FINITE ELEMENT MODEL UPDATING

Linda Simo Mthembu

A thesis submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfillment of the requirements for the degree of Doctor of Philosophy

Johannesburg, 2012

Declaration

I declare that this thesis is my own unaided work, except where otherwise acknowledged. It is being submitted for the degree of Doctor of Philosophy to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other University.

Signed this ____ day of _____2012

_____

Linda Simo Mthembu

Abstract

This thesis focuses on engineering, specifically structural, systems that are approximated by finite element models (FEMs). Initial FEMs are found to have poor accuracies and improved or updated models are sort. From the literature we note that a common set of challenges still persists in all FEM work. These are; which aspects of the model are most uncertain, how can we efficiently update the model and finally how do we know that our chosen model is the best for the system at hand. This is the finite element model updating problem.

These challenges are reinforced by the number of different FEMs that can be proposed for any one system and the difficulty of determining the best model from these. Moreover all the said challenges are applicable to all possible FEMs. To address these challenges we propose that the FEM updating problem be analyzed in a multi-model context. What is implicit in this proposal is that updating one model in isolation will not be very informative. This proposed context requires that all proposed methods in this thesis be general enough to be applicable to any set of FEMs.

To address the challenge of identifying the most uncertain parameters of a FEM, we propose using an evolution based procedure; population based incremental learning (PBIL). The main assumption for this method is that a list of uncertain model parameters can be represented as a vector. PBIL then probabilistically selects and updates, from this vector, the most uncertain parameters. To verify the consistency of this PBIL method, it is tested on two different objective functions and under two different measurement datasets.

The second challenge of finding an efficient way to update a FEM is also addressed via an evolution based procedure. In the proposed multi-model framework, efficiently means updating models quickly and without bias. We thus propose the updating of multiple FEMs using particle swarm optimization (PSO). This approach allows all models to be simultaneously updated and evaluated under one scheme. The result is the interaction of models as they are updated and

an accuracy ordering of these. Simulations of a real beam are carried out on a number of models and two objective functions.

To determine whether our chosen model is the best in the multi-model setting we propose using the Bayesian model evidence statistic. The model evidence is calculated using the Nested sampling algorithm. Jeffrey's scale is used to evaluate the significance of model evidence differences. Simulations on two real systems, using multiple models for each, are performed. The proposed method concisely shows and justifies the model ordering.

Acknowledgements

I would like to thank my parents; Lindiwe and Shole Mthembu for their unwavering support, guidance and giving me a better background than they had. I would also like to thank my sister, Sindi and brother Menzi for their support and companionship. I would like to thank Namwawa Siwale for the encouragement, patience and support throughout this thesis. I would also like to thank Professor Tshilidzi Marwala for his encouragement and supervision during this work. In addition I thank Professors Michael Friswell and Sondipon Adhikari for their insights and recommendations during this thesis.

Thank you!

Table of Contents

List of Figures

List of Tables

List of Symbols

θ	Model Parameters
D	Real measured system data
H	Finite element model (Mathematical model)
Z	Evidence
N_S	Number of samples
I	Number of iterations
L	Likelihood probability
π	Prior Probability
c	Damping matrix
k	Stiffness matrix
m	Mass matrix
$\ddot{x}(t)$	Node acceleration
$\dot{x}(t)$	Node velocity
$x(t)$	Node displacement
w	Natural frequency vector
ϕ	Mode shape vector
μ	Mean

Nomenclature

AIC	**A**kaike **I**nformation **C**riterion
APSF	**A**verage **P**ercentage **S**election **F**requency
BIC	**B**ayesian **I**nformation **C**riterion
GA	**G**enetic **A**lgorithm
GM	**G**eometric **M**easurements
I-Gain	**I**nformation **G**ain
FEM	**F**inite **E**lement **M**odel
FEMUP	**F**inite **E**lement **M**odel **U**pdating **P**roblem
NS	**N**ested **S**ampling
NF	**N**atural **F**requency
PBIL	**P**opulation **B**ased **I**ncremental **L**earning.
PSO	**P**article **S**warm **O**ptimization.
PNFD	**P**ercentage **N**atural **F**requency **D**ifference
SSE	**S**um of **S**quared **E**rrors
SMP	**S**tandard **M**aterial **P**roperties
d	FEM dimension (Number of parameters)
$\text{fem}_{\text{results}}$	FEM analytical results
m_{id}	i^{th} EM position at parameter d
M_{max}, M_{min}	maximum and minimum model position respectively
msrd_{data}	measured structural data
p_{id}	best position for the i^{th} FEM at parameter d
p_{gd}	Global best FEM at parameter d
v_{id}	i^{th} FEM velocity at parameter d
V_{max}, V_{min}	maximum and minimum model velocity respectively
w_{k}	k^{th} dimension inertia weight

Chapter 1

Introduction to Finite Element Model Updating

1. Introduction

This chapter describes the model updating problem faced by every initial model of a sufficiently complex system and the challenges posed by the problem. It also describes the systematic approach followed in addressing each challenge. This thesis does not assume the reader is familiar with the field of finite element models. For such a reader – the words 'finite element model'- can be replaced by the words 'mathematical model' without any loss of specificity. Science and specifically engineering is concerned with the understanding of natural phenomenon and or systems. To understand any system the general scientific process is: measure[1] the natural system, devise a mathematical model to describe and explain the behavior of the said system and finally check that the model is a good approximation of this system.

To this end finite element models (FEMs) are a particular class of models used to mathematically describe the mechanics and dynamics of complex engineering systems. These systems may be for example; biological (e.g. the mechanics of knee joint movement) or electrical (e.g. magnetic field in a motor) or mechanical (the stresses on a vehicle chassis). Besides mathematically describing such complex systems, modeling is also used to predict and analyze unforeseen

[1] The word measure is used generically but it is clear that some phenomenon cannot be physically measured. In such cases some aspects of the phenomenon of interest can be gathered to form data.

behaviour of these real systems in a variety of settings. System modeling forms an important stage of many engineering design problems. The results from the model either confirm or highlight limitations of the design. An analyst is usually interested in the accuracy, confidence range and more critically the correctness of the assumed mathematical model. In this thesis the model domain is structural finite element models (FEMs).

