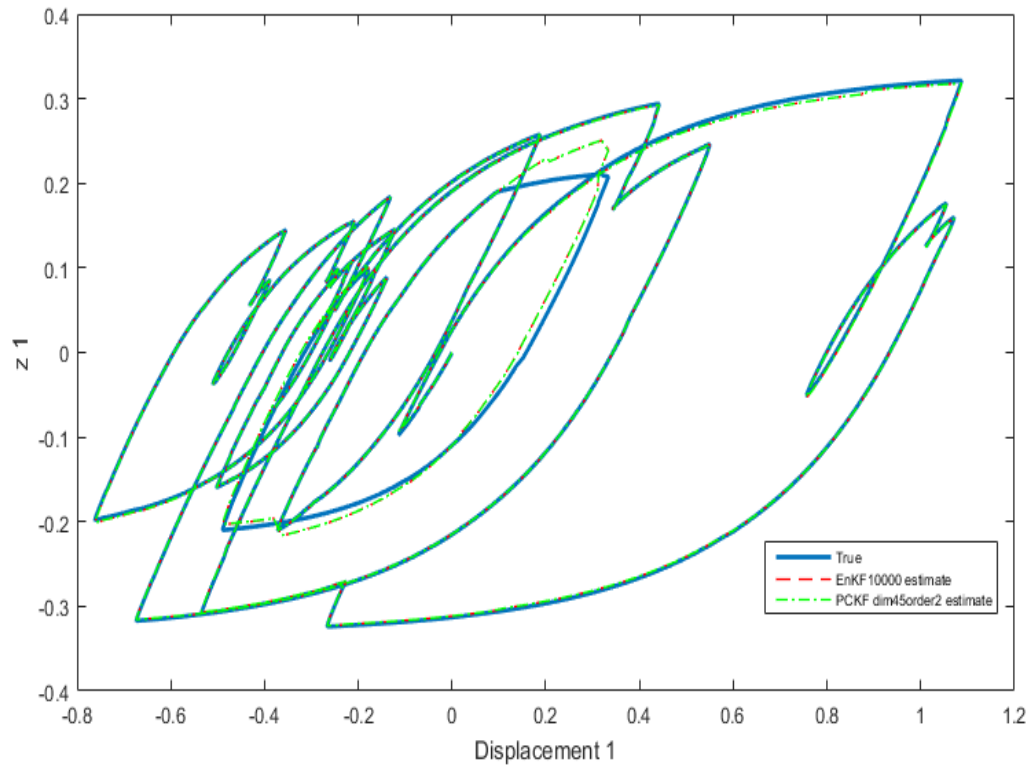ated arises. As previously mentioned, the second objective of this dissertation is to present a novel framework based on combining Genetic Algorithm techniques with the Ensemble Kalman filter approach to identify the optimal sensor locations for structural system identification and damage detection purposes. The GA approach is first used to generate a random initial set of sensor locations, then through a minimization procedure, the best locations of the sensors are determined. The fitness function to be minimized is taken to be the difference between synthetically generated actual measurement data and their respective predicted values, calculated using EnKF through estimating and updating the system state and model parameters.

The proposed framework is general enough to allow applicability to a wide range of structural problems exhibiting hysteretic behaviors and suffering from limited real-time measured data. The use of the EnKF for evaluating the fitness function for the GA allows for quantifying the uncertainty within the system by relying on a Monte Carlo scheme. Although the GA might not rapidly converge to the optimal solution; in the presented setting, the Genetic Algorithm-Ensemble Kalman Filter (GA-EnKF) framework is carried out in the planning and design stages where ample time is

63

available for optimization purposes. A limitation on the use of the GA-EnKF approach may be encountered if the presented algorithm is to be used for online identification purposes. The robustness and efficiency of the suggested methodology are tested through a sensitivity analysis on the initial positions of the sensors of a ten-story shear building subjected to El-Centro earthquake excitation at its base, where the percentages of convergence of the GA-EnKF scheme to the optimal solutions, calculated using the brute-force search method, are analyzed. The sensitivity of the presented framework to the population size of the GA is also assessed.

5.1. The Genetic Algorithm (GA)

The genetic algorithm is a search technique that uses random natural evolution operations to solve optimization problems. It was first introduced by John Holland in 1975 (Holland, 1975). The algorithm starts by randomly creating a population of individuals, from which fittest parents are selected. These initial parents undergo a series of natural evolution operations to create new better offsprings, from which the best new parents are selected based on their fitness. This is repeated until a certain termination criterion is satisfied.

The GA can be summarized in the following outline (Chow H. , Lam, Yin, & Au, 2011; Lam , Yang, & Hu, 2011; Chou & Ghabboussi, 2001):

1. Initialization: The algorithm starts by randomly generating an initial set of individuals or solutions (Population), where each individual is represented by a chromosome. There are many encoding methods in the literature to represent the individuals of a population, but the most used way is the binary string (1 or 0)

where each chromosome is represented by one binary string. For the case of optimal sensor location problems, if for example two sensors are available to be placed on a four-story building and the individual is encoded as 1010, this means that one sensor is placed on the first floor and the second sensor is placed on the third floor.

2. <u>Fitness evaluation:</u> The next step is to determine the fitness or objective function to be optimized, that is problem dependent, and evaluate the fitness of each individual of the initial population.

3. <u>Creation of new solutions:</u> The following steps must be repeated until convergence:

   i. <u>Selection:</u> In this step, two parents (sub-population) are selected from the population based on their fitness values to produce new offsprings (children). The fitter the chromosome, the better it has chance to be selected to propagate its genetic information.

   ii. <u>Crossover (Recombination):</u> This operation is used to create new offsprings by recombining genes from the selected parent chromosomes. A crossover point or site should be randomly selected, the part of the strings before this point of the first parent is combined with the part of the strings after this point of the second parent to create the first offspring, whereas the second offspring is created by combining the remaining two parts of the strings from the two parents. Figure 14 below shows a simple crossover example.
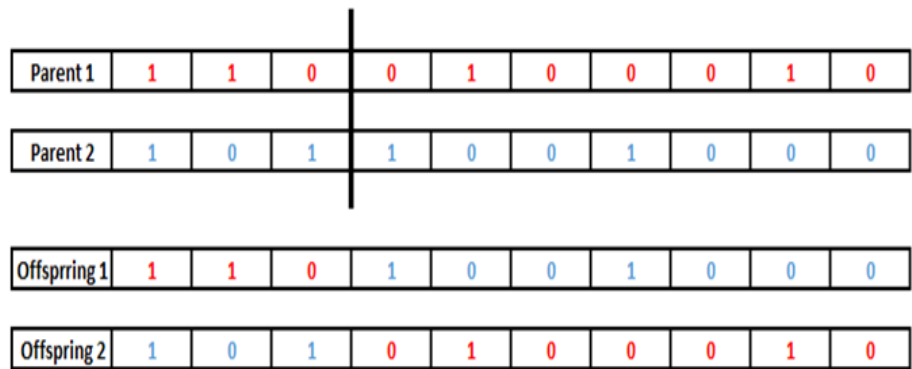
Figure 14. Crossover Example

iii. Mutation: This operation randomly changes some of the portions of
the new individuals to avoid being trapped into a local optimum. For
binary encoding, mutation randomly flips some bits from 0 to 1 or
from 1 to 0. Figure 15 illustrates a simple mutation example.
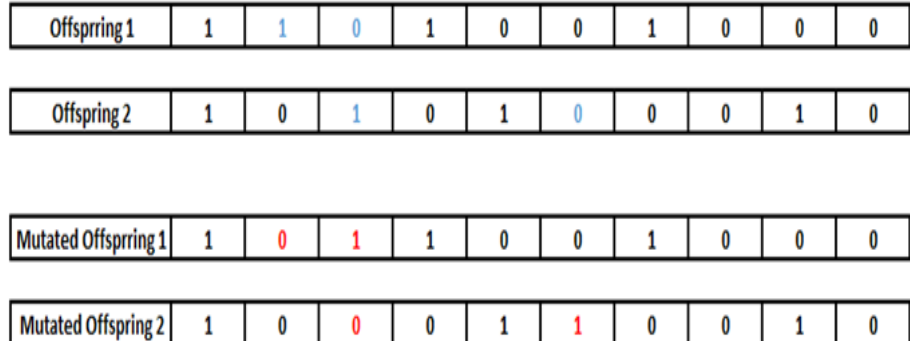
Figure 15. Mutation Example

iv. The new, hopefully better, solutions are used as parents now. The
fitness of the new offsprings should be evaluated and the loop should
be repeated until convergence to the best individuals.

4. Termination conditions: For each problem, many termination conditions can be
predefined, those may include: number of generations, time of the run, plateau
(no more improvement of the best solution), and many others.

66

The general outline of the GA approach is shown in Figure 16 below.
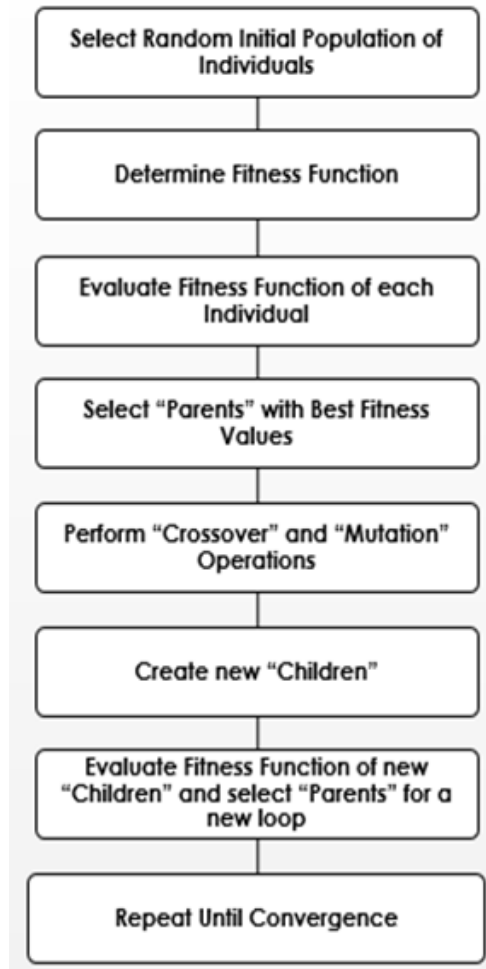
Figure 16. GA general outline

5.2. Genetic Algorithm-Ensemble Kalman Filter (GA-EnKF) Methodology for Optimal Sensor Placement

The main objective of this study is to identify the optimal placement of a fixed number of sensors within a structure. Once a finite number of sensors is selected, an initial population of sensor locations is randomly generated as a first step in the GA approach. The EnKF is then employed to assimilate data for each set of sensor locations identified by the GA independently. The EnKF first propagates an ensemble of

realizations forward in time using the Runge-Kutta time integration scheme, then update and correct the system states and parameters whenever measurements data are available. The aim of the filtering is to estimate the model state and parameters using measurement data from selected sensors only. The updated model parameters are then used to forecast the future response of the structure. The mismatch between the predicted structural response (e.g. displacements, velocities, acceleration …) and the synthetically measured data serves as the objective function for the GA optimization. Once the objective functions corresponding to all individuals within the initial population of sensor locations are evaluated, the ones having the best fitness values (the least mismatch between the measured and predicted data) are selected as the "Parents". New better "Offsprings" (new sensor locations) are created after performing some natural evolution operations, the crossover and mutation operations, on the old parents (old sensor locations). The introduction of these natural evolution operations in the proposed scheme makes the population of sensor locations more diverse in each generation and increases its immunity against being trapped in local minima and thus leading to undesirable premature convergence.

The real-time measured data and the EnKF predicted data corresponding to the new sensor locations are next calculated and the fitness of each of the new solutions (new sensor locations) is evaluated. The best individuals of the population, having the least mismatch between predicted and measured data, are selected as new parents for a new loop. The algorithm is repeated until one of the predefined convergence conditions is satisfied and the best locations of the sensors are determined.

The chart, presented in Figure 17, summarizes the general steps used in the aforementioned methodology.
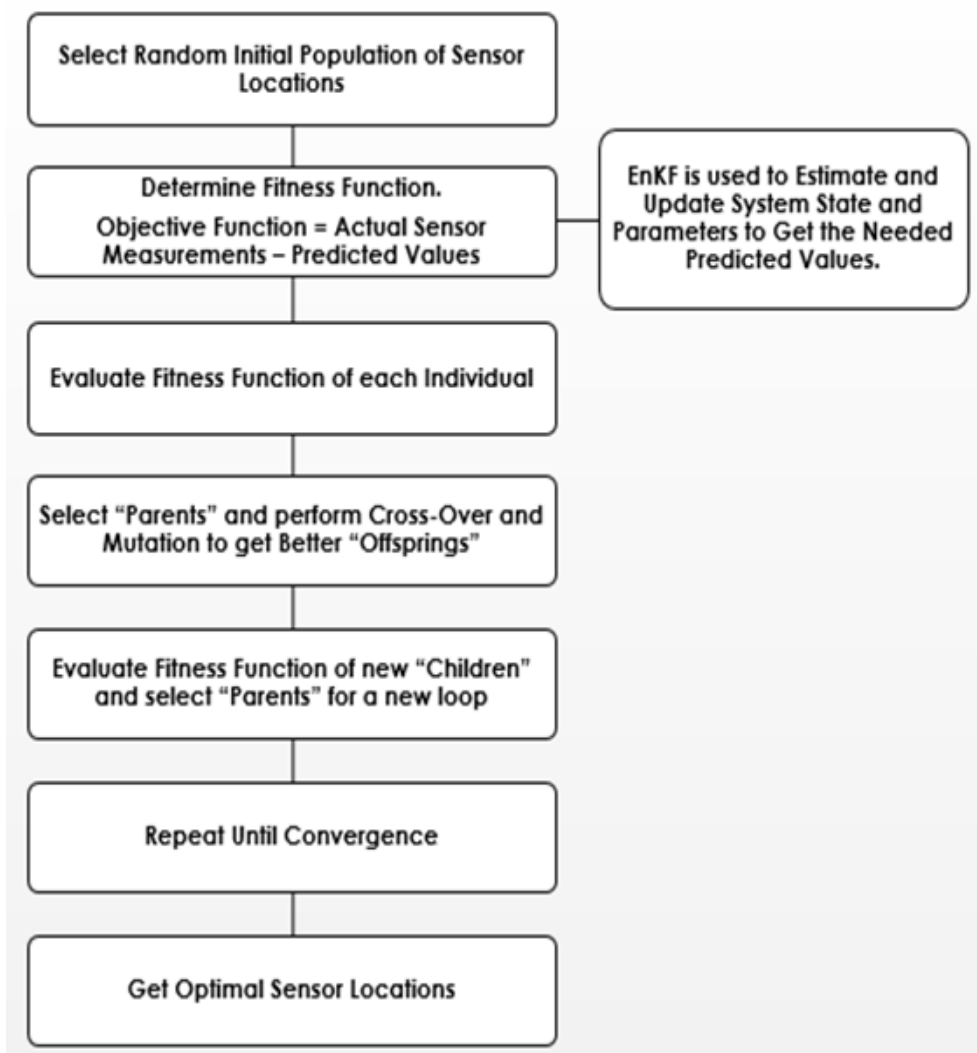
Figure 17. GA-EnKF Methodology Summary Chart

5.3. Numerical Example

The numerical problem in this chapter consists of a ten-degree-of-freedom

shear building, as shown in Figure 18, subjected to an El-Centro earthquake excitation

at its base. The floor mass corresponding to stories one, two and three is taken to be