1.1 Background

In finite element modeling, dynamic structures are analyzed by discretizing the structure into constituent elements. When assembled these elements constitute a system described by a second-order matrix differential [1,2,3] equation of the form:

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{f}(t) \tag{1.1}$$

where **M**, **C** and **K** are of equal size and are the mass, damping and stiffness matrices, or alternatively the system matrices, $\mathbf{f}(t)$ is the input force, $x(t), \dot{x}(t)$ and $\ddot{x}(t)$ are the response vector displacement, velocity and acceleration respectively. If equation (1.1) is transformed to the modal domain [1, 2] and the structure considered is lightly damped or undamped, $\mathbf{C} = \mathbf{0}$, the corresponding eigenvalue equation for the *j*th mode becomes:

$$\left[-\left(\omega_j^m\right)^2 \mathbf{M} + \mathbf{K} \right] \phi_j^m = \mathbf{0} \tag{1.2}$$

where ω_j^m and ϕ_j^m are the *j*th measured (superscript m) system natural frequency and its corresponding mode shapes, together known as modal properties. If the natural frequency and mode shape are replaced by analytical values (superscript a) then the eigenvalue equation is not exactly satisfied, but becomes:

$$\left[-\left(\omega_j^a\right)^2 \mathbf{M}+\mathbf{K}\right]\phi_j^a = \varepsilon_j \tag{1.3}$$

where ε_j is the residual vector for the *j*th analytical frequency, ω_j^a, and the corresponding mode shape ϕ_j^a. In the above setting, given a set of measured system modal data, D, the finite element model problem is to determine the model that realistically approximates the mass and stiffness matrices while producing modal data as close to the measured modal data as possible (so called data-match). If these predictions are not sufficiently accurate, the residual vector will be non-zero and some aspects of the model will need to be modified. This exercise is made more difficult by the fact that a multitude of mathematical models of the structure, with varying levels of complexity, can be developed, leading to non-unique solutions for any particular system. Two main directions of research have been established in the area of finite element model updating; direct and indirect (iterative) methods.

In the direct model updating paradigm [2,4,5] the model modal parameters are directly equated to the measured modal data. Model updating is then characterized by the direct updating of the system matrices' elements. This effectively constrains the modal properties and frees the system matrices for updating. This approach often results in unrealistic elements in the system matrices e.g. large and physically impossible mass elements. In the indirect or iterative model updating approach [2,3,6,7] the updating problem is formulated as a 'relaxed' optimization problem with no hard constraints on the analytical modal outputs. This means physical system parameter values (e.g. structural thickness, mass, etc) are iteratively updated and the resultant analytical modal outputs are optimized to minimize their difference from the measured data**.** The iterative updating scheme has been applied within the classical eigenvalue sensitivity analysis [1,2,8,9,10,11 12,13,14] model regularization [15,16,17,18] maximum likelihood methods [2,4,5,9,19] evolution based optimization [20,21,22,23,24] and more recently, Bayesian approaches [3,7,25,26,27].

1.2 The Finite Element Model Updating Problem (FEMUP)

Finite element models are limited by definition; they are an approximation of a real system and will thus never produce dynamic results that are equal to the measured system's data. The question is then what can be done to the initial model for it to better reflect the real system's dynamic results? This leads to the need for automatic and intelligent methodology to improve models. This has to be attained whilst using realistic characteristic parameters of the system in question.

It is clear from the problem background that there are many ways to tackle this FEMUP. To see what the problem is fundamentally about and to provide a global perspective we have plotted Figure 1.1. This figure shows three spaces, S^1, S^2 and S^3, each inscribed by an oval. The S^1 space is the space of all possible measurements of the real system. S^2 is the space of all possible initial models and S^3 is the space of all possible updated models. The real system of interest exists outside the space S^1. This is due, amongst other reasons, to the measured data being incomplete from the impracticality of capturing the full dynamics of the system at every degree over the full behaviour range. This fact is shown by the different accuracy measurement datasets possible for a particular real system. In the case shown dataset 3 is the most accurate measurement of the system because the distance[2] between the two is smallest.

The figure also shows three hypothetical models of a particular measured system; M_A ($\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5$), M_B ($\beta_1, \beta_2, \beta_3$) and M_C ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$). The figure shows their initial positions prior to updating and their potential positions after updating. The location of the ideal optimal model is also shown in S^3. This particular 'optimal model' is the closest any model can be to the real system. Practically we do not work with the real system per se but consider the measured data to sufficiently represent the real system.

[2] Distance is a generic word for the difference magnitude between two points. This will be appropriately defined later in the thesis.