50,000 Kg each, stories from four till eight have a floor mass of *40,000 Kg* each, and stories nine and ten have a *30,000 Kg* floor mass each. All structural elements of this building are assumed to undergo hysteretic behaviors. To simplify the analysis, a non-parametric model is assumed to simulate the hysteretic behavior where the restoring force function is represented as a truncated polynomial function of the inter-story drifts and velocities of the different floors of the building (Ghanem & Ferro, 2006; Saad, Ghanem, & Masri, 2007; Saad & Ghanem, 2011). Consequently, the equation of motion becomes

$$M\ddot{u}(t) + F(u, \dot{u}) = -M\ddot{u}_g(t) \tag{72}$$

where *M* is the mass matrix and *F* is the non-parametric representation of the nonlinearity (nonlinear restoring force), whose i^{th} component is given by

$$F_i(u, \dot{u}) = a_i(u_i - u_{i-1}) + a_{i+1}(u_i - u_{i+1}) + b_i(u_i - u_{i-1})^3 + b_{i+1}(u_i - u_{i+1})^3 + c_i(\dot{u}_i - \dot{u}_{i-1}) \tag{73}$$
$$+ c_{i+1}(\dot{u}_i - \dot{u}_{i+1}) + d_i(u_i - u_{i-1})(\dot{u}_i - \dot{u}_{i-1}) + d_{i+1}(u_i - u_{i+1})(\dot{u}_i - \dot{u}_{i+1})$$

where $i = 1,..., \# floors$ and $\{a_i\}, \{b_i\}, \{c_i\}, and \{d_i\}$ are the coefficients related to the floor *i* damping and stiffness properties. Coefficients {a$_i$}, multiplying the inter-story drift of floor i, are directly related to the stiffness corresponding to floor i. Coefficients {c$_i$}, multiplying the inter-story velocity of floor i, are directly related to the damping of floor i. Coefficients {b$_i$} and {d$_i$}, respectively multiplying the cube of inter-story drift and the product of inter-story drift and velocity of floor i, are indirectly related to the damping and stiffness properties of floor i. It should be noted that an adaptive refinement procedure was performed to select suitable basis functions for the restoring force *F*, which is a trade-off between the size of the basis and the accuracy of the results.
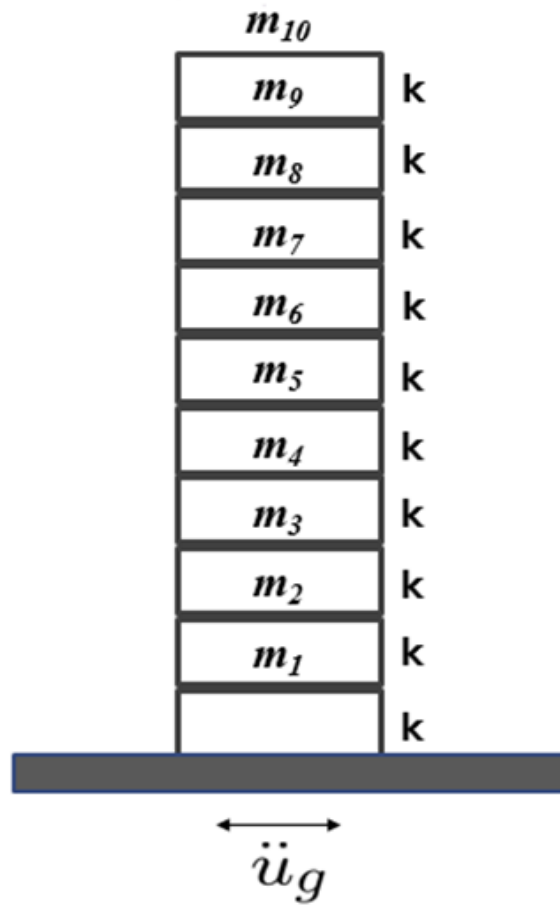
Figure 18. Ten-story shear building

A simpler case study consisting of a four-story shear building subjected to seismic excitation at its base was investigated in an earlier study (Nasr & Saad, 2015). The current study builds on the previous work and is geared to test the efficiency and robustness of the proposed GA-EnKF methodology in determining the best sensor locations for higher dimensional structures through a sensitivity analysis on the initial locations of the sensors (the starting point of the optimization algorithm). In this study, an exhaustive search procedure, taking into account every single possible combination of sensor locations, is performed to determine the optimal scenario and consequently assess the accuracy of the proposed algorithm.

Due to the lack of real time measurement data of the displacement and velocity of each floor, the model parameters listed in table 1 are used to synthetically generate the measured displacements and velocities of the different floors. The parameters are assumed stationary and equal for all floors. However, to represent measurement errors, an additive Gaussian white noise perturbation having a standard deviation equal to 0.5% of the actual data is added to the simulated floors displacements and velocities. Within the optimization framework, it is assumed that the model parameters are unknown a priori, and the approach starts from an initial assumption for the restoring force parameters. The assumed model parameters used for optimization purposes are Gaussian random variables with the initial guess having the mean values shown in table 2 and a standard deviation of 10% of the initial assumption. The perturbed synthetic data are used to calibrate the model parameters and predict the response of the building. The MATLAB's genetic algorithm function (ga) (Mathworks) is combined with an EnKF algorithm to identify the optimal sensor locations. As a base case, the number of individuals in each population is taken to be 50, from which two elite individuals having the best fitness functions are selected. Two main stopping criteria are considered for this problem: (1) the algorithm stops in case the number of generations exceeds 100 generations, or (2) in case after 20 successive generations (stall generations "Stall (G)"), the average change in the fitness function value is less than the function tolerance. An adaptive refinement analysis is performed on the value of the tolerance and the best value for the problem at hand is found to be equal to 10^{-10}.

Table 1. Model parameters used to generate synthetic measurements in forward model

Model Parameters	Values
a	1e8
b	2e5
c	9.4e5
d	4.5e5

Table 2. Initial model parameters used within the optimization

Model Parameters	Initial Mean Values
a	1e7
b	1e4
c	4.7e4
d	1.5e4

Within the framework of the EnKF, an ensemble of state vectors is used to represent the uncertainty within the system. Sensitivity analysis is conducted to optimally select the number of ensembles to be used for the model parameter calibration process. Based on the analysis results, it was decided to use an ensemble size of 400 for adequate representation of the system noise. The initial ensemble is propagated forward in time using the fourth order Runge-Kutta time integration scheme. The model error is represented via an additive Gaussian white noise having a standard deviation equal to 1% of the forecasted state vectors. Whenever measurements are available, the model

state vectors are updated using the Ensemble Kalman Filter equations. After performing a sensitivity analysis on the time step value, a time step of 0.01 seconds is chosen and the measurements are assumed to be available every 10 time steps.

The EnKF updated state vector is used to predict the response of the structure. The L2-norm of the difference between the mean predicted displacement and velocity and the respective measured displacement and velocity for all floors, divided by the synthetically measured data is adopted as the fitness function for the GA optimization. The penalty value is therefore calculated as:

$$\text{Penalty value} = \sqrt{\frac{\sum_{i=1}^{\#floors}\left(\left|u_p^i - u_m^i\right|^2 + \left|v_p^i - v_m^i\right|^2\right)}{\sum_{i=1}^{\#floors}\left(\left|u_m^i\right|^2 + \left|v_m^i\right|^2\right)}} \tag{74}$$

where, $i = 1,..., \#floors$, u^i and v^i correspond to the displacement and velocity of floor i respectively, and the subscripts p and m correspond to the predicted and measured values, respectively. The fittest individuals, having the lowest mismatch between the predicted and actual data, have a better chance to be selected as they will produce better offsprings after undergoing the crossover and mutation processes.

5.4. Results and Discussions

Four different scenarios of available number of sensors are tested, two, three, four and five sensors case studies. For each scenario, the objective is to identify the optimal sensors placement using the GA-EnKF framework so as to minimize the mismatch between the measured and predicted displacement and velocity of each floor.

The brute-force search approach is used to test the convergence of the proposed GA-EnKF framework to the optimal sensor locations. This exhaustive search technique consists of systematically calculating the penalty value corresponding to each possible combination of sensor locations for the four different cases of available sensors. Moreover, a sensitivity analysis study is conducted to evaluate the robustness of the presented approach through random selection of initial locations of the sensors for the four previous cases. For each case, 100 simulations of the GA-EnKF framework are conducted starting with a different initial guess of sensor positions each time, and the percentage of convergence of this proposed methodology to the brute-force search's optimal sensor location is calculated.

5.4.1. Two sensors case

For the case of two available sensors, the penalty values corresponding to the 45 possibilities of sensors placement are calculated using the brute-force search method and the lowest fitness value corresponded to the case where the sensors are placed on floors 1 and 10. For the base case of 50 individuals within each population, 88% of the simulations using the proposed GA-EnKF framework, starting each time from a different random initial guess of sensor locations, converged to the brute-force optimal result.

Figure 19 represents the results of the proposed GA-EnKF framework starting from an initial population of sensors locations randomly selected to be floors 8 and 10 (Initial Population = [8 10]). The best penalty values (dots) and the mean penalty values (circles) at each generation are presented in part (a) of Figure 19. The best sensor locations for this case are found to be on floors 1 and 10 (Figure 19 part (b)). This

75

finding matches the result found using the brute-force search and is very logical since

one can expect that the best scenario is to place one sensor at the top of the structure and

the other at the bottom, when a limited number of sensors are available, to cover all the

building and get the most informative measured data. It can be noticed that the

algorithm converged before attaining the maximum number of generations because

there was no improvement in the value of the fitness function for 20 successive stall

generations (Stall(G)), as shown in part (c) of Figure 19. For this problem, the time and

the Stall time limit (Stall (T)), which is the time over which the fitness function exhibits

no improvement, are not included as stopping criteria.

Figure 19. Two Sensors Scenario: (a) Best and mean fitness values at each generation, (b) Best Individuals (Optimal Sensor Locations), and (c) Stopping Criteria

Figures 20 and 21 show the displacement and velocity estimates of floor 8

based on measurements obtained from sensors placed according to the first generation

76

guess [8 10] and last generation estimate [1 10], respectively. The comparison between these two figures shows good improvement in the mismatch between the EnKF predictions and the synthetic measurement values for both the displacement and the velocity when the sensors are placed at their optimal locations.

Figure 20. Estimates of the eighth floor displacement and velocity at first generation: sensors placed at 8th and 10th floors

Figure 21. Estimates of the eighth floor displacement and velocity at final generation: sensors placed 1st and 10th floors

5.4.2. Three sensors case

For the case of three available sensors, the smallest penalty value among the 120 possible combinations of sensor locations, found using the brute-force search method, corresponded to the candidate [1 7 10] and an 84% convergence to this optimal solution using the proposed framework was recorded for the base case scenario. Figure 22 represents the results of the GA-EnKF algorithm starting by randomly placing the three available sensors on floors 2, 5 and 6. Similar to the previous case, Figure 22 part (a) shows the comparison between the best and mean values of the fitness function. The best sensor configuration is found to be on floors 1, 7 and 10, as shown in Figure 22 part (b). This result matches the brute-force search's best sensor configuration of three available sensors placed on the ten-story building under consideration. In this

78

case, the algorithm also converges before reaching the maximum number of generations as shown in part (c) of Figure 22.
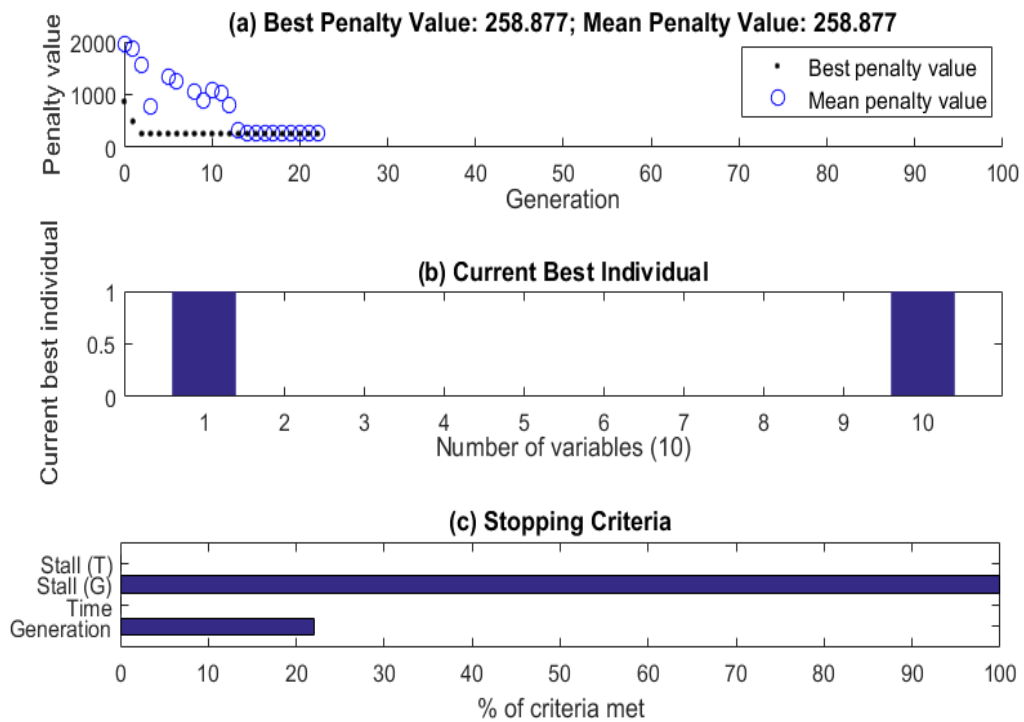

Figure 22. Three Sensors Scenario: (a) Best and mean fitness values at each generation, (b) Best Individuals (Optimal Sensor Locations), and (c) Stopping Criteria

Comparing the EnKF displacement and velocity estimates of the eighth floor with their respective synthetic real-measured data between the first generation guess [2 5 6] (Figure 23) and the final generation estimate [1 7 10] (Figure 24), an obvious improvement in the mismatch between the real synthetic values and their respective estimates calculated using the EnKF can be noticed when the sensors are placed at their optimal locations.
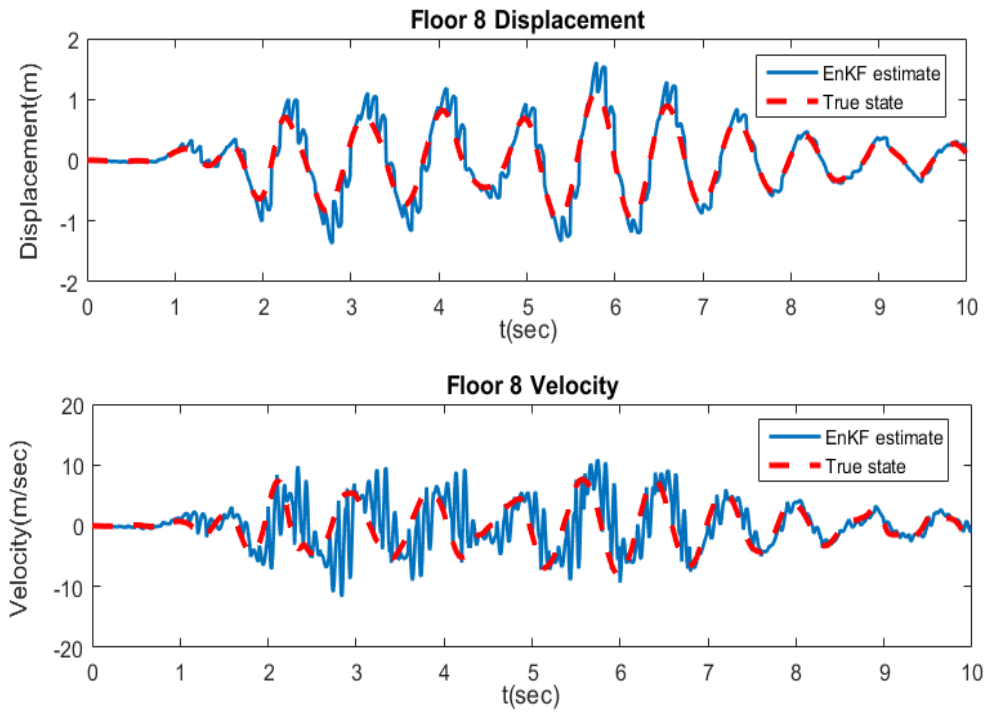
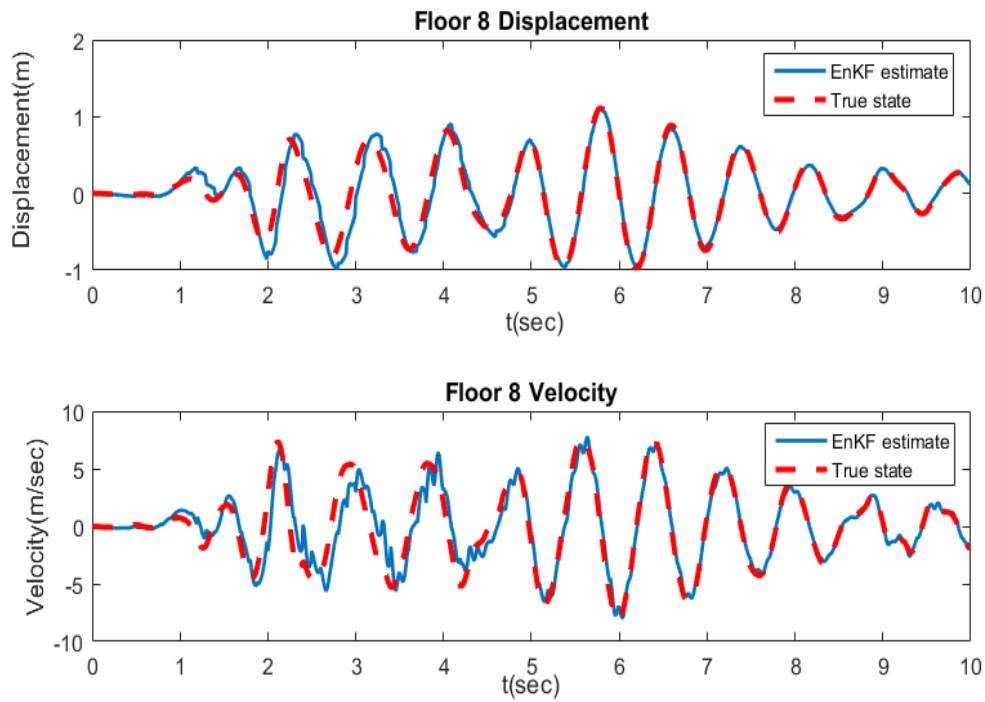**Figure 23.** Estimates of the eighth floor displacement and velocity at first generation: sensors placed at 2nd, 5th and 6th floors