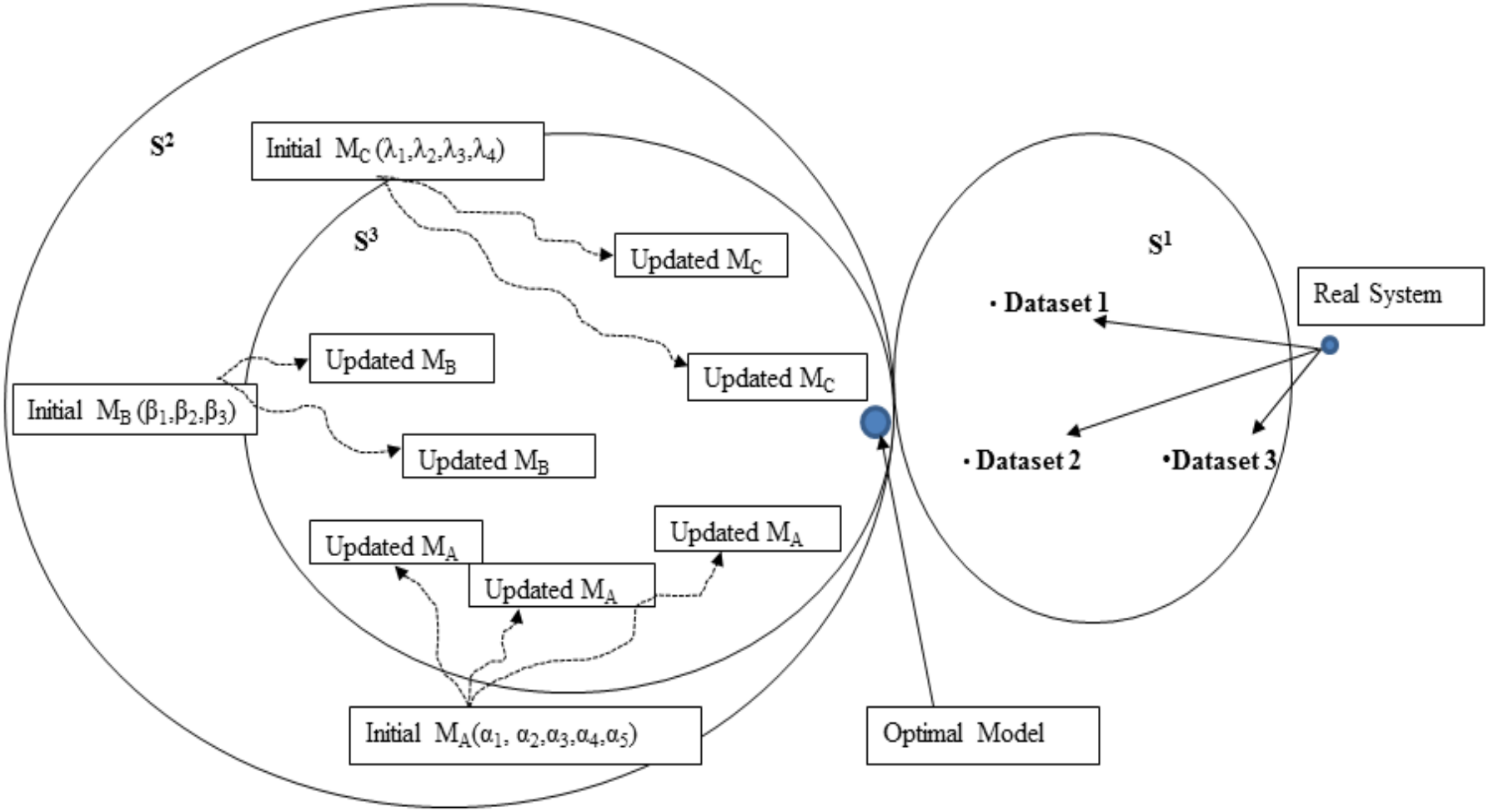


Figure 1.1 Finite element model updating scheme

1.3The need for FEM updating

Assume our model to be M_C and measured dataset to be dataset 2. As depicted in Figure 1.1 the initial model M_C is initially 'far' from the measured dataset. The main reason for creating the model is for it to be as accurate in approximating the measured data as possible. The goal therefore is to reduce the distance between this initial model and the measured dataset. For a given model reducing this distance is achieved by improving or updating the initial model.

1.3.1 The challenges

In practice engineers often propose a number of models for one real system for example the three models; M_A (α_1, α_2, α_3, α_4, α_5), M_B (β_1, β_2, β_3) and M_C (λ_1, λ_2, λ_3, λ_4) proposed in Figure 1.1. Each model has its own type and number of uncertain parameters. Before performing model updating, a number of fundamental questions need to be addressed, these are:

- **Which aspects of the model need to be updated?**

  Said differently, which features/parameters of the initial model are uncertain or incorrectly modelled? The above question effectively cast model updating as a system identification problem [29,30,31,32,33,34]. System-identification is concerned with the development of parameterized mathematical approximations to some complex system whose features or order is unknown and is to be identified. This simply translates to which and how many model parameters are sufficient to correctly capture the dynamics of the underlying process/system. In the FE models of concern there are a variety of structural parameters that can be mathematically modeled [35,36,37]. These can vary from geometric to material properties. Some of the most difficult and uncertain geometric parameters to model are points of structural variation such as joints and/or welds. Material uncertainties arise from incorrectly proposed properties due to lack of manufacturing and/or operational condition knowledge.

Given that a model can have a large number of uncertain parameters we do not want to blindly search for the parameter combination that produces a good solution. This combination parameter search can easily become a combinatorial problem. A significant constraint to such a procedure is that the updated parameters must remain physically realistic. That is when updated the values of the parameters must be within practical limits.

The first challenge therefore is to automatically identify and or select the most uncertain parameters in our proposed FE model. The next question is;

- **How can the chosen model be efficiently updated?**

  Ideally as straight a line as possible is required to move from the initial model in space S^2 to the optimal model position in Figure 1.1. Realistically this trajectory will not be straight but will be determined by the form of the initial model, the model parameter set and the mechanics of the proposed updating procedure.

  Since each FE model is different the challenge here is to design an updating procedure independent of any particular FE model. In this context of multiple models this is our efficiency criterion.

The final question to be considered in model updating should be;

- **Is the final updated model the best one?**

  As concluded in [13,38,39] it is difficult to compare and decide on the best model in a set of competing models of one system. This question essentially requires some form of proof that the updated model is the best. All models, constrained by the practicality clause, will attain certain optimal positions once updated. One can then simply calculate the distances between these positions and measured data and determine the closest model. In Figure 1.1 the updated model M_C achieves the smallest distance while M_B the largest. This does not strictly mean M_C is the best updated model. Perhaps M_C is not

significantly better than M_A and or M_A may require less 'adjustments' to achieve a position close to M_C even thou it is more complex than M_C.

Therefore the challenge here is to develop an evaluation criterion for which model is the best in a given set.

1.4 Thesis Approach

As can be seen from the general FEM updating problem overview given above; the challenges faced by any one specific FE model are faced by all possible FE models. The approach taken in this work is to propose addressing the stated challenges within the multi-FE model framework. We assume a number of potentially correct FE models are a priori proposed for any one measured real system. The advantage is that whichever set of models one starts off with the proposed method in each chapter will be viable.

1.5 Research Objectives

The thesis objectives are as follows:

- Propose and implement an automatic procedure to identify and select the most uncertain parameters in any FEM. The aim is to propose a generic method that can be used on any model. The attraction of such a proposal would be that anyone can plug their model onto this algorithm and it will automatically produce the model's most uncertain parameter(s). Furthermore the optimal parameter selection will be reproducible.

- Design and implement an efficient FEM updating procedure in the context of multiple competing models. The aim here is to be able to update multiple FE models in one procedure and to determine which model is the best. Updating one model in isolation is not that informative given that someone else can propose a different model for the same system.

- Propose and evaluate a relative goodness measure for the updated FEM. After developing a model goodness measure, quantify why a particular

model is better than another. This is fundamental to FE updating especially in the multi-model context.

1.6 Thesis Scope

This thesis addresses the following: Given a set of competing initial FE models; how can we identify the most uncertain parameters in each model, efficiently update the models and determine which model is the best updated model.

This thesis has the following limitations:

- This work is not concerned with finding the optimal system measurement data. This means we do not address how to obtain the best real system measurements i.e. to locate the best position in space S^1. It is assumed this can always be improved on with, for example, better measuring instruments but doing so will not change the fundamental FE model updating problem.

- This thesis is not concerned with finding the best working domain(s); whether it is the frequency, time or a combination of both, to address the problem. The model updating methods proposed herein are general enough, albeit were developed on a specific domain, to be implementable for any chosen domain.

- This work is not concerned with individually optimizing any one particular FE model as each model exists in a pool of other potentially correct models.

- This thesis only considers lightly damped structures as examples but the updating methods proposed are general enough to be adaptable to heavily damped systems. This is because we do not consider specific model details but general problems that are faced by any FEM updating scheme.

1.7 Thesis Contributions

- To address the first challenge this thesis proposes and implements a variation of an evolution based algorithm for identifying the most uncertain model parameters.

The method proposed simultaneously identifies and updates these parameters over a number of iterations. The result is a selection of the updated, most uncertain parameter combinations, which result in a good model. Furthermore one has a history of which other parameter combinations performed well at different stages of the updating process. When the proposed method was used with a particular objective function it arrived at the final best parameter set early on in the evolution process. This is a very attractive aspect of this method as evolution processes are normally computationally expensive.

- To address the second challenge the thesis proposes and implements an evolution based algorithm for multiple FEM updating.

This method simultaneously updates any number of proposed FEMs over a set number of generations. The result is the fittest (or most accurate) model on the problem. Again one has a history of which model was the leading model at each generation. This is helpful when one needs to study the behaviour and interactions of the models throughout the generations. This approach to model updating is attractive in that we have a global picture of how competing models behave and compare as opposed to updating one model at a go and not having an idea how other models would perform under the same evaluation.

- To validate that one model is the best, a Bayes' theorem based measure is proposed and implemented on a number of competing FE models.

The result is one; an updated set of FEMs and two; their performance comparison and ordering with justification using established statistical measures. One measure, Jeffrey's scale defines the significance in the differences between the

updated models. The other measures reveal how the model used the parameter space while it was updated. The by-product of the method is the posterior probability distribution of the model updating parameters. These posterior probabilities can further be used to understand how the modal properties change within this probability distribution.

1.7.1 Publications

The following papers contributed to the results presented here and were published:

- Mthembu L, Marwala T, Friswell M.I., Adhikari S, **Finite Element Model Selection Using Particle Swarm Optimization,** In Proceedings of the XXVIII International Modal Analysis Conference (IMAC), 1- 4 February, 2010, Jacksonville, Florida, USA.
- Republished by Springer as: Mthembu L, Marwala T, Friswell M.I, Adhikari S, **Finite element model selection using Particle Swarm Optimization**, Dynamics of Civil Structures, Vol. 4, Conference Proceedings of the Society for Experimental Mechanics Series, 2011, Volume 13, 41-5.
- Mthembu L, Marwala T, Friswell M.I, Adhikari S, **Model selection in finite element model updating using the Bayesian evidence statistic**, Mechanical Systems and Signal Processing, 25 (7) (2011) 2399-2412.
- Mthembu L, Marwala T, Friswell M.I, Adhikari S, **FE Model Updating Using a Population-based Incremental Learning Approach**. Mechanical Systems and Signal Processing (Submitted 2011).

1.8 Chapter Summary

This chapter started by introducing the general idea of models. A concise background to finite element models was then presented. The FEM updating problem is given together with a graphical illustration of the fundamental meaning of the problem. The reasons for updating the initial model have been presented as that an initial finite element model of a system needs to be updated because it results in an approximation that differs from the measured real system results. The

need for model updating and the challenges of FEM updating are further explained using the same graphical illustration. The first challenge was identified to be the ability to develop an automatic FEM uncertain parameter identification or select methodology. The second challenge was identified as the ability to develop an efficient updating procedure and the third was to propose measure how good any updated model is in a set of models. The thesis scope lists the issues not addressed in this work and the reason thereof. The next two sections covered the research objectives and the novel thesis contributions.

1.9 Thesis Layout

As a guide to each chapter, a statement to be immediately addressed is posted in the beginning of the chapter. Each chapter then introduces the proposed method to address the question. A graphical explanation of the idea is presented for Chapters 2 and 3. Next a detailed technical description of the method is presented. Finally some simulation results and discussion of the application of the proposed method are presented. The thesis concludes with an Appendix of modeled structures.

Chapter 2. This chapter addresses the uncertain parameter selection problem. An automatic uncertain parameter selection or identification procedure is introduced by applying it on a mechanical structure.

Chapter 3. This chapter addresses the efficient FEM updating problem. An efficient method to simultaneously update a number of competing models is introduced. The main assumption in this chapter is that such a list of competing models is a priori available or can easily be generated.

Chapter 4. This chapter addresses the question of best updated FEM identification. A method to measure or provide evidence that the updated model is the best is presented.

Chapter 5. This chapter concludes the thesis. A summary of the proposed methods is presented. Their assumptions, advantages and limitations are expanded.

CHAPTER 2

Uncertain Parameter Identification

How to identify the most uncertain parameters of a model?

Given an initial model with a certain number of parameters and the fact that the analytical results of this model do not equate to the measured results means there is some inherent system modeling uncertainty. There are certain aspects of real systems that are currently known to be difficult to model accurately. For structural systems these can vary from member joints, sensors, damping and substructure characterization etc. This chapter focuses on the identification of such uncertainty, specifically the identification of parametric uncertainty in the initial model.

2.1 Introduction

Since FEM updating is a system-identification problem, it is concerned with which and how many model parameters are sufficient to correctly capture the true dynamics of the real system. This problem is further complicated by the constraint placed on the quantification of the model parameters. An evolution-based approach for optimal FEM updating parameter selection is proposed. The assumption of which is that an initial model with a set of potential updating parameters exist. The modeling uncertainty in this case becomes which combination of these will result in a good FE model. The proposed algorithm, population based incremental learning (PBIL), provides an automatic mechanism to finding the optimal combination of updating parameters and updating them. This procedure addresses the, "Which and how many model parameters are required for a good FE model?" challenge set out in Chapter 1. A number of papers have proposed other methods (mainly parameter sensitivity analysis) for parameter selection e.g. [46,59,68].

2.2 Chapter outline

Section 2.3 introduces the population based incremental learning idea and presents its interpretation in different contexts. Section 2.4 shows the graphical interpretation of the mechanics of the algorithm. Section 2.5 presents the technical description of the algorithm. Sections 2.6 and 2.7 introduce the modelled structure and the candidate FE models used in the simulations. Sections 2.8 and 2.9 discuss the proposed PBIL pseudo code used in all of the experiments and two cost/objective functions are defined. The experimental results are presented in Section 2.10 and Section 2.11 concludes the chapter.

2.3 Population-Based Incremental Learning (PBIL)

The PBIL algorithm was first introduced by Baluja [42]. The algorithm is a variation on the standard evolution-based class of search algorithms, the most popular of which is the Genetic Algorithm [23,43,44,45,46]. A Genetic Algorithm (GA) was applied to FEM updating in [7,16,21,22]. GAs have recently been

applied to damage detection in beams [17,20,24]. The characteristic performance dynamics of GAs has been studied in detail in [43,44]. All evolution based algorithms share the same fundamental characteristics;

- They consist of individuals - each individual is a potential solution to the problem
- There is some form of reproduction and or mutation operation(s)
- The algorithms are stochastic
- There is a fitness/cost/performance function
- The evolution stopping criterion is normally the predefined number generations

The individuals form a population which exists within a generation. These individuals evolve/update via specific operations not unlike biological ones. There is an element of randomness introduced to the evolution process. Each individual's performance/fitness on the problem is evaluated through some predefined cost function. All these characteristics are also applicable to the PBIL algorithm [47]. PBIL has been used in a number of fields e.g. to minimize frequency span allocation in telecommunication networks [48], to solve the Prisoner's Dilemma problem [47], to find the optimal position for the watermark on an image [49] and to solve general inverse problems with constrained parameter values[50]. The next two sections describe the typical implementation of PBIL in the area of pattern recognition (PR) [51,52,53] and our adaptation to the FEM context.

2.3.1 PBIL in the PR context

In pattern recognition measured data is derived from some phenomenon, e.g. housing prices, stock market valuations, images, biomedical gene analysis etc. This data is often characterized by a large number of parameters (i.e. data is high dimensional) due to the lack of certainty on what features/parameters essentially

determine the observed phenomenon. For example housing prices have a large number of parameters that could affect them; house location, room sizes, proximity to schools, age, etc. To understand patterns in this type of data, PBIL is typically applied as a feature-selection/reduction method [54,55,56]. In that research area one seeks a small subset of parameters that captures the *essence* of the normally high dimensional data. This is particularly useful in the clustering problems where the data label is unavailable [57,58]. This is useful in visualizing and forming lower complexity mathematical models to the data for prediction purposes. The PR implementation of PBIL does not place a constraint on the final number of data parameters. For example, by applying PBIL in feature selection, data with 30 parameters could be reduced to just 4 parameters that explain the data well. In the PR implementation of PBIL the other 26 parameters are simply discarded/eliminated from the model. This will result in a much simpler mathematical model to capture the underlying patterns in the data.