Figure 24. Estimates of the eighth floor displacement and velocity at final generation: sensors placed 1st, 7th and 10th floors

5.4.3. Four sensors case

For the case of four available sensors, the total number of possible candidates of sensor places is 210 and the optimal locations found using the brute-force search were on floors 1, 2, 7 and 10. For the base case of 50 individuals per population, the GA-EnKF methodology converged 92% of the times to this optimal scenario when starting from different random initial positions at each simulation.

The results of the GA-EnKF framework randomly starting with an initial population of sensor locations corresponding to floor 1, 3, 4 and 6 are shown in figure 23. Same observations can be made concerning the best and mean values of the fitness function (Figure 25 part (a)) and the stopping criteria (Figure 25 part (c)) as the previous two cases. The best sensor locations are found to be on floors 1, 2, 7 and 10 (Figure 25 part (b)). This configuration of sensor placement is the same as the optimal one found using the exhaustive search technique.
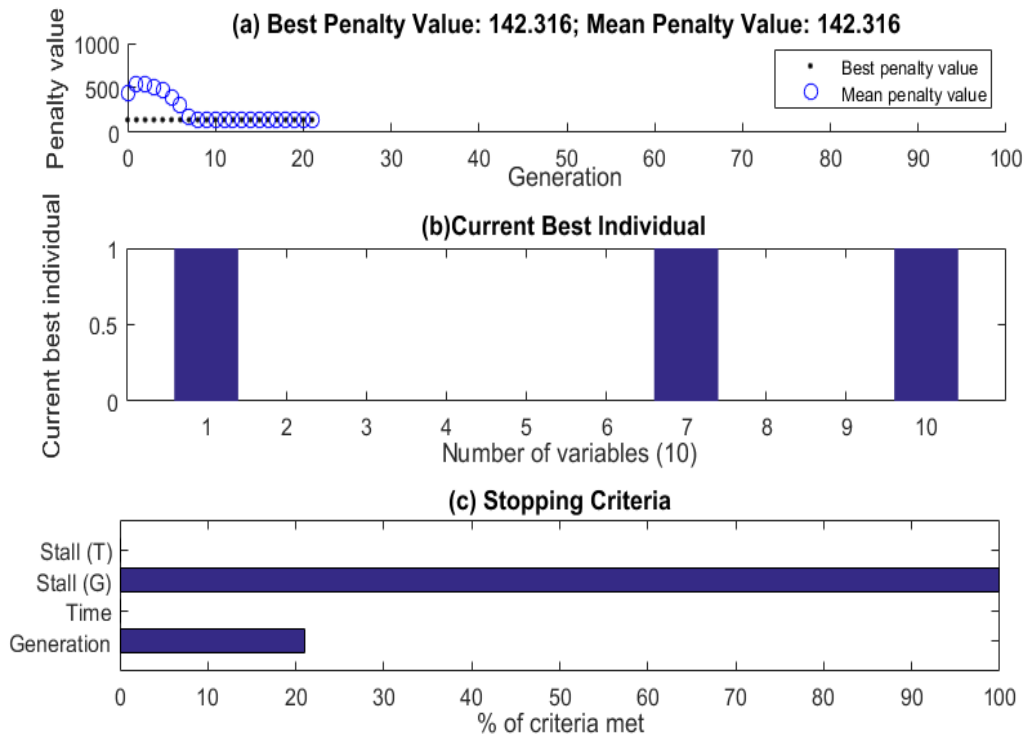
Figure 25. Four Sensors Scenario: (a) Best and mean fitness values at each generation, (b) Best Individuals (Optimal Sensor Locations), and (c) Stopping Criteria

An excellent improvement in the mismatch between the synthetically generated actual displacement and velocity of floor 8 and their respective predicted values calculated using the EnKF is shown when comparing the results of Figure 26 (first generation [1 3 4 6]) and Figure 27 (final generation [1 2 7 10]).
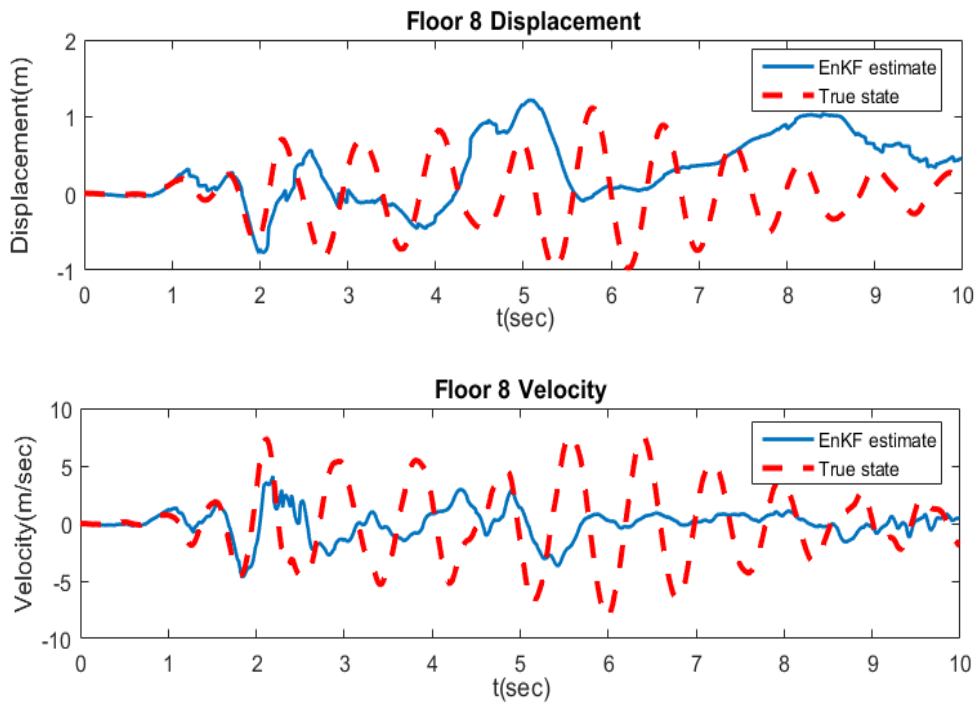
Figure 26. Estimates of the eighth floor displacement and velocity at first generation: sensors placed at 1st, 3rd, 4th and 6th floors
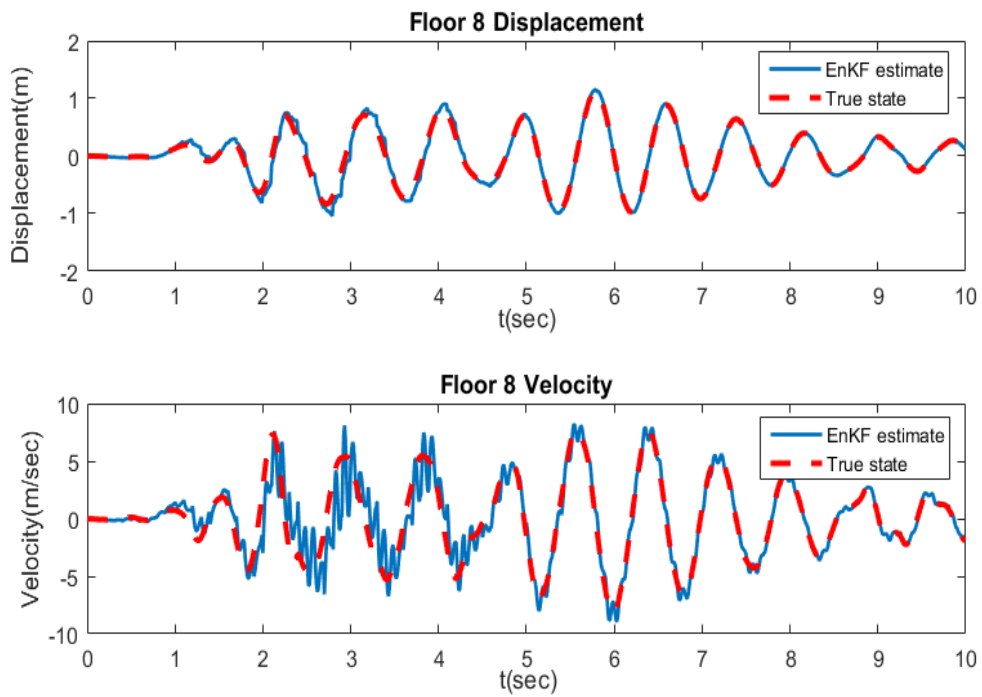

Figure 27. Estimates of the eighth floor displacement and velocity at final generation: sensors placed at 1st, 2nd, 7th and 10th floors

5.4.4. Five sensors case

Finally, for the case of five available sensors, a total of 252 combinations of sensor locations are possible and the best positions to place the sensors on are found to be the floors 1, 2, 5, 7 and 10, using the exhaustive search method. The proposed framework converged 82% of the times to the brute-force's optimal solution when starting from different initial random positions.

Figure 28 represents the GA-EnKF results, where the initial sensor locations are randomly chosen to be on floors 3, 5, 6, 9 and 10. The best sensor positions are found to be on floors 1, 2, 5, 7 and 10 (Figure 28 part (b)), which are similar to the findings of the brute-force search technique for the case of five available sensors. Since the number of available sensors is larger in this case, the best places of these monitoring devices become at the middle of the building in addition to the top and bottom locations.
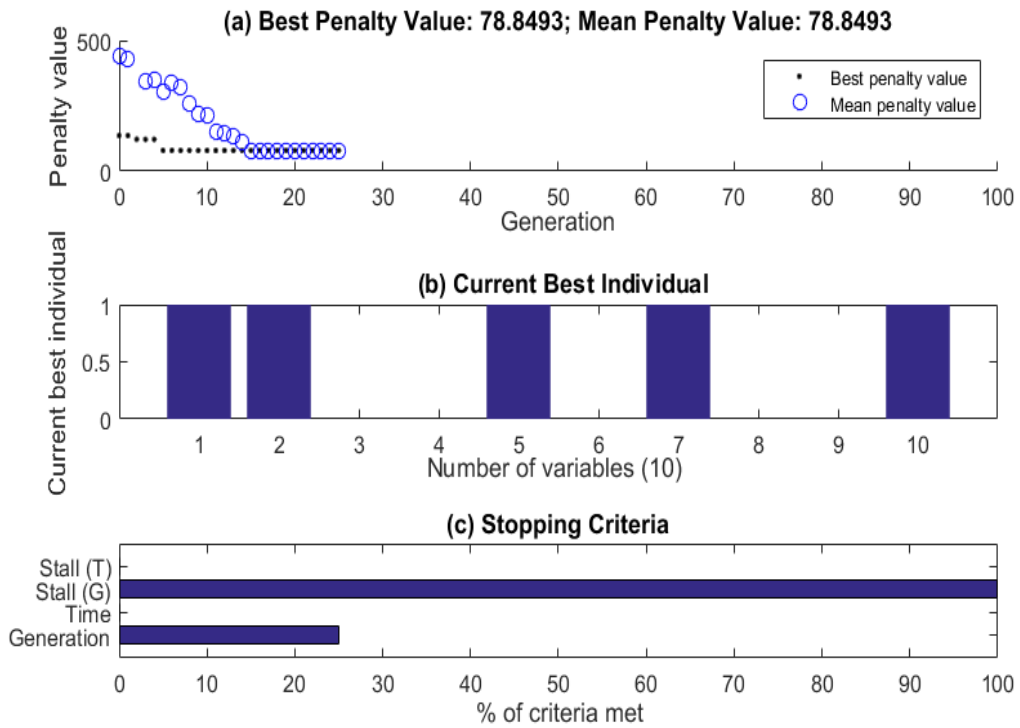
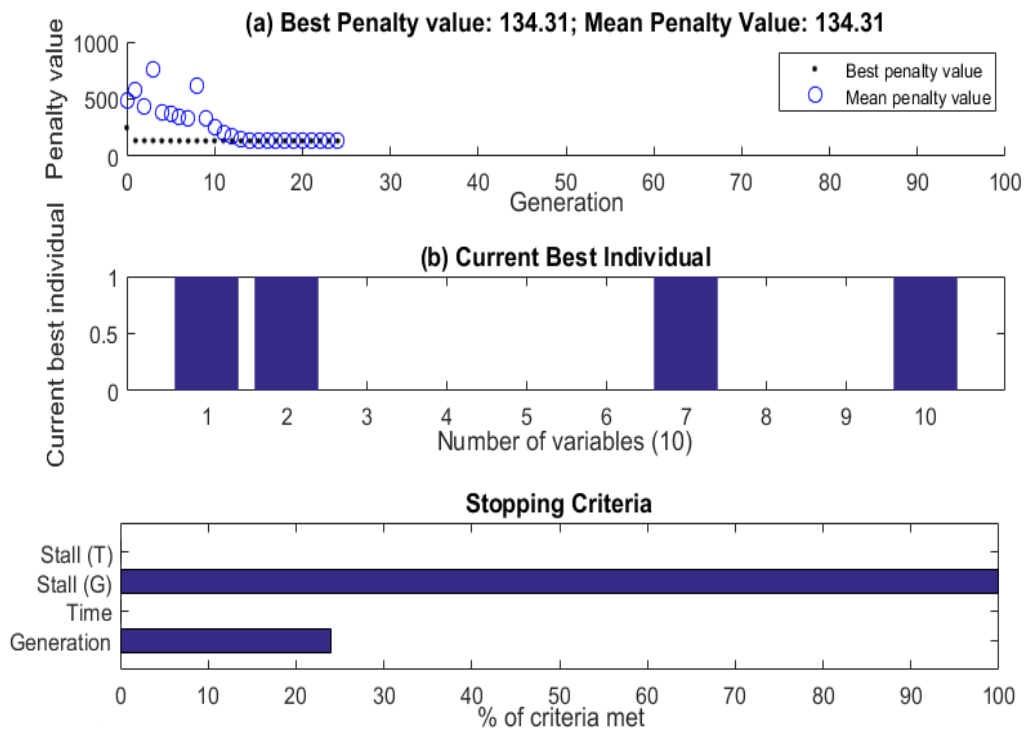**Figure 28.** Five Sensors Scenario: (a) Best and mean fitness values at each generation, (b) Best Individuals (Optimal Sensor Locations), and (c) Stopping Criteria

Figures 29 and 30 show a comparison between the mismatch between actual and predicted displacement and velocity of floor 8 when the sensors are placed at their random initial locations ([3 5 6 9 10]) and at their best locations ([1 2 5 7 10]), respectively.
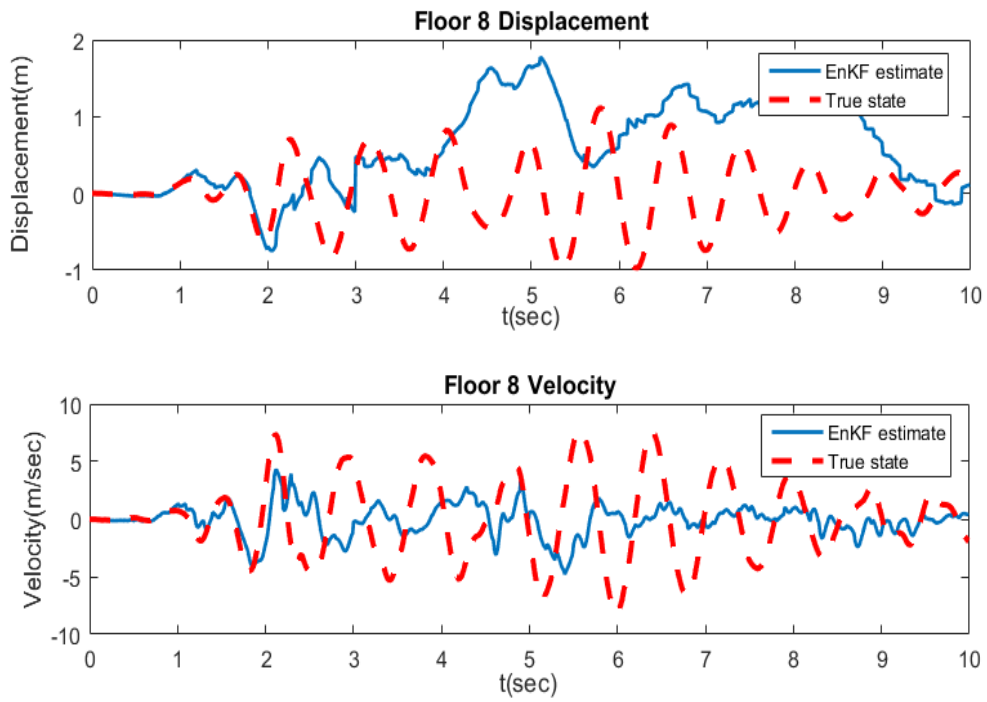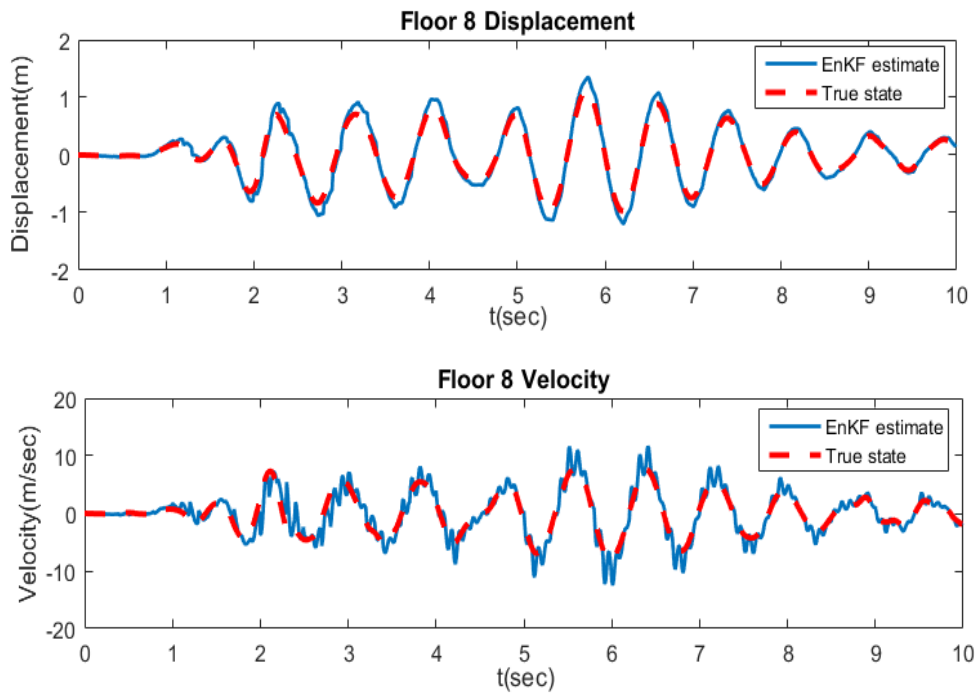
Figure 29. Estimates of the eighth floor displacement and velocity at first generation: sensors placed at 3rd, 5th, 6th, 9th and 10th floors
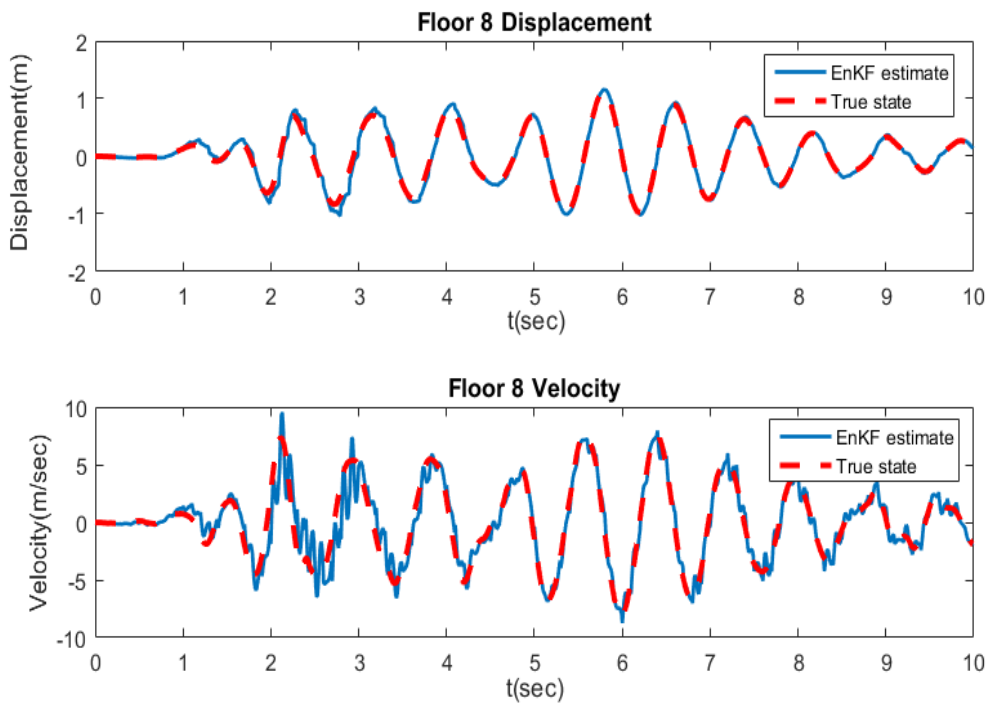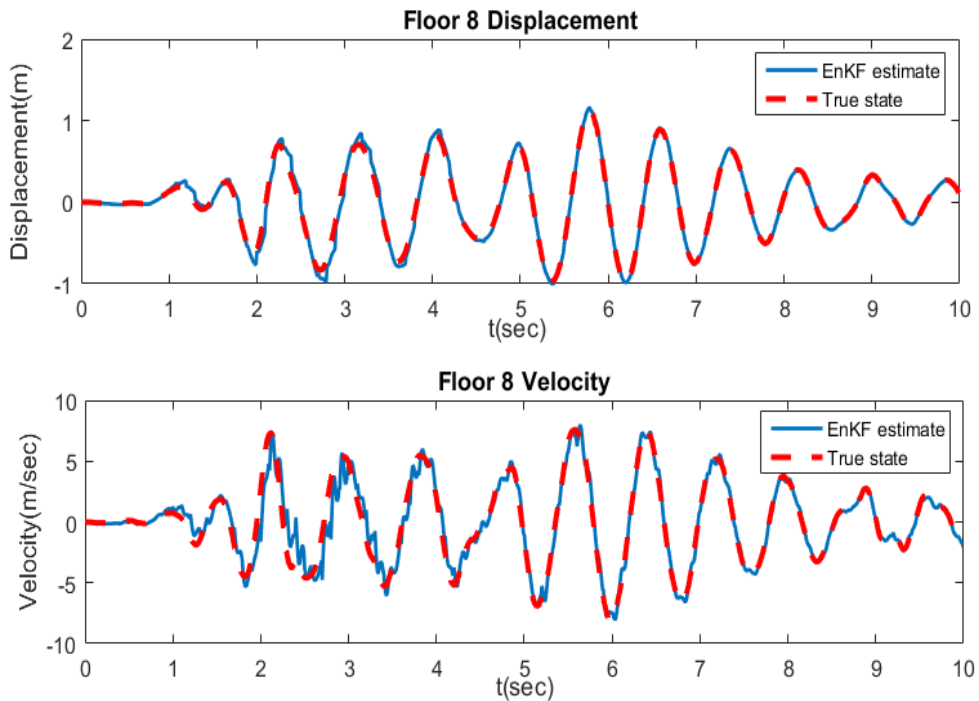

Figure 30. Estimates of the eighth floor displacement and velocity at final generation: sensors placed 1st, 2nd, 5th, 7th and 10th floors

86

Table 3 shows the percentage convergence of the GA-EnKF framework for the base case of 50 individuals per population for the different number of sensors used. Table 4 presents the sensitivity of this convergence rate to population size for the four different scenarios. The data in Table 4 is also graphically presented in Figure 31 showing the percentages of convergence of the proposed GA-EnKF framework to the brute force solutions versus the size of population used. It is noted that, the larger the population size is, the higher the percentage of convergence for the four cases of available sensors. A trade-off between the accuracy of the results and the computational cost should be performed to select the best size of population in each generation.

Table 3. Percentage of Convergence of GA-EnKF Results to Optimal Brute Force Result Starting from Different Initial Sensor Locations

Number of Sensors Used	Total Number of Sensor Location Combinations	Brute Force OSP Results	% of Convergence of GA-EnKF Results to Optimal Brute Force Result
2 Sensors	45	[1 10]	88%
3 Sensors	120	[1 7 10]	84%
4 Sensors	210	[1 2 7 10]	92%
5 Sensors	252	[1 2 5 7 10]	82%

Table 4. Percentage of Convergence of GA-EnKF Results to Optimal Brute Force
Result Using Different Population Sizes

Number of Sensors Used	% of Convergence of GA-EnKF Results to Optimal Brute Force Result				
	Population Size				
	10	30	50	70	100
2 Sensors	26 %	48%	88%	92%	94%
3 Sensors	24%	42%	84%	86%	94%
4 Sensors	22%	52%	92%	94%	98%
5 Sensors	20%	48%	82%	92%	96%
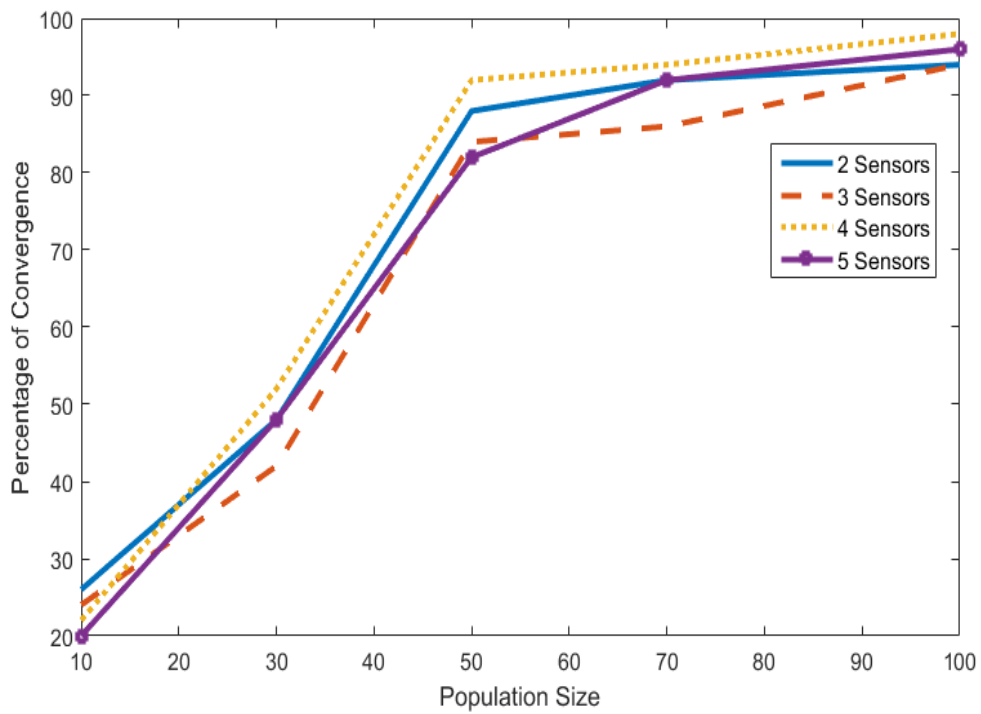
Figure 31. Percentage of Convergence of GA-EnKF method to Brute-Force's optimal
solutions versus population size

In Figure 32, the penalty values corresponding to the optimal sensor locations are plotted versus the available number of sensors. It can be seen that this best fitness value, which is nothing but the mismatch between predicted and measured data corresponding to the optimal sensor locations, decreases as the number of available sensors increases, to reach its lowest value when the sensors are placed on all the floors of the building under consideration. It should be noted that the logical optimum scenario is to place sensors everywhere on a structure (on each floor), resulting in a total of 10 sensors in this case, to get the most accurate results and the least mismatch between predicted and measured displacements and velocities. However as aforementioned, although this solution provides the most accurate results, it comes at a higher cost.
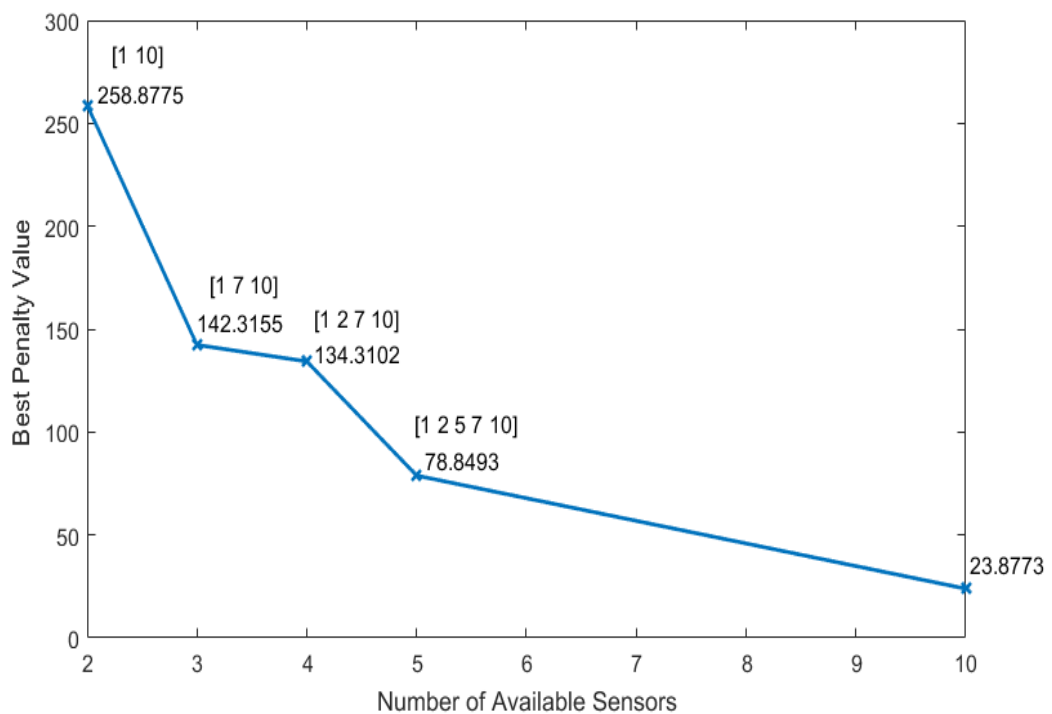
Figure 32. Best Penalty value versus number of available sensors

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

In this dissertation, the importance of uncertainty quantification is highlighted through a comparative study between intrusive and non-intrusive techniques used to quantify and represent the available uncertainties. For this reason, a comparison between two different variations of the Kalman filter technique, Ensemble Kalman Filter and Polynomial Chaos Kalman Filter, is performed. The comparison is based on the computational burden of the simulation runs required by each method to identify the system state and parameters and on the accuracy and performance of each filter in quantifying the uncertainty for SHM purposes. This is illustrated through a numerical example, consisting of a 4-degrees of freedom nonlinear system subjected to seismic excitation and suffering from hysteretic behaviors represented by the Bouc-Wen model. A pre-defined damage of the first degree of freedom is imposed ten seconds after the excitation hits the system.