2.3.2 PBIL in the FEM context

The main difference between the implementation of PBIL in the PR and FEM context is if structural thickness, for example, is an uncertain parameter it cannot be eliminated from the model. An FE structural model cannot have the thickness variable eliminated as the thickness is part of the geometry that describes the structure. In the FEM context a PBIL eliminated uncertain parameter means that it is not chosen to be updated. The said parameter retains its mean/prior or measured value in the updated model. This means in every generation, all individuals *always* have *d* potential updating parameters but not all *d* will be simultaneously updated at any one generation.

2.4 Graphical Interpretation

Figure 2.1 shows a hypothetical setting of one model, M_A with five uncertain parameters (α_1, α_2, α_3, α_4, α_5), each parameter a potential updating parameter. G1 to G4 are the chosen number of generations the algorithm will run for. The algorithm is set to have a population of three individuals. Every generation always has three potential solutions to the model updating problem; M_{A1}, M_{A2} and M_{A3}

and each model also always has five parameters. In the first generation three random models are generated by randomly selecting parameters to be updated from the five available. These selected parameters are identified by '1' meaning a parameter is chosen for updating and the '0' means not. The outcome is; $M_{A3}(0,1,1,1,0)$, $M_{A2}(1,0,0,0,1)$ and $M_{A1}(1,0,1,0,0)$. The parameters are then updated. The evaluation of this particular set of individuals using the chosen objective function produces a minimum error for model $M_{A3}(0,1,1,1,0)$. This can be seen by the shorter distance between this model and the measured data in Figure 2.1.

After this evaluation the argument is model $M_{A3}(0,1,1,1,0)$ performed better because of its chosen parameters. Future models should have similar traits/parameters/features to model $M_{A3}(0,1,1,1,0)$. In generation G2 the second, third and fourth parameters in other models begin to have a higher probability of being selected for updating. Models M_{A1} and M_{A2} start adopting some of model M_{A3}'s features. Model M_{A2} adopts the fourth parameter whilst retaining its previous set of parameters. Model M_{A1} selects the second parameter for updating. These models are prevented from wholly adopting all model M_{A3}'s features by 'relaxing' the probability vector (see Section 2.5 equation 2.4).

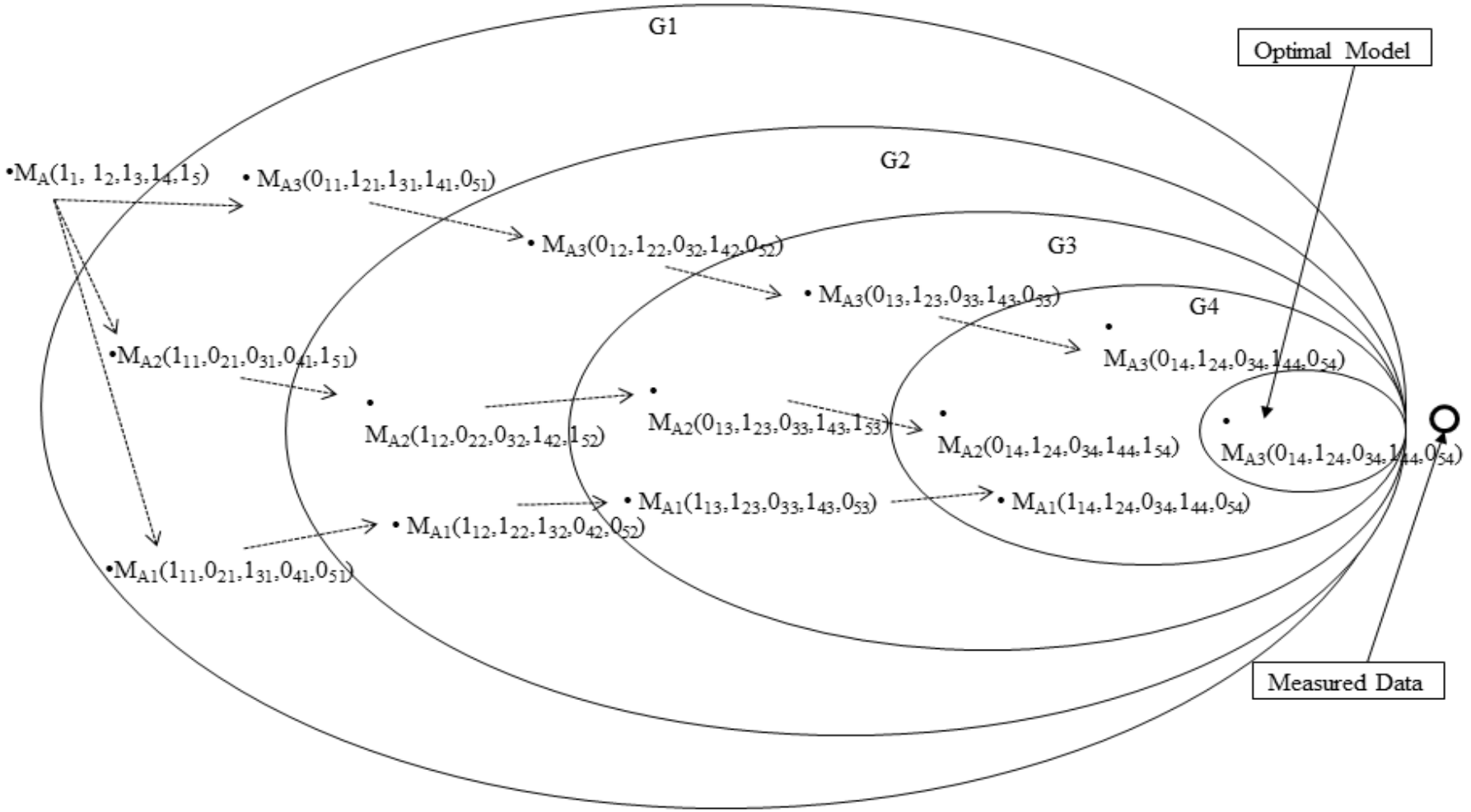


Figure 2.1 General PBIL updating scheme

In the second generation model $M_{A3}(0,1,1,1,0)$ changes slightly to $M_{A3}(0,1,0,1,0)$ due to some randomness in the algorithm. This updated model still evaluates well and its features are reinforced on the others. In generation G3 models M_{A1} and M_{A2} change some of the selected updating parameters but model M_{A3} does not. In the last generation all the models do not change their updated parameters but may change the magnitude of the updated parameters. This is shown by the evaluation of model M_{A1} surpassing that of M_{A2} in the last generation even though the selected updating parameters have not changed.