A sensitivity analysis was performed on the ensemble size for the EnKF method and a relatively large ensemble size equals to 10,000 was selected to attain a sufficiently high accuracy in estimating the parameters and response of this complex and highly nonlinear system for damage detection purposes. On the other hand, an exhaustive analysis was performed on the dimensionality of the PCE used that is increased every time an independent source of error, due to model or measurement uncertainty, is incorporated in the system. For this reason, the PCE bases were limited to finite terms while maintaining a good approximate propagation of the covariance matrix, resulting in

a minimum dimension equals to 45 for the PCKF method. This minimum dimension of the PCE was used along with an order equals to 2 in the comparative analysis. While both variations of the Kalman filter method were able to locate the damage in space and time and to accurately approximate the system state and unknown parameters for SHM purposes, the PCKF method outperformed the ordinary EnKF approach in terms of computational effort.

As a conclusion, since the EnKF belongs to the class of non-intrusive methods that use black-box models to solve forward problems, it is easily implementable if compared to the intrusive methods such as the PCKF approach. On the other hand, for highly nonlinear and complex systems, the EnKF approach requires a relatively large ensemble size to attain a comparable high accuracy with the PCKF method and consequently a higher computational burden.

In addition, a novel optimal sensor placement methodology based on combining the genetic algorithm and the ensemble Kalman filter approaches is presented in this thesis. This combination improves the applicability of the method to a wider range of structural problems with hysteretic behaviors and with limited available real-time measured data.

The efficiency and robustness of the proposed method are illustrated through a numerical example consisting of a ten-story shear building subjected to seismic excitation. The GA is first applied to randomly generate an initial population of sensor locations from which the individuals with the best fitness functions are selected as the parents, then through a minimization procedure, the best locations of the sensors are determined. The fitness function in this dissertation is taken to be the difference

between the synthetically generated measurements of the displacements and velocities of the different floors of the building and their respective predicted data calculated using the EnKF method. It was shown in the presented numerical example that even for a small number of available sensors and consequently a limited amount of measured data, the EnKF was successfully able to approximate and update the system state and parameters. Subsequently, the GA was able to converge to the optimal locations of the sensors corresponding to the lowest penalty function.

The accuracy of the results obtained via the proposed methodology is assessed by comparing these results with the optimal ones identified using a brute-force search approach. This exhaustive search consists of systematically calculating the fitness value of each possible combination of sensor locations for the four different cases of available sensors, 2, 3, 4 and 5 sensors cases. A sensitivity analysis on the initial guesses of sensor locations is performed and the percentages of convergence of the proposed GA-EnKF methodology to the brute-force search's optimal results are calculated. High percentages of convergence are recorded, ascertaining the robustness and accuracy of the proposed framework for OSP. Furthermore, a sensitivity analysis is conducted to assess the dependence of the presented framework on the size of the population of the GA. Results show a systematic increase in the GA-EnKF convergence rate to the optimal brute force solution as the size of the population increases to reach a minimum of 94% convergence, for all cases studied, for a population of 100 individuals.

To summarize the analysis results, the best locations of the available sensors are identified as: (1) floors 1 and 10 for the two sensors case, (2) floors 1, 7 and 10 for the three sensors case, (3) floors 1, 2, 7 and 10 for the four sensors case, and (4) floors 1, 2, 5, 7 and 10 for the five sensors case. Consequently, for a limited number of

available sensors, the best sensor locations are at the lowest and highest floors of the building, and for a higher number of available sensors, the optimal locations are at the bottom, middle and top of the building to cover all the structure under consideration and provide the user with the most informative needed data.

It is proven in this dissertation that placing the sensors at their optimal locations reduces the mismatch between the synthetically measured displacements and velocities of the different floors of the building and their respective predicted values calculated using the EnKF.

Furthermore, it is shown that as the number of available sensors is increased, the value of the penalty function is lower, until reaching the lowest value when all the floors are monitored by sensors. Therefore, a trade-off between the accuracy needed and the cost and time should be taken into account to determine the number of sensors required.

Future work may consist of respectively acquiring the synthetic real data and the predicted data from two different mathematical models, instead of using the same model as in the case of Chapter 4 of this dissertation, where the Bouc-Wen model was used in the forward and inverse problems to get the synthetic real displacements and velocities of the different DOFs of the system under consideration and their respective predicted values; or as in the case of Chapter 5, where the non-parametric model was used to synthetically generate the measured displacements and velocities of the different floors of the 10-story building as well as their respective predicted values. A great care should be taken when selecting the appropriate error values coming from the parametric, measurement and model uncertainties.

On the other hand, future work may also include dealing with experimental real models to extract the true state of the system under consideration, instead of synthetically generating the true responses using approximate numerical models. In this case, the expected main challenge that will be faced lies in accurately predicting the errors coming from the initial guess uncertainty of the unknown parameters, the model error depending on how accurately can the mathematical model describe the true physics of the real system, and from the variability coming from the real collected measured data.

APPENDICES

APPENDIX 1

1.1. Ensemble Kalman Filter Codes for Parameters and State Characterization of

the 4-DOF system

```
clear all
close all
clc

load('un11');  % Measurements Perturbed
load('unn11'); % Measurements Not Perturbed


dt = 0.01;


ie  = 0.05;  % initial guess error
Me1 = 0.005; % model error for state
Me2 = 0.01;  % model error for parameters
mme = 0.03;  % measurement error


NumTimeSteps=2000;


% Number of DOF
ndof = 4;

% Mass and Stiffness
M = 2*eye(ndof,ndof);


k = 8.5;
c = 0.27;

K = zeros(ndof,ndof);
C = zeros(ndof,ndof);

B = [2;2;2;2];
gamma = [1;1;1;1];


N=10000;

for i = 1
    K(i,i)   = 2*k;
    K(i,i+1) = -k;
    C(i,i)   = 2*c;
    C(i,i+1) = -c;
end
for i = 2:(ndof-1)
    K(i,i-1) = -k;
    K(i,i) = 2*k;
```

```matlab
    K(i,i+1) = -k;
    C(i,i-1) = -c;
    C(i,i)   = 2*c;
    C(i,i+1) = -c;
end
for i = ndof
    K(i,i-1)  = -k;
    K(i,i) = k;
    C(i,i-1) = -c;
    C(i,i)   = c;
end


% Generalized Loading
read = fopen('El Centro 1940 data.txt', 'r');
ElCentro = zeros(NumTimeSteps*(dt/0.02),1);
P = zeros(NumTimeSteps*(dt/0.02),1);
for t = 0:dt:(NumTimeSteps*(dt/0.02)*dt)
    d = t/dt;
    j = int32(d) + 1;
    ElCentro(j) = fscanf(read,'%f',1);
    P(j) = ElCentro(j)*9.81*10;
end

% Linear Interpolation of Excitation
if (dt==0.02)
    Pinterp = P;
else
    n  = 0:0.02:(NumTimeSteps*(dt/0.02)*0.02);
    n2 = 0:dt:(NumTimeSteps*(dt/0.02)*0.02);
    Pinterp = interp1(n',P,n2','linear');
end

alpha = 0.15;

% Measurements from the Bouc-Wen Model

un1(:,1)=zeros(ndof,1);
udotn1(:,1)=zeros(ndof,1);
zn1(:,1)=zeros(ndof,1);
xall1=[zeros(3*ndof,1);B;gamma;k;c];

for i=1:NumTimeSteps-1
    tinit = (i-1)*dt;
    tf = (i)*dt;
[un1(:,i+1),udotn1(:,i+1),zn1(:,i+1)] =
PredictedData2Damage(un1(:,i),udotn1(:,i),zn1(:,i),xall1(:,i),Pinter
p,M,ndof,dt,alpha,tinit,tf,i);
xall1(:,i+1)=[un1(:,i+1);udotn1(:,i+1);zn1(:,i+1);B;gamma;k;c];
end

%Measurements Not Perturbed

unn11=[un1' udotn1']; % if not load
% [unn11]=unn11; % if load
```

```matlab
% Measurements perturbed

for nnn=1:NumTimeSteps-1
un1(:,nnn)=un1(:,nnn)+un1(:,nnn)*randn*mme;
udotn1(:,nnn)=udotn1(:,nnn)+udotn1(:,nnn)*randn*mme;
end

[un11] = [un1' udotn1'];% if not load
% [un11] = un11; % if load

% Check when we have measurements
TimeMeasurement=zeros(NumTimeSteps,2);

for i=1:NumTimeSteps-1
    TimeMeasurement(i,1)=(i-1)*dt;

    if (mod(i,20)==0) % Measurement at every 20 time steps
        TimeMeasurement(i,2)=1;
    end

end

% Ensemble Kalman Filtering
% ------------------------

%Ensemble Size

Au1 = zeros(5*ndof+2,N); % 22x10000
Af1 = zeros(5*ndof+2,N); % 22x10000


% Observation Matrix
% H: 8x22
H = zeros(2*ndof,5*ndof+2);

% All DOFs observed (u and udot)
for i = 1:1:2*ndof
    H(i,i) = 1;
end


% Initial Matrix

uint      = 0*ones(ndof,1);
udotint   = 0*ones(ndof,1);
zinit     = 0*ones(ndof,1);
Binit     = [2.5;2.5;2.5;2.5];
gammainit = [1.2;1.2;1.2;1.2];
Kinit     = 7;
Cinit     = 0.4;

x0 = [uint;udotint;zinit;Binit;gammainit;Kinit;Cinit];

counter1 = 1;
```

```matlab
% 1) Initial Guess Error

for j = 2:N
    randd=randn;
    for i=3*ndof+1:4*ndof
    Au1(i,1) = x0(i);

        Au1(i,j)=x0(i)+ie*x0(i)*randd;
    end
end

for j = 2:N
    randd=randn;
    for i=4*ndof+1:5*ndof
    Au1(i,1) = x0(i);

        Au1(i,j)=x0(i)+ie*x0(i)*randd;
    end
end

for j = 2:N
%     randd=randn;
    for i=5*ndof+1:5*ndof+2
    Au1(i,1) = x0(i);

        Au1(i,j)=x0(i)+ie*x0(i)*randn;
    end
end

Saved1{counter1, 1} = Au1;
counter1 = counter1 +1;


un2 = zeros(NumTimeSteps,ndof);
udotn2 = zeros(NumTimeSteps,ndof);
zn2 = zeros(NumTimeSteps,ndof);

tic;

% 2) Model Error & 3) Measurement Error (to u and udot only)

rr=1;
for io=1:NumTimeSteps-1
    io

    tinit = (io-1)*dt;
    tf = (io)*dt;
    t(io)=tf;
    for s = 1:1:N
        s;
```

```matlab
        [un2,udotn2,zn2] =
PredictedData2(Au1(1:1:ndof,s),Au1(ndof+1:1:2*ndof,s),Au1(2*ndof+1:3
*ndof,s),Au1(:,s),Pinterp,M,ndof,dt,alpha,tinit,tf,io);
        un2;
        udotn2;
        zn2;

        for j = 1:1:ndof
            Af1(j,s) = un2(j);
        end
        mnn = 1;
        for j = ndof+1:1:2*ndof
            Af1(j,s) = udotn2(mnn);
            mnn = mnn+1;
        end
        mn = 1;
        for j = 2*ndof+1:1:3*ndof
            Af1(j,s) = zn2(mn);
            mn = mn+1;
        end
    end
    Af1(3*ndof+1:5*ndof+2,:)=Au1(3*ndof+1:5*ndof+2,:);

if (mod(t(io),0.05)==0&&t(io)>0.45)
     for it=1:5*ndof+2
    meanxT(it,:)=mean(Af1(it,:));
     end

for j = 1:N
    randd=randn;
    for i=1:1*ndof

        Af1(i,j)=Af1(i,j)+Me1*meanxT(i,1)*randd;
    end
end

for j = 1:N
    randd=randn;
    for i=1*ndof+1:2*ndof

        Af1(i,j)=Af1(i,j)+Me1*meanxT(i,1)*randd;
    end
end

for j = 1:N
    randd=randn;
    for i=2*ndof+1:3*ndof

        Af1(i,j)=Af1(i,j)+Me1*meanxT(i,1)*randd;
    end
end

for j = 1:N
    randd=randn;
    for i=3*ndof+1:4*ndof

        Af1(i,j)=Af1(i,j)+Me2*meanxT(i,1)*randn;
```

```matlab
    end
end

for j = 1:N
    randd=randn;
    for i=4*ndof+1:5*ndof

        Af1(i,j)=Af1(i,j)+Me2*meanxT(i,1)*randn;
    end
end

for j = 1:N
%     randd=randn;
    for i=5*ndof+1:5*ndof+2
    Au1(i,1) = x0(i);

        Af1(i,j)=Af1(i,j)+Me2*meanxT(i,1)*randn;
    end
end
end

    for klm = 1:2*ndof
        umeasured (1,klm) = un11(counter1,klm);
    end

if((TimeMeasurement(io,2)==1)&&t(io)>0.5)

tin(rr)=t(io);
            [Au1, Gke, Pe, Re,e]=
EnkFCorrect2(Af1,un11(counter1,:),H,mme);
  rr=rr+1

    else
                Au1 = Af1;
    end

    Saved1{counter1,1}=Au1;
    counter1=counter1+1;
end

TimeofRun = toc

for i=1:NumTimeSteps

for j=1:5*ndof+2


    Au1=Saved1{i,1};

    vector(j,i)= mean(Au1(j,:));
    stdvect(j,i) = std(Au1(j,:));


end
end
```

1.2. Polynomial Chaos Kalman Filter Codes for Parameters and State

Characterization of the 4-DOF system

```
clear all
close all
clc

tic;

load('un11');  % Measurements Perturbed
load('unn11'); % Measurements Not Perturbed

store=0;
stop=0;

options.TolFun = 1e-9;
options.MaxFunEvals=2000000;
options.TolX=10e-9;

% Dimension 45
num1=3;
num2=1;

% % Dimension 58
% num1=4;
% num2=1;

% % Dimension 71
% num1=5;
% num2=1;

count1=1;
count2=1;

dt = 0.01;

% load('dim45order1') % 4 Floors; num1=3 and num2=1
load('dim45order2') % 4 Floors; num1=3 and num2=1

% load('dim58order1') % 4 Floors; num1=4 and num2=1
% load('dim58order2') % 4 Floors; num1=4 and num2=1

% load('dim71order1') % 4 Floors; num1=5 and num2=1
% load('dim71order2') % 4 Floors; num1=5 and num2=1

OrderU=2;
% OrderU=1;
% OrderU=3;

ie  = 0.05;  % initial guess error
Me1 = 0.005; % model error for state
Me2 = 0.01;  % model error for parameters
mme = 0.03;  % measurement error
```

```matlab
NumTimeSteps=2000;

% Number of DOF
ndof = 4;



% Mass and Stiffness
M = 2*eye(ndof,ndof);


k = 8.5;
c = 0.27;


K = zeros(ndof,ndof);
C = zeros(ndof,ndof);


B = 2*ones(ndof,1);
gamma = 1*ones(ndof,1);

for i = 1
    K(i,i)   = 2*k;
    K(i,i+1) = -k;
    C(i,i)   = 2*c;
    C(i,i+1) = -c;
end
for i = 2:(ndof-1)
    K(i,i-1) = -k;
    K(i,i) = 2*k;
    K(i,i+1) = -k;
    C(i,i-1) = -c;
    C(i,i)   = 2*c;
    C(i,i+1) = -c;
end
for i = ndof
    K(i,i-1)  = -k;
    K(i,i) = k;
    C(i,i-1) = -c;
    C(i,i)   = c;
end



% Generalized Loading
read = fopen('El Centro 1940 data.txt', 'r');
ElCentro = zeros(NumTimeSteps*(dt/0.02),1);
P = zeros(NumTimeSteps*(dt/0.02),1);
for t = 0:dt:(NumTimeSteps*(dt/0.02)*dt)
    d = t/dt;
    j = int32(d) + 1;
    ElCentro(j) = fscanf(read,'%f',1);
    P(j) = ElCentro(j)*9.81*10;
end

% Linear Interpolation of Excitation
if (dt==0.02)
    Pinterp = P;
else
```

```matlab
    n  = 0:0.02:(NumTimeSteps*(dt/0.02)*0.02);
    n2 = 0:dt:(NumTimeSteps*(dt/0.02)*0.02);
    Pinterp = interp1(n',P,n2','linear');
end


alpha = 0.15;

% Measurements from the Bouc-Wen Model

un1(:,1)=zeros(ndof,1);
udotn1(:,1)=zeros(ndof,1);
zn1(:,1)=zeros(ndof,1);
xall1=[zeros(3*ndof,1);B;gamma;k;c];
for i=1:NumTimeSteps-1
    tinit = (i-1)*dt;
    tf = (i)*dt;
[un1(:,i+1),udotn1(:,i+1),zn1(:,i+1)] =
PredictedData2Damage(un1(:,i),udotn1(:,i),zn1(:,i),xall1(:,i),Pinter
p,M,ndof,dt,alpha,tinit,tf,i);
xall1(:,i+1)=[un1(:,i+1);udotn1(:,i+1);zn1(:,i+1);B;gamma;k;c];
end

%Measurements Not Perturbed

% unn11=[un1' udotn1']; % if not load
[unn11]=unn11; % if load



% Measurements perturbed

for nnn=1:NumTimeSteps-1
un1(:,nnn)=un1(:,nnn)+un1(:,nnn)*randn*mme;
udotn1(:,nnn)=udotn1(:,nnn)+udotn1(:,nnn)*randn*mme;
end



% [un11] = [un1' udotn1']; % if not load
[un11] = un11; % if load



% Check when we have measurements
TimeMeasurement=zeros(NumTimeSteps,2);

for i=1:NumTimeSteps
    TimeMeasurement(i,1)=(i-1)*dt;

   if (mod(i,20)==0) % Measurement at every 20 time steps
       TimeMeasurement(i,2)=1;
   end

end

rr=1;%for tracking updates number
```

```matlab
% Polynomial Chaos Kalman Filtering
% -------------------------------

% Observation Matrix
% H: 8x22
H = zeros(2*ndof,5*ndof+2);

% All Floors Observed
for i = 1:1:2*ndof
    H(i,i) = 1;
end


uint      = 0*ones(ndof,1);
udotint   = 0*ones(ndof,1);
zinit     = 0*ones(ndof,1);
Binit     = [2.5;2.5;2.5;2.5];
gammainit = [1.2;1.2;1.2;1.2];
Kinit     = 7;
Cinit     = 0.4;


x0 = [uint;udotint;zinit;Binit;gammainit;Kinit;Cinit];

counter1 = 1;

% Initial Guess Error

locB=2;
locgamma=3;
lock=4;
locc=5;

locf=locc;

finitememory = 1;
DimU = 4+(5+2*ndof)*num1+2*num2;

stdB = ie*Binit;
stdgamma = ie*gammainit;
stdk1 = ie*Kinit;
stdc1 = ie*Cinit;


BasisU=Basis(DimU,OrderU);
HermitePolD= Hermite(OrderU);

% nsample=100;   % dim45order1: 46x45
nsample=1200; % 1100 or 1200 dim45order2: 1081x45

% nsample=60;    % dim58order1: 59x58
% nsample=1800; % 1800 or 2000 dim58order2: 1770x58

% nsample=80;    % dim71order1: 72x71
% nsample=2700;   % dim71order1: 2628x71
```

```
s=latin_hs(zeros(DimU,1),ones(DimU,1),nsample,DimU);
for i= 1:nsample
i
rndd=s(i,:)';
[basis(:,i)]=evalPC_N(BasisU,rndd(:,1)',HermitePolD);
end

coefB=cell([1,ndof]);
coefgamma=cell([1,ndof]);
coefk=cell([1,1]);
coefc=cell([1,1]);


coefk{1}=zeros(1,size(BasisU,1));
coefk{1}(1,1)=Kinit(1);
coefk{1}(1,lock)=stdk1(1);

coefc{1}=zeros(1,size(BasisU,1));
coefc{1}(1,1)=Cinit(1);
coefc{1}(1,locc)=stdc1(1);

for i=1:ndof
coefB{i}=zeros(1,size(BasisU,1));
coefB{i}(1,1)=Binit(i);
coefB{i}(1,locB)=stdB(i);

coefgamma{i}=zeros(1,size(BasisU,1));
coefgamma{i}(1,1)=gammainit(i);
coefgamma{i}(1,locgamma)=stdgamma(i);

U{i}(:,1)=zeros(size(BasisU,1),1);
Udot{i}(:,1)=zeros(size(BasisU,1),1);
Z{i}(:,1)=zeros(size(BasisU,1),1);
end

% If not load('dimord')
% To load dimension and order of PCE

% [Cat,List,L,F]= Out2(BasisU);
% [DD]= categories(BasisU,OrderU,DimU,Cat,List,L,F);



NormU=NormPC(BasisU,DimU);

for i=1:ndof
        Au1(i,:)=U{i}(:,1)';
        Au1(ndof+i,:)=Udot{i}(:,1)';
        Au1(2*ndof+i,:)=Z{i}(:,1)';
        Au1(3*ndof+i,:)=coefB{i};
        Au1(4*ndof+i,:)=coefgamma{i};

end
Au1(5*ndof+1,:)=coefk{1};
Au1(5*ndof+2,:)=coefc{1};
```

```matlab
Saved1{counter1, 1} = Au1;
counter1 = counter1 +1;



% 2) Model Error & 3) Measurement Error (to u and udot only)

for i=1:NumTimeSteps-1
    i


    tinit = (i-1)*dt;
    tf = (i)*dt;
    t(i)=tf;
    Af1=Au1;
    [un2,udotn2,zn2] =
PredictedData2PC(Au1(1:1:ndof,:),Au1(ndof+1:1:2*ndof,:),Au1(2*ndof+1
:3*ndof,:),Au1,Pinterp,M,ndof,dt,alpha,DD,NormU,tinit,tf,i,basis,Bas
isU,nsample);


        Af1(1:ndof,:) = un2;

        Af1(ndof+1:2*ndof,:) = udotn2;

        Af1(2*ndof+1:1:3*ndof,:) = zn2;



if (mod(t(i),0.05)==0&&t(i)>0.45)
   t(i)
if (stop<1)
if count1==num1
    count1=0;
    stop=1;
end
for nn=1:3
    KK{i}(nn,:)=find(BasisU(:,(locf-1+nn*num1-count1)));
    Af1(:,KK{i}(nn,:))=0;
    Af1((nn-1)*ndof+1:nn*ndof,locf+nn*num1-count1)=Me1*Af1((nn-
1)*ndof+1:nn*ndof,1);
end

for nn=4:3+2*ndof
    KK{i}(nn,:)=find(BasisU(:,(locf-1+nn*num1-count1)));
    Af1(:,KK{i}(nn,:))=0;
    Af1(3*ndof+nn-3,locf+nn*num1-count1)=Me2*Af1(3*ndof+nn-3,1);
end
cc=1;
for nn=4+2*ndof:5+2*ndof
    KK{i}(nn,:)=find(BasisU(:,(locf-1+nn*num1-count1)));
    Af1(:,KK{i}(nn,:))=0;
    Af1(5*ndof+cc,locf+nn*num1-count1)=Me2*Af1(5*ndof+cc,1);
    cc=cc+1;
end

count1=count1+1;
```

```matlab
else
A2=Af1(:,2:DimU+1).^2;
A3=sum(A2);
A3=sqrt(A3);
[A3d Indd]=sort(A3,'descend');
[A3a Inda]=sort(A3,'ascend');
Indd=Indd+1;
Inda=Inda+1;

    count1=0;
   for nn=1:5+2*ndof
   KK{i}(nn,:)=find(BasisU(:,(locf-1+nn*num1-count1)));
   remvdBas=Af1(:,KK{i}(nn,:));
   end

   fstInMax= max(List{2});



fstInR=[KK{i}(KK{i}<=fstInMax);((5+2*ndof)*(num1)+locf+1:(5+2*ndof)*
(num1)+locf+2*num2)'];%adding location of sensor error;

   B1=calCov(Af1(:,2:end),NormU(2:end));

   fstAll=2:fstInMax;

   fstStay=fstAll((ismember(fstAll,fstInR)==0));

   Ab=Af1;
   Z1=length(fstStay);


XX1=fsolve(@(X)PaDiff1(X,NormU,5*ndof+2,Z1,B1),Af1(:,Indd(1:length(f
stStay))),options);%only first order

   Asol1=reshape(XX1,5*ndof+2,Z1);

   Af1(:,2:end)=0;%only first order

   Af1(:,fstStay)=Asol1;

    for nn=1:5+2*ndof
    Af1(:,KK{i}(nn,:))=0;
    end
   Aa=Af1;
for nn=1:3
    Af1((nn-1)*ndof+1:nn*ndof,locf+nn*num1-count1)=Me1*Af1((nn-
1)*ndof+1:nn*ndof,1);
end

for nn=4:3+2*ndof
    Af1(3*ndof+nn-3,locf+nn*num1-count1)=Me2*Af1(3*ndof+nn-3,1);
end

cc=1;
for nn=4+2*ndof:5+2*ndof
    Af1(5*ndof+cc,locf+nn*num1-count1)=Me2*Af1(5*ndof+cc,1);
```

```matlab
        cc=cc+1;
end


end
end

  if((TimeMeasurement(i,2)==1)&&t(i)>0.5)
        counter1
  if count2==num2
      count2=0;
  end

  for nn=1:2
      KK{i}(nn,:)=find(BasisU(:,locf-1+(5+2*ndof)*(num1)+nn*num2-
count2));
      Afl(:,KK{i}(nn,:))=0;
end

        B=zeros(2*ndof,length(BasisU));
        B(:,1)=un11(counter1,:);

        B(1:ndof,locf+(5+2*ndof)*num1+num2-count2)=mme*B(1:ndof,1);
        B(ndof+1:2*ndof,locf+(5+2*ndof)*num1+2*num2-
count2)=mme*B(ndof+1:2*ndof,1);


        count2=count2+1;
      Pa=zeros(5*ndof+2,5*ndof+2);
      R=zeros(2*ndof,2*ndof);

      for ii=2:length(BasisU)
          Pa=Pa+Af1(:,ii)*Af1(:,ii)'*NormU(ii);
          R=R+B(:,ii)*B(:,ii)'*NormU(ii);
      end


      Au1=Af1+Pa*H'/(H*Pa*H'+R)*(B-H*Af1);

     tin(rr)=t(i);
      rr=rr+1;
  else
       Au1 = Af1;
  end


    for ii=1:ndof
        U{ii}(:,counter1) = Au1(ii,:)';
        Udot{ii}(:,counter1)=Au1(ndof+ii,:)';
        Z{ii}(:,counter1)=Au1(2*ndof+ii,:)';
        coefB{ii}(counter1,:)=Au1(3*ndof+ii,:);
        coefgamma{ii}(counter1,:)=Au1(4*ndof+ii,:);
    end
    coefk{1}(counter1,:)=Au1(5*ndof+1,:);
    coefc{1}(counter1,:)=Au1(5*ndof+2,:);

    Saved1{counter1,1}=Au1;
```

```matlab
    counter1=counter1+1;

end

stdvect = zeros(5*ndof+2,NumTimeSteps);

for i=1:NumTimeSteps

    Au1=Saved1{i,1};
    vector(:,i)= (Au1(:,1));

    stdvect(:,i) = (Au1(:,2:end).^2)*NormU(2:end)';

end
stdvect=sqrt(stdvect);
TimeofRun = toc
```

APPENDIX 2

2.1. Genetic Algorithm-Ensemble Kalman Filter Framework for Optimal Sensor

Placement of the 10-Story Building

```matlab
function [ x,output ] = GaRun()


% 10 Floors
% 5 sensors case
LowerBound = zeros(10,1);
UpperBound = ones(10,1);


A = [1 1 1 1 1 1 1 1 1 1; -1 -1 -1 -1 -1 -1 -1 -1 -1 -1];
b = [5; -5];

Options =
gaoptimset('PopulationSize',50,'EliteCount',2,'Generations',100,'Sta
llGenLimit',20,'TolFun',1e-10, 'Display','iter','PlotFcns',
{@gaplotbestf,@gaplotbestindiv,@gaplotstopping});

[x,fval,exitflag,output] =
ga(@myfunctiondt2,10,A,b,[],[],LowerBound,UpperBound,[],[1 2 3 4 5 6
7 8 9 10],Options);


end