The next section presents the general mathematical operators of the PBIL algorithm.

2.5 PBIL Operators

Let n be the number of individuals in every population of every generation, *gen*. Each individual (potential solution) is denoted by a vector of parameters given in [42] by:

$$I_j = \{p_{j1}, p_{j2}, p_{j3}, \ldots . p_{jd}\} \quad 1 \le j \le n \tag{2.1}$$

I_j is the j^{th} individual in the population and p_{j3} is the j^{th} individual's third updating parameter. The problem dimension (d) is defined by the maximum number of potential updating parameters for that particular model. Thus for d model parameters the maximum number of valid individuals (parameter combination) that can be generated is 2^d-1.

In the first generation each individual is a vector of randomly selected parameter combinations. Subsequent generation individuals are generated based on the probability vector (***pv***). The probability vector tracks the best performing parameter combination in each generation and probabilistically aligns future individuals to have these combinations. The probability vector is the same dimension as each population individual, i.e. it has d elements. Before the first generation of individuals are generated this vector is set to [42]:

$$pv = \{0.5\} \quad \forall d \tag{2.2}$$

asserting our priori uncertainty of which of the d model parameters is likely to be selected for updating. Setting the probability to 0.5 means each parameter has a 50% chance to be selected. A vector s with d elements - the same size as pv - has its entries randomly generated from a uniform distribution. The entries of s and pv are compared by a relational operator (see Table 2.4). The result of this operand is a binary vector m. The indices of the true m entries correspond to the model parameters selected for updating. The false entries are not updated. At the end of each population the probability vector is updated. This is done through [42]:

$$pv = 0.9 \cdot pv + 0.1 \cdot bk \tag{2.3}$$

where bk is the binary code vector that records the best performing parameter combination after each population. Applying equation 2.3 results in the probability vector increasing the probability of future vector-m entries being similar to the bk vector. This is not completely desirable as it can lead to a premature convergence of the algorithm to a sub-optimal set of updating parameters. To maintain diversity of future generated individuals the probability vector is immediately computed as follows:

$$pv = 0.95 \cdot pv - 0.05 \cdot (pv - 0.5) \tag{2.4}$$

This particular form of this equation was found to give the best population diversity while retaining focus on the better features. Section 2.8 outlines the implementation of the FEM-PBIL algorithm. The next section describes the modelled structure.

2. 6 Modelled Structure - GARTEUR SM-AG19 Testbed

Three FE models of the GARTEUR SM-AG19 testbed aeroplane structure were used to evaluate the uncertain parameters selection ability of the PBIL algorithm. Appendix A2 gives background information of the GARTEUR SM-AG19 testbed.

2.6.1 Finite Element Models

All the FE models are modelled using the Structural Dynamics Toolbox, SDT® 6.2, for Matlab®. Figure 2.2 shows the two types of structural element choices; plate and beam, used to model the GARTEUR aeroplane. In our models all element materials are standard isotropic. The beam model uses Euler–Bernoulli type elements and the plate model uses Kirchhoff shells. Two real experimental data; one from the DLR center, Gottingen, Germany and the other data from the Imperial College of Science, Technology and Medicine (IC), United Kingdom are considered. The natural frequencies measured by these three institutes are shown in Table 2.1.

Table 2.1 Measured modal data from two Institutes, DLR and IC

		Mode No.	1	2	3	4	5	6	7	8	9
Institute	**DLR**	**Natural Frequency (Hz)**	6.38	16.10	33.13	33.53	35.65	48.38	49.43	55.08	63.04
	IC		6.54	16.55	34.86	35.30	36.53	49.81	50.63	56.39	64.96
Difference (%) (DLR-IC)/IC			-2.50	-2.79	-5.22	-5.27	-2.46	-2.95	-2.42	-2.37	-3.04

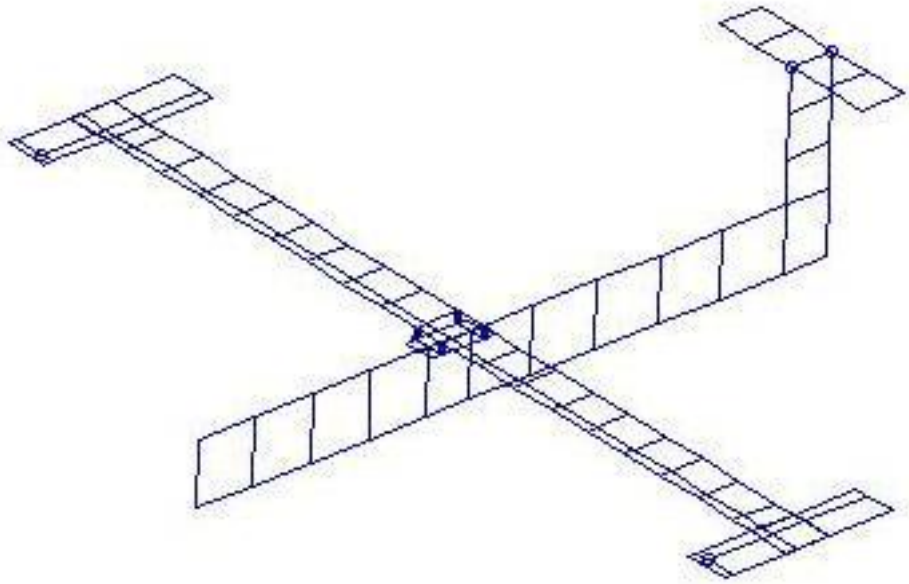


Figure 2.2a Plate FEM of Garteur SM-AG19 structure

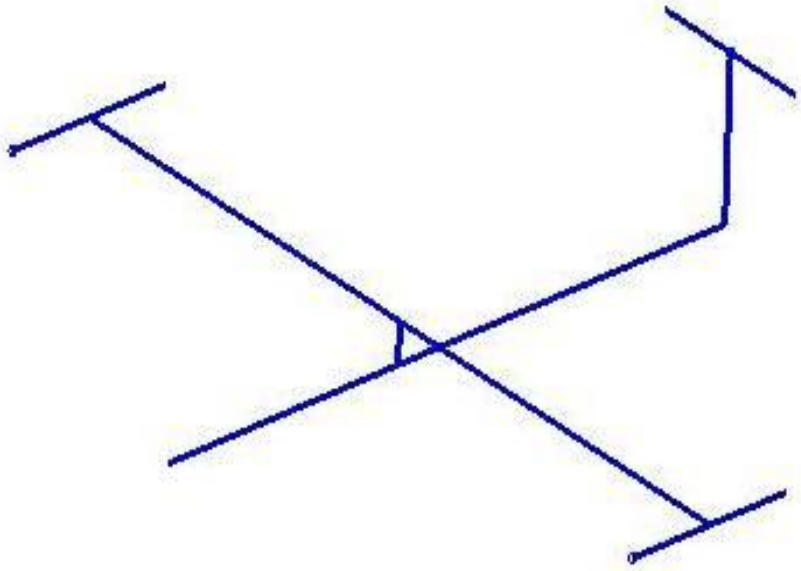


Figure 2.2b Beam FEM of Garteur SM-AG19 structure

2.7 Candidate FE Models

Three models used for comparison consist of one plate element model and two beam element models for the same GARTEUR SM-AG19 test structure [correct 39]. The models are named according to their finite element types, e.g. model B_3 is a beam element model, and the model numbering is arbitrary. The beam models B_1 and B_3 have 6 and 7 updating parameters respectively while the plate model P_1 has 8 parameters. As can be seen from Table 2.2 the updating parameters are quite varied across the three models .