```

```matlab
function [Fval] = myfunctiondt2(x)
x
x1=find(x==1)



% The input x is related to the observation matrix H. For example if
floors
% 1 and 4 are observed, H(1,1) = 1 & H(4,4) = 1 & H(5,5) = 1 &
H(8,8) = 1.
% So x is [1 4]

sense_vect=500;
tic;

for jjj=1:length(sense_vect)

zz=1;
dt = 0.01;
```

```
ie  = 0.1;      % initial guess error
me1 = 0.01;     % me = 0.01 model error for state
me2 = 0.005;    % me = 0.005 model error for parameters
mme = 0.005;    % measurement error

NumTimeSteps=500;

% Number of DOF
ndof = 10;

% Mass and Stiffness

M = eye(ndof,ndof);
k = eye(ndof,ndof);

for i = 1:3
    M(i,i) = 50000;
    k(i,i) = 40e6;
end
for i = 4:8
    M(i,i) = 40000;
    k(i,i) = 35e6;
end
for i = 9:10
    M(i,i) = 30000;
    k(i,i) = 20e6;
end

K = zeros(ndof,ndof);
for i = 1
    K(i,i) = k(i,i)+k(i+1,i+1);
    K(i,i+1) = -k(i+1,i+1);
end
for i = 2:(ndof-1)
    K(i,i-1) = -k(i,i);
    K(i,i) = k(i,i)+k(i+1,i+1);
    K(i,i+1) = -k(i+1,i+1);
end
for i = ndof
    K(i,i-1) = -k(i,i);
    K(i,i) = k(i,i);
end

C  = 19000;

% Generalized Loading
% El Centro Earthquake Excitation
read = fopen('El Centro 1940 data.txt', 'r');
ElCentro = zeros(NumTimeSteps,1);
P = zeros(NumTimeSteps,1);
for t = 0:dt:(NumTimeSteps*dt)
    d = t/dt;
    j = int32(d) + 1;
    ElCentro(j) = fscanf(read,'%f',1);
    P(j) = ElCentro(j)*9.81*10;
end
```

```matlab
% Linear Interpolation of Excitation
if (dt==0.02)
    Pinterp = P;
else
    n  = 0:0.02:(NumTimeSteps*0.02);
    n2 = 0:dt:(NumTimeSteps*0.02);
    Pinterp = interp1(n',P,n2','linear');
end



% Check when we have measurements
TimeMeasurement=zeros(2*NumTimeSteps,2);

for i=1:2*NumTimeSteps
    TimeMeasurement(i,1)=(i-1)*dt;

   if (mod(i,10)==0) % Measurement at every 10 time steps
        TimeMeasurement(i,2)=1;
   end

end

%Ensemble Size
N=400;
Au1 = zeros(6*ndof,N);

% Observation Matrix
% H
H = zeros(2*ndof,6*ndof);

% Floors Observed
H(x1(1),x1(1)) = 1;
H(x1(2),x1(2)) = 1;
H(x1(3),x1(3)) = 1;
H(x1(4),x1(4)) = 1;
H(x1(5),x1(5)) = 1;

for iii = 1:ndof
    if (H(iii,iii) == 1)
        H(iii+ndof,iii+ndof) = 1;
    end
end

% Initial Matrix
uint    = 0*ones(ndof,1);
udotint = 0*ones(ndof,1);

a = 1e7*ones(ndof,1);
b = 1e4*ones(ndof,1);
c = 4.7e4*ones(ndof,1);
d = 1.5e4*ones(ndof,1);

x0 = [uint;udotint;a;b;c;d];

counter1 = 1;
```

```matlab
% 1) Initial Guess Error
for i=2*ndof+1:6*ndof
    Au1(i,1) = x0(i);

    for j = 2:N
        seed = j^2;
        randn('seed',seed);
        Au1(i,j)=x0(i)+ie*x0(i)*randn(1);
    end
end
Saved1{counter1, 1} = Au1;
counter1 = counter1 +1;

% Measurements from non-parametric model
A1=[uint;udotint;1e8*ones(ndof,1);2e5*ones(ndof,1);9.4e5*ones(ndof,1
);4.5e5*ones(ndof,1)];

for s = 1:1:2*NumTimeSteps-1
        tinit = (s-1)*dt;
        tf = (s)*dt;


        [A1(1:ndof,s+1), A1(ndof+1:2*ndof,s+1)] =
rungekuttamult(@funudoubledotmult,tinit,A1(1:ndof,s),A1(ndof+1:2*ndo
f,s),A1(:,s),M,Pinterp,dt,ndof);
        A1(2*ndof+1:6*ndof,s+1)= A1(2*ndof+1:6*ndof,s);
end

for kk = 1:2*ndof
    A11measurement(kk,:) = A1 (kk,:);
End

% Measurements perturbed
randn('seed',10);
A1measurement = A11measurement +
mme.*A11measurement.*randn(2*ndof,2*NumTimeSteps);



% 2) Model Error & 3) Measurement Error (to u and udot only)

for i=1:2*NumTimeSteps-1

    tinit = (i-1)*dt;
    tf = (i)*dt;

    for s = 1:1:N

        [Af1(1:ndof,s) Af1(ndof+1:2*ndof,s)] =
rungekuttamult(@funudoubledotmult,tinit,Au1(1:ndof,s),Au1(ndof+1:2*n
dof,s),Au1(:,s),M,Pinterp,dt,ndof);
```

```matlab
Af1(1:2*ndof,s)=Af1(1:2*ndof,s)+me1*randn(2*ndof,1).*Af1(1:2*ndof,s)
;


Af1(2*ndof+1:6*ndof,s)=Au1(2*ndof+1:6*ndof,s)+me2*randn(4*ndof,1).*A
u1(2*ndof+1:6*ndof,s);

        end


if((TimeMeasurement(i,2)==1))
  [Au1 Gke Pe Re]= EnkFCorrect2(Af1,A1measurement(:,counter1),H,mme);

    else
        Au1 = Af1;
    end


    for j = 2*ndof+1:3*ndof
        for k = 1:N
            if (Au1(j,k)<500)
                seed = seed +1;
                randn('seed',seed);
                Au1(j,k) = 500 + abs(10*randn);
            end
        end
    end

    for j = 3*ndof+1:6*ndof
        for k = 1:N
            if (Au1(j,k)<1)
                seed = seed +1;
                randn('seed',seed);
                Au1(j,k) = 1 + abs(randn);
            end
        end
    end


    Saved1{counter1,1}=Au1;
    counter1=counter1+1;

end

for j=1:6*ndof
for i=1:2*NumTimeSteps

    Au1=Saved1{i,1};
    vector(j,i)= mean(Au1(j,:));
    stdvect(j,i) = std(Au1(j,:));

end
end

end
```

```matlab
for i = 1:2*ndof
    vector2(i,:) = vector(i,:);
end
Fval = norm((vector2-A1measurement)/A1measurement)

ST = fclose('all');


end
```

```matlab
function udoubledotmult=funudoubledotmult(t,u,udot,vect,M,P,dt,ndof)

u=[u;0];
udot=[udot;0];
a=[vect(2*ndof+1:3*ndof);0];
b=[vect(3*ndof+1:4*ndof);0];
c=[vect(4*ndof+1:5*ndof);0];
d=[vect(5*ndof+1:6*ndof);0];

cc = fix(t/dt);

%calcultig F;
if(ndof==1)
F(1)=a(1)*u(1)+b(1)*u(1)^3+c(1)*udot(1)+d(1)*u(1)*udot(1);
udoubledotmult(1,1)=(-F(1)-M(1)*P(cc+1))/M(1,1);
else
    for n=1:ndof
            if (n==1)
            F(n)=a(n)*(u(n)-0)+a(n+1)*(u(n)-u(n+1))+b(n)*(u(n)-
0)^3+b(n+1)*(u(n)-u(n+1))^3+c(n)*(udot(n)-0)+c(n+1)*(udot(n)-
udot(n+1))+d(n)*(u(n)-0)*(udot(n)-0)+d(n+1)*(u(n)-u(n+1))*(udot(n)-
udot(n+1));
            else
            F(n)=a(n)*(u(n)-u(n-1))+a(n+1)*(u(n)-u(n+1))+b(n)*(u(n)-
u(n-1))^3+b(n+1)*(u(n)-u(n+1))^3+c(n)*(udot(n)-udot(n-
1))+c(n+1)*(udot(n)-udot(n+1))+d(n)*(u(n)-u(n-1))*(udot(n)-udot(n-
1))+d(n+1)*(u(n)-u(n+1))*(udot(n)-udot(n+1));
            end
        udoubledotmult(n,1)=(-F(n)-M(n,n)*P(cc+1))/M(n,n);
    end

end
```

116

```matlab
function [Anew, Gke, Pe, Re, D]=EnkFCorrect2(A,d,H,z)


[dummy, N]=size(A);
ONEN=(1/N).*ones(N,1);
n=length(d');

for k=1:N;
Abar(:,k)=A*ONEN;
End

Apurt=A-Abar;

Pe=(Apurt*(Apurt)')./(N-1);

for i=1:N
    for j=1:n

        e(j,i)=d(j)*z.*randn;
        D(j,i)=d(j)+e(j,i);
    end
end

Re=(e*e')./(N-1);

Gke=(Pe*H')/((H*Pe*H'+Re));

Anew=A+Gke*(D-H*A);

end
```

117

BIBLIOGRAPHY

Al-Hussein, A., & Haldar, A. (2016). Unscented Kalman filter with unknown input and weighted global iteration for health assessment of large structural systems. *Structural Control and Health Monitoring, 23*(1), 156-175.

Arulampalam, S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing, 50*(2), 174-188.

Azam , S., Chatzi, E., Papadimitriou, C., & Smyth, A. (2015a). Experimental validation of the Kalman-type filters for online and real-time state and input estimation. *Journal of Vibration and Control*, 1077546315617672.

Azam, S., Chatzi, E., & Papadimitriou, C. (2015b). A dual Kalman filter approach for state estimation via outpu-only acceleartion measurements. *Mechanical Systems and Signal Processing, 60*, 866-886.

Azam, S., Chatzi, E., Papadimitriou, C., & Smyth, A. (2015c). *Experimental Validation of the Dual Kalman Filter for Online and Real-Time State and Input Estimation* (Vol. Volume 3). In Model Validation and Uncertainty Quantification. Springer International Publishing.

Beck, J. (2010). Bayesian system identification based on probability logic. *Structural Control and Health Monitoring, 17*(7), 825-847.

Begambre, O., & Laier, J. (2009). A hybrid particle swarm optimization-simplex algorithm (PSOP) for structural damage identification. *Advances in Engineering Software, 9*, 883-891.

Bertino, L., Evensen, G., & Wackernagel, H. (2003). Sequential data assimilation techniques in oceanography. *International Statistical review, 71*(2), 223-241.

Black, K. (2004). *Business Statistics for Contemporary Decision Making.* India: Wiley. ISBN 978-81-265-0809-9.

Blanchard, E., Sandu, A., & Sandu, C. (2010). A Polynomial chaos-based Kalman filter approach for parameter estimation of mechanical systems. *ASME J. Dyn. Syst. Mea. Control 132, 18, Special Issue on Physical Modeling*.

Blanchard, E., Sandu, A., & Sandu, C. (2010). Polynomial chaos-based parameter estimation methods applied to vehicle system. *Proceedings of the Institution of mechanical Engineers, Part K: Journal of Multi-body Dynamics, 224*, 59-81.

Blanchard, E., Sandu, C., & Sandu, A. (2007). A polynomial chaos based Bayesian approach for estimating uncertain parameters of mechanical systems. *ASME 2007 International Design Engineeirng Technical Conferences and Computers and Information in Engineering Conference*, (pp. 1041-1048).

Burgers, G., Leeuwen, P., & Evensen, G. (1998). Analysis scheme in the ensemble Kalman filter. *Monthly Weather Review, 126*(6), 1719-1724.

Cerny, V. (1985). Thermodynamical approach to the traveling salesman problem; An efficient simulation algorithm. *Journal of Optimization Theory and Applications, 45*(1), 41-51.

Chang, P., Faltau, A., & Liu, S. (2003). Review Paper: health monitoring of civil infrastructure. *Structural Health Monitoring, 2*(2), 257-267.

Chase, J., Bigoc, V., & Barroso, L. (2005). Efficient structural health monitoring for a benchmark structure using adaptive RLS filter. *Computers and Structures, 83*(8), 639-647.

Chatzi, E., & Smyth, A. (2009). The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing. *Structural Control and Health Monitoring, 16 (1)*, 99-123.

Chatzi, E., & Smyth, A. (2013). Particle filter scheme with mutation for the estimation of time-invariant parameters in structural health monitoring applications. *Structural Control and Health Monitoring, 20 (7)*, 1081-1095.

Ching , J., & Beck, J. (2004). Bayesian analysis of the phase II IASC-ASCE structural health monitoring experimental benchmark data. *Journal of Engineering Mechanics, 130*(10), 1233-1244.

Chiu, P. L., & Lin, F. (2004). A simulated annealing algorithm to support the sensor placement for target loaction. *Canadian Conference on Electrical and Computer Engineering. 2*, pp. 867-870. IEEE.

Chou, J. H., & Ghabboussi, J. (2001). genetic algorithm in structural damage detection. *Computers and Structures, 79*(14), 1335-1353.

Chou, J., & Ghabboussi, J. (2001). genetic algorithm in structural damage detection. *Computers and Structures, 79*(14), 1335-1353.

Chow, H., Lam, H. F., Yin, T., & Au, S. K. (2011). Optimal sensor configuration of a typical transmission tower for the purpose of structural model updating. *Structural Control and Health Monitoring, 18*(3), 305-320.

Chow, H., Lam, H., Yin, T., & Au, S. (2011). Optimal sensor configuration of a typical transmission tower for the purpose of structural model updating. *Structural Control and Health Monitoring, 18*(3), 305-320.

Chowdhary, G., & Jategaonkar, R. (2010). Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter. *Aerospace Science and Technology, 14 (2)*, 106-117.

Chowdhary, G., & Jategaonkar, R. (2010). Aerodynamic parameter estimation from flight data applying extended Kalman filter and unscented Kalman filter. *Aerospace Science and Technology, 14*(2), 106-117.

Christian, R., & Casella, G. (2013). Monte Carlo statistical methods. *Springer Science & Business Media*.

Chui, C., & Chen, C. (1991). *Kalman filtering with real time applications.* New York: Springer, 2nd edition.

Chui, C., & Chen, G. (1991). *Kalman Filtering with Real Time Applications* (second ed.). Springer.

Corigliano, A., & Mariani, S. (2004). Parameter identification in explicit structural dynamics: performance of the ectended Kalman filter. *Computer Methods in Applied Mechanics and Engineering, 193*(36), 3807-3835.

Cunha, A., Nasser, R., Sampaio, R., Lopes, H., & Breitman, K. (2014). Uncertainty quantification through the Monte Carlo method in a cloud computing setting. *Computer Physics Communications, 185*(5), 1355-363.

Daley, R. (1997). Atmospheric data assimilation. *Journal of Meteorological Society of Japan Series 2, 75*, 319-329.

Der Kiureghiana, A., & Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter? *Structural Safety, 31*(2), 105-112.

Ding, F., Liu, X., & Liu, G. (2011). Identification methods for Hammerstein nonlinear systems. *Digital Signal Processing, 21*(2), 215-238.

Doebling, S., Farrar, C., Prime, M., & Shevitz, D. (1996). *Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review.* Los Alamos National Lab., NM (United States).

Ebrahimian, H., Astroza, R., & Conte, J. (2015). Extended Kalman filter for material parameter estimation in nonlinear structural finite element models using direct differentiation method. *Earthquake Engineering & Structural Dynamics, 44*(10), 1495-1522.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo to forecast error statistics. *Journal of Geophysical Research: Oceans (1978-2012), 99*(C5), 10143-10162.

Evensen, G. (2003). The ensemble Kalman filer: Theoretical formulation and practical implementation. *Ocean Dynamics, 53*(4), 343-367.

Evensen, G. (2003). The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics, 53(4)*, 343-367.

Evensen, G. (2009, June). The ensemble Kalman filter for combined state and parameter estimation. *IEEE Control Systems Magazine*, 83-104.

Explorable.com. (2009, October 4). Retrieved from System Sampling: https://explorable.com/systematic-sampling

Explorable.com. (2009, June 2). Retrieved from Stratified Sampling Method: https://explorable.com/stratified-sampling

Explorable.com. (2009, October 18). Retrieved from Cluster Sampling: https://explorable.com/cluster-sampling

Farrar, C., & Worden, K. (2007). An introduction to Structural Health Monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 365*(1851), 303-315.

Figueiredo, E., Park, G., Farrar, C., Worden , K., & Figueiras, J. (2011). Machine learning algorithms for damage detection under operational and environmental variability. *Structural Health Monitoring, 10(6)*, 559-572.

Flynn, E. B., & Todd, M. D. (2010). A Bayesian approach to optimal sensor placement for structural health monitoring with application to active sensing. *Mechanical Systems and Signal Processing, 24*(4), 891-903.

Franco, G., Betti, R., & Lus, S. (2004). Identification of structural systems using an evolutionary strategy. *Journal of Engineering Mechanics, 130*(10), 1125-1139.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica, 7*, 1317-1339.

Ghanem, R. (1999). Ingredients for a general purpose stochastic finite element formulation. *Computer Methods in Applied Mechanics and Engineering, 168*(1), 19-34.

Ghanem, R., & Ferro, G. (2006). Health monitoring of strongly nonlinear systems using the ensemble Kalman filter. *Structural Control and Health Monitoring, 13*(1), 245-259.