The updating parameter numbering varies according to the property considered and the part of the test structure updated e.g. E_2I_{2min}, means the Young's Modulus and the Second moment of inertia of the vertical tail are updated. Table 2.2 shows the three chosen models and the types of updating parameters used. The plate model has two parameters for the main wing; one for aluminium (unconstrained) and one for the sandwich part (constrained). The spring stiffness updates the four-spring parameters on the wing to fuselage connection (see Appendix A2.1).

2.7.1 Initial FE models

In order to qualify the model updating capabilities of the proposed PBIL algorithm, three models with standard material/geometric properties based on the previously identified (20,55,59) updating parameters are simulated. Table 2.3 shows the FEM natural frequency (NF), percentage natural frequency difference (PNFD) and average error (Avg. Error) results for each model compared to the two measured data. The STD_{mdl} column lists the natural frequency of each model using standard material properties (SMP) and geometric measurements (GM).

The Δf (%) column lists the percentage difference between the STD_{mdl} and measured natural frequency data from the relevant institute. The last row lists the Average Error (Avg. Error) which is the sum average of the absolute error between the STD_{mdl} and the measured data.

Table 2.2 FEMs and updating parameters for GARTEUR SM-AG19 structure

Model Name & Element Type	**Fuselage**	**Wing**	**R-Wing**	**L-Wing**	**Wing/Fuselage connection**	**V - Tail**	**Wing Thickness**	**R-Drum**	**L - Drum**	**Overall Density**	**Residual Type**
P_2 (Plate)	E_1 G_1	E_6 G_6			E_7 G_7 *(Steel)*	E_2 G_2	W_t	*Mass* (M_l)	*Mass* (M_r)	ρ	*Frequency*
B_1 (Beam)			I_{6MIN}, I_{6MAX},J_6	I_{5MIN}, I_{5MAX}, J_5		I_{2min}				ρ	*Frequency*
B_3 (Beam)	$E_1 I_{1min}$	E_4 I_{4MIN}, E_4 I_{4MAX}, ρ_4	G_6J_6	G_5J_5		E_2 I_{2min}					*Frequency*

$E\,I_{min,max:}$ minimum and maximum bending stiffness. GJ: torsional rigidity. E: Young's modulus. G: shear modulus. ρ: mass density. M_R/M_L: right and left mass. W_T: wing thickness. Frequency: natural frequency.

Table 2.3 NF, PNFD and Avg. Errors for P_1, B_1 and B_3 using SMP and GM

	Model P_1			**Model B_1**			**Model B_3**		
	STD_{mdl}	**DLR**	**IC**	**STD_{mdl}**	**DLR**	**IC**	**STD_{mdl}**	**DLR**	**IC**
Wn	**Hz**	**Δf (%)**	**Δf (%)**		**Δf (%)**	**Δf (%)**		**Δf (%)**	**Δf (%)**
1	6.69	-4.87	-2.30	5.77	9.64	11.85	5.76	9.65	11.86
2	18.99	-18.01	-14.80	15.44	4.10	6.70	15.44	4.11	6.71
3	40.47	-22.16	-16.10	30.98	6.49	11.13	31.10	6.11	10.77
4	41.56	-23.96	-17.75	31.49	6.08	10.79	31.66	5.59	10.32
5	41.88	-17.47	-14.64	33.78	5.25	7.53	33.82	5.14	7.43
6	53.34	-10.26	-7.09	45.15	6.67	9.35	45.15	6.67	9.36
7	53.51	-8.27	-5.71	54.82	-10.90	-8.27	54.79	-10.86	-8.23
8	57.69	-4.74	-2.31	56.14	-1.93	0.44	56.13	-1.91	0.45
9	71.20	-12.95	-9.61	60.02	4.79	7.60	60.01	4.80	7.61
	Avg. Error	**13.64**	**10.40**		**6.21**	**8.19**		**6.21**	**8.19**

In both sets of measured data, model P_1 has the highest average error. Clearly the initial beam models are a better approximation of the given GARTEUR SM-AG19 structure based on both measured dataset. The main difference between model B_1 and B_3 is that model B_1 mainly updates structural geometric parameters bar density. B_3 updates a combination of material and geometric parameters as seen in Table 2.3. The next section describes the PBIL code implemented to update the parameters of these three models.

2.8 PBIL Pseudo-Code

The FEM-PBIL pseudo-code with additional algorithm settings used in all simulations is presented in Table 2.4. This is the Matlab® algorithm implementation for one FE model.

2.8.1 PBIL Pseudo Code Description

In Line 4 the initial FE model with standard material/geometric values for the *d* parameters is evaluated. The fitness is the objective function used to measure the goodness (error) of the model, see Section 2.9. The resultant model fitness value is defined as the best so far in Line 5a. Next the probability vector is initialized such that all entries have an equal chance for update selection in Line 6.

Table 2.4 FEM-PBIL algorithm pseudo code

```
1.  gen = 30; n=500;            %  No.  of Generations & Population size
2.  d =  6;                     %  No. of potential update parameters, 6 for model B1
3.  I_init = [μ];               %  Set update parameters to Standard/mean material/geometric values.
4.  imdl_fit = init_fit (I_init);  %  Calc Fitness* of initial FEM.
5a. best_fit = imdl_fit;        % Best fitness value so far is initial FEM result.
5b. worse_fit = imdl_fit;       % Worse fitness value so far
6.  pv = [0.5];                 % Set all d Probability Vector entries equal to 0.5.
% Begin Iterations
7.  for 1 to gen                %  Number of generations
8.      for  1 to n             %  Number of individuals in a population
9.          s = rand (d);       % Generate d random numbers from uniform
                                %  distribution [0,1].
9.          m = s < pv;         % Evaluate inequality. The vector m is equivalent to
                                % an individual.
10a.        Update model parameters that correspond to true m entries  &
10b.        Set model parameters for which m is false to standard measured values.
11a.        current_fit = Fitness of model with updated parameters (& un-updated
                          parameters).
11b.        iff current_fit < best_fit
12.           best_fit = current_fit;     % Current fitness is now best fitness
13.           bk = m;                     % Keep vector m which is the proxy
                                          % for current best parameter combination
14.         elseif current_fit > worse_fit
15.           go back to loop
16          end if loop
18.   end % End Population loop
19.    Apply equation 2.3;      % Update probability vector
20.    Apply equation 2 4;      % Relax probability vector
21. end  % End Gen loop
```

**The above code assumes the problem fitness or objective function has been pre-defined (see Section 2.9).*

In every generation when each individual is generated, *d* uniform random numbers are generated to form vector *s.* This vector is compared to the current probability vector in Line 9. The vector *s* entries corresponding to random numbers that are less than 0.5 are assigned true (1) and the others are set to false (0) in the vector *m*. Vector *m* is a proxy for an individual. The model parameters that correspond to the true entries in the *m* vector are selected for updating. Those that correspond to the false entries are set to standard material/geometric values.