Ghanem, R., & Shinozuka, M. (1995). Structural systems identification I: Theory. *Journal of Engineering Mechanics, 121*(2), 255-264.

Ghanem, R., & Spanos, P. (2003). *Stochastic Finite Elements: A Spectral Approach.* Courrier Corporation.

Ghanem, R., Masri, S., Pellissetti, M., & Wolfe, R. (2005). Identification and prediction of stochastic dynamical systems in a polynomial chaos basis. *Computer Methods in Applied Mechanics and Engineering, 194*(12), 1641-1654.

Ghosh, S., Roy, D., & Manohar, C. (2007). New formd of extended Kalman filter via transversal linearization and applications to structural system identification. *Conputer Methods in Applied Mechanics and Engineering, 196*(49), 5063-5083.

Gillijns, S., Mendoza, O., Chandrasekar, J., De Moor, B., Bernstein, D., & Ridley, A. (2006). What is the ensemble Kalman filter and how well does it work? *Proc., American Control Conference* (pp. pp. 6-pp). IEEE.

Gillijns, S., Mendoza, O., Chandrasekar, J., De Moor, B., Bernstein, D., & Ridley, A. (2006). What is the ensemble Kalman filter and how well does it work? *American Control Conference.* Minneapolis.

Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Conputers and Operations Research, 13*(5), 533-549.

Glover, F. (1989). Tabu search: Part I. *ORSA J Computing, 1*(3), 190-206.

Glover, F. (1990). Tabu search: Part II. *ORSA J Computing, 2*(1), 4-32.

Goyal, D., & Pabla, B. (2015). The vibration monitoring methods and signal processing techniques for structural health monitoring: a review. *Archives of Computational Methods in Engineering*, 1-10.

Grewal, M., & Andrews, A. (2008). *Kalman Filtering Theory and Applications using MATLAB* (third edition ed.). John Wiley & Sons.

Guo, H. Y., Zhang, L., Zhang, L., & Zhou, J. (2004). Optimal placement of sensors for structural health monitoring using improved genetic algorithms. *Smart Materials and Structures, 13*(3), 528.

Helton, J. (2000). Sampling-based methods for uncertainty and sensitivity analysis. *Multimedia Environmental Models, 32*(2), 73.

Helton, J., & Davis, F. (2002). Illustration of sampling-based methods for uncertainty and sensitivity analysis. *Risk Analysis, 22*(3), 591-622.

Helton, J., & Oberkampf, W. (2004). Alternative representations of epistemic uncertainty. *Reliability Engineering & System Safety, 85*(1), 1-10.

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* Oxford, England: U Michigan Press.

Hommels, A., Murakami, A., & Nishimura, S.-I. (2009). A comparison of the ensemble Kalman filter with the Unscented Kalman filter: application to the construction of a raod embankment. *Geotechniek, 13(1)*, 52.

Hoshiya, M., & Saito, E. (1984). Structural identification be extended Klaman filter. *Journal of Engineering Mechanics, 110*(12), 1757-1770.

Iman, R., Davenport, J., & Zeigler, D. (1980). *Latin Hypercube sampling (Program user's guide).* Sansia Labs., Albuquerque, NM (USA).

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering, 82*(1), 35-45.

Kalnay, E. (2003). *Atmospheric Modeling, Data Assimilation and Predictability.* Cambridge University Press.

Kamdepu, R., Foss , B., & Imsland, L. (2008). Applying the unscented Kalman filter for nonlinear state estimation. *Journal of Process Control, 18*(7), 753-768.

Kandepu, R., Foss, B., & Imsland, L. (2008). Applying the unscented Kalman filter for nonlinear state estimation. *Journal of Process Control, 18 (7)*, 753-768.

Kennedy, M., & O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistival Society, 63*(3), 425-464.

Kerschen, G., Worden, K., Vakakis, A., & Golinval, J. (2006). Past, present and future of nonlinear system identification in structural dynamics. *Mechanical Systems and Signal Processing, 20*(3), 505-592.

Keshavarzzadeh, V., Ghanem, R., Masri, S., & Aldraihem, O. (2014). Convergence acceleration of polynomial chaos solutions via sequence transformation. *Computer Methods in Applied Meechanics and Engineering, 271*, 167-184.

Kirkpatrick, S., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*(4598), 671-680.

Lam , H., Yang, J., & Hu, Q. (2011). how to install sensors for structural model updating. *Procedia Engineering, 14*, 450-459.

Lam, H. F., Yang, J. H., & Hu, Q. (2011). how to install sensors for structural model updating. *Procedia Engineering, 14*, 450-459.

Laviola, J. (2003). A comparison of unscented and extended Kalman filtering for estimating quaternion motion. *American Control Conference. 3.* IEEE.

LeDimet, F., & Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus A, 38*(2), 97-110.

Li, H., Qin, S., Tsotsis, T., & Sahimi, M. (2012). Computer simulation of gas generation and transport in landfills: VI-Dynamics updating of the model using the ensemble Kalman filter. *Chemical Engineering Science, 74*, 69-78.

126

Li, J., & Xiu, D. (2009). A generazlized polynomial chaos based ensemble Kalman filter with high accuracy. *Journal of Computational Physics, 228*(15), 5454-5469.

Liberatore, S., Speyer, J., & Hsu, A. (2006). Application of a fault detection filter to structural health monitoring. *Automatica 42, 42*(7), 1199-1209.

Lin, G., Engel, D., & Esliner, P. (2012). *Survey and avaluate uncertainty quantification methodologies.* PNNL-20914. Richland.

Ljung, L. (1979). Asymptotic behavir of the extended Kalman filter as a parameter estimator for linear systems. *IEEE Transactions on Automatic Control, 24*(1), 36-50.

Lyons , J., & Nasrabadi, H. (2013). Well placement optimization under time-dependent uncertainty using an ensemble Kalman filter and a genetic algorithm. *Journal of petrouleum Science and Engineering, 109*, 70-79.

Masri, S., Caffrey, J., Caughey, T., Smyth, S., & Chassiakos, A. (2004). Identification of the state eqaution in complex nonliear systems. *International Journal of Nonlinear Mechanics, 39*(7), 1111-1127.

Masri, S., Ghanem, R., Arrate, F., & Caffrey, J. (2009). A data based procedure for analyzing the response of uncertain nonlinear systems. *Structural Control and Health Monitoring, 16*, 724-750.

Mathworks. (n.d.). Retrieved from http://www.mathworks.com/help/gads/genetic-algorithm-options.html/.

McKay, M., Beckman, R., & Conover, W. (1979). A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer

Code. *Technometrics. American Statistical Association, 21*(2), 239-245. doi:10.2307/1268522

Metropolis, N., & Ulam, S. (1949). The Monte Carlo method. *Journal of the American Association, 44*(247), 335-341.

Nasr, D., & Saad, G. (2015). Sensor placement optimization using Ensemble Kalman Filter and Genetic Algorithm. *5th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDyn 2015)*. Crete, Greece.

Nasrellah, H., & Manohar, C. (2011). Finite element method based Monte Carlo filters for structural system identification. *Probabilistic Engineering Mechanics, 26*(2), 294-307.

Navon, I., Zou, X., Derber, J., & Sela, J. (1992). Variational data assimilation with an adiabatic version of the NMC spectral model. *American Meterological Society, 120*(7), 1433-1446.

Neal, R. (2001). Annealed importance sampling. *Statistics and Computing, 11*(2), 125-139.

Nechak, L., Berger, S., & Aubry, E. (2011). A polynomial chaos approach to the robust analysis of the dynamic behavior of friction systems. *European Journal of Mechanics A/ Solids, 271*, 594-607.

Papadimitriou , C. (2004). Optimal sensor placement methodology for parametric identification of structural systems. *Journal of Sound and Vibration, 278*(4), 923-947.

Papadimitriou, C., Beck, J. L., & Au, S. K. (2000). Entropy-based optimal sensor location for structural model updating. *Journal of Vibration and Control, 6*(5), 781-800.

Rao, A. R., & Anandakumar, G. (2007). Optimal placement of sensors for structural system identification and health monitoring using a hybrid swarm intelligence technique. *Smart Materials and Structures, 6*, 2658.

Ribeiro, M. (2004). Kalman and extended Kalman filters: Concept, derivation and properties. *Institute for Systems and Robotics, 43*.

Roth, M., Fritsche, C., Hendeby, G., & Gustafison, F. (2015). The ensemble Kalman filter and its relations to other nonlinear filters. *Signal Processing Conference (EUSIPCO)* (pp. 1236-1240). 23rd European: IEEE.

Saad, G., & Ghanem, R. (2009). Characterization of reservoir simulation models using a polynomial chaod-based ensemble Kalman filter. *Water Resources Research, 45*(4).

Saad, G., & Ghanem, R. (2011). Robust Structural Health Monitoring using a polynomial chaos sequential data assimilation technique. *COMPDYN 2011*. Corfu, Greece.

Saad, G., Ghanem, R., & Masri, S. (2007). Robust system identification of strongly non-linear dynamics using a polynomial chaos based sequential data assimilation technique. *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, *6*, pp. 6005-6013. Honolulu.

Sajeeb, R., Manohar, C., & Roy, D. (2009). A conditionally linearized Monte Carlo filter in non-linear structural dynamics. *International Journal of Nonlinear Mechanics, 44*(7), 776-790.

Salehi, H., Das, S., Chakrabartty, S., Biswas, S., & Burgueno, R. (2015). Structural assessment and damage identification algorithms using binary data. *ASME 2015 Conference on Smart Materials, Adaptive Structures* (pp. V002T05A011-V002T05A011). Colorado Springs, Colorado, USA: American Society of Mechanical Engineers.

Seyedpoor, S. (2012). A two stage method for structural damage detection using a modal strain energy based index and particle swarm optimization. *International Journal of Non-Linear Mechanics, 1*, 1-8.

Shahrokh Esfahani, M., & Dougherty, E. R. (2014). Effect of seperate sampling on classification accuracy. *Bioinformatics, 30*(2), 242-250.

Shen, Y., Golnaraghi, M., & Heppler, G. (2005). Analytical and experimental study of the response of a suspension system with a magnetorheological damper. *Journal of Intelligent Material Systems and Structures, 16*(2), 135-147.

Silva, M., Santos, A., Figueiredo, E., Santos, R., Sales, C., & Costa, J. (2016). A novel unsupervised approach based on genetic algorithm for structural damage detection in bridges. *Engineering Applications of Artificial Intelligence, 52*, 168-180.

Slika, W., & Saad, G. (2016). A practical Polynomial Chaos Kalman Filter Implementation Using Non-Linear Error Projection on a Reduced Polynomial Chaos Expansion. *International Journal for Numerical Methods in Engineering*.

Slika, W., & Saad, G. (2016). A robust polynomial chaos Kalman filter framework for corrosion detection in reinforced concrete structures. *VII European Congress on Computational Methods in Applied Sciences and Engineering.* Crete Island, Greece.

Slika, W., & Saad, G. (2016). An ensemble Kalman filter approach for service life

    prediction of reinforced concrete structures subject to chloride-induced

    corrosion. *Construction and Building Materials, 115*, 132-142.

Sohn, H., Farrar, C., Hemez, F., & Czarnechi, J. (2002). A review of Structural Health

    Monitoring litearture 1996-2001. (LA-UR-02-2095).

Spiridonakos, M., Chatzi, E., & Sudret, B. (2016). Polynomial Chaos Expansion

    Models for the Monitoring of Structures under Operational Variability. *ASCE-*

    ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil

    Engineering, B4016003.

Staszewski, W. J., & Worden, K. (2001). Overview of optimal sensor location methods

    for damage detection. *SPIE's 8th Annual International Symposium on Smart*

    Structures and Materials.

St-Pierre, M., & Gingras, D. (2004). Comparison between the unscented Kalman filter

    and the extended Kalman filter for the position estimation module of an

    integrated navigation information system. *IEEE Intelligent Vehicules*

    Symposium.

St-Pierre, M., & Gingras, D. (2004). Comparison between the unscented Kalman filter

    and the extended Kalman filter for the position estimation of an integrated

    navigation information system. *IEEE Intelligent Vehicules Symposium.*

Sudret, B., & Der Kiureghian, A. (2000). *Stochastic finite element methods and*

    reliability: A state-of-the-art report. University of California, Berkeley -

    Department of Civil & Environmental Engineering.

Sudret, B., & Der Kiureghian, A. (n.d.). *Stochastic finite element methods and*

    reliability: A state-of-the-art report.

Sudret, B., Berveiller, M., & Lemaire, M. (2006). A stochastic finite element procedure for moment and reliability analysis. *European Journal of Computational Mechanics/Revenue Europeenne de Mechanique Numerique, 15*(7-8), 825-866.

Vanik, M., Beck, J., & Au, S. (2000). Bayesian probabilistic approach to structural health monitoring. *Journal of Engineering Mechanics, 126*(7), 738-745.

Veach, E., & Guibas, L. (1995). Optimally combining sampling techniques for Monte Carlo rendering. *22nd Annual Conference on Computer Graphics and Interactive Techniques* (pp. 419-428). New York, NY, USA: ACM. doi:10.1145/218380.218498

Welch, G., & Bishop, G. (2006). *An introduction to the Kalman Filter.* North Carolina, US: University of North Carolina: Chapel Hill.

Wiener, N. (1938). The Homogeneous Chaos. *American Journal of Mathematics*, 897-936.

Yang, J., Lin, S., Huang, H., & Zhou, L. (2006). An adaptive extended Kalman filter for structural damage identification. *Structural Control and Health Monitoring, 13*(4), 849-867.

Yang, J., Pan, S., & Huang, H. (2007). An adaptive extended Kalman filter for structural damage identification II: Unknown inputs. *Structural Control and Health Monitoring, 14*(3), 497-521.

Yao, L., Sethares, W. A., & Kammer, D. (1993). Sensor placement for on-orbit modal identification via a genetic algorithm. *AIAA Journal, 31*(10), 1922-1928.

Yao, L., Sethares, W. A., & Kammer, D. (1993). Sensor placement for on-orbit modal identification via a genetic algorithm. *AIAA Journal, 31*(10), 1922-1928.

Yates, D., Moore, D., & Starnes, D. (2008). *The Practice of Statistics, 3rd Ed.* Freeman. ISBN 978-0-7167-7309-2.

Yi, T. H., & Li, H. N. (2012). Methodology development in sensor placement for health monitoring of civil infrastructures. *International Journal of Distributed Sensor Networks*.

Yi, T., & Li, H. (2012). Methodology developments in sensor placement for health monitoring of civil infrastructures. *International Journal of Distributed Sesnor Networks*.

Yuen, K. V., Katafygiotis, L. S., Papadimitriou, C., & Mickleborough, N. C. (2001). Optimal sensor placement methodology for identification with unmeasured excitation. *Journal of Dynamics Systems, Measurement, and Control, 123*(4), 677-686.

Yuen, K., Au, S., & Beck, J. (2004). Two-stage structural health monitoring approach for phase I benchmark studies. *Journal of Engineering Mechanics, 130*(1), 16-33.

Zhang, F., Xiong, H., Shi, W., & Ou, X. (2016). Structural health monitoring of Shanghai Tower during different stages using a Bayesian approach. *Structural Control and Health Monitoring*.

Zhang, H., Foliente, G., Yang, Y., & Ma, F. (2001). Parameter identification of inelastic structures under dynamic loads. *Earthquake Engineering and Structural Dynamics, 31*(5), 1113-1130.

Zhou, L., Wu, S., & Yang, J. (2008). Experimental study of an adaptive extended Kalman filter for structural damage identification. *Journal of Infrastructure Systems, 14*(1), 42-51.

Zhou, Q., Zhou, H., Zhou, Q., Yang, F., Luo, L., & Li, T. (2015). Structual damage

    detection based on posteriori probability support vector machine and Dempster-

    Shafer evidence theory. *Applied Soft Computing, 36*, 368-374.