In Line 10a each selected parameter is updated using:

$$\mu_{i+i} = \mu_i + \lambda\sigma \tag{2.5a}$$

$$where \;\; \mu = mean, \lambda = (2 \cdot r - 1), \sigma = step \tag{2.5b}$$

The variable *r* is a random number from a uniform distribution between [0, 1] and σ is the step size (See Table 2.5). This form of updating is chosen to eliminate large variations in the updating of each parameter. This is critical if a parameter is to be consistently considered important by the probability vector. As an example Figure 2.3a shows the sampling of the density parameter space using equation 2.5a over 200 iterations. This figure clearly shows the small steps taken in updating the density parameter.

Sampling from a normal distribution significantly varied the potential-updating parameter values between populations and generations. This resulted in a consistently varying best-parameter combination through successive generations. Figure 2.3b shows the sampling of the density parameter from the normal distribution space over 200 iterations.

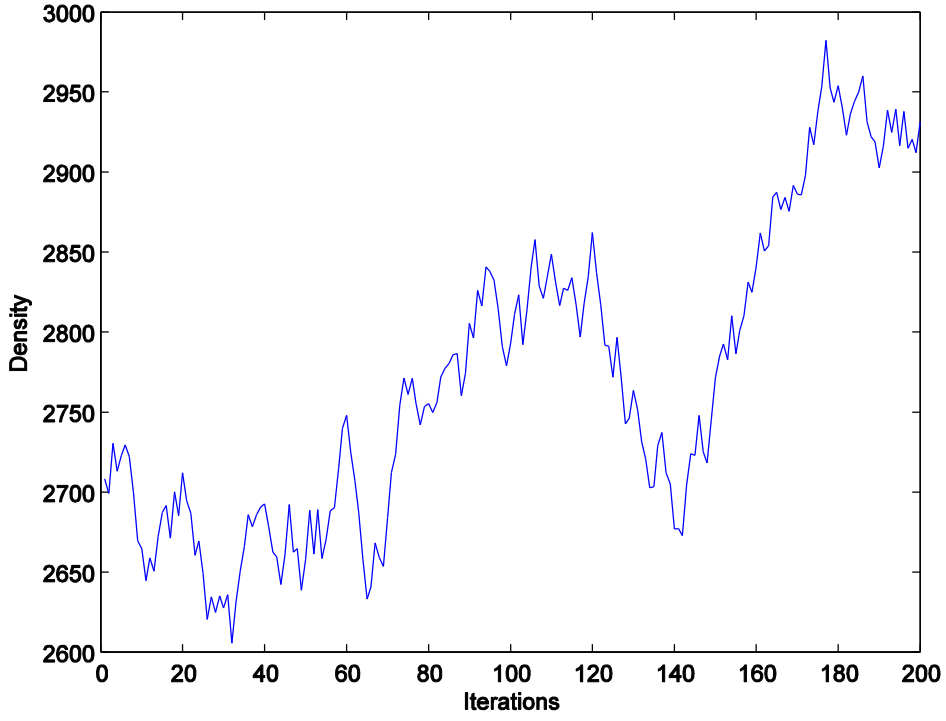


Figure 2.3a Density samples using Equation 2.5

Line 11a evaluates the fitness of the finite element model with the updated parameters. If the current FEM fitness is better than the last best fitness its binary mask (m) is stored in the vector bk in Line 13. At each generation the probability vector is updated as shown in Lines 19 and 20 (as discussed in Section 2.5).

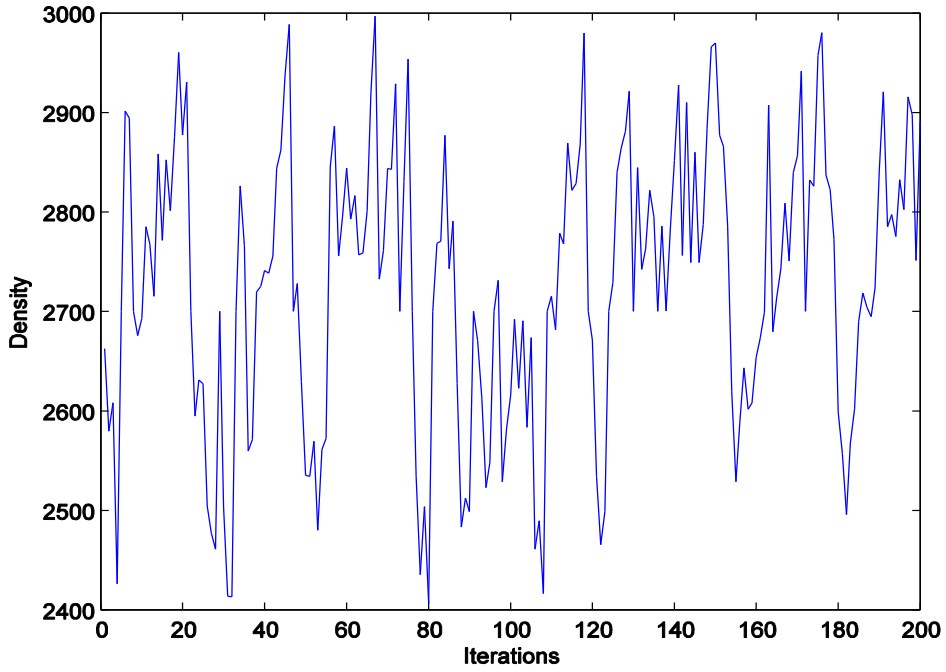


Figure 2.3b Density samples from a normal distribution

2.9 Fitness Functions

To determine if the best updating parameter combination is affected by the choice of objective function we compare two cost/objective/fitness functions; the Bayesian Information Criterion (BIC) [60,61,62,63] and the average Squared Sum of Errors (SSE).

The BIC function is given by the following equation:

$$m\log(\delta^2) + d\log m \tag{2.6a}$$

where,

$$\delta^2 = \frac{\sum_{i=1}^{m}(f_{data} - f_{fem})^2}{m} \tag{2.6b}$$

and d is the number of model parameters to be updated, m is the number of measured modes, f_{data} and f_{fem} are the measured and analytical natural frequencies respectively. The average sum of squared errors is given by [60]:

$$SSE = \frac{\sum_{i=1}^{m}(f_{data} - f_{fem})^2}{m} = \delta^2 \tag{2.7}$$

The first term of equation 2.6a is commonly referred to as the data-fit term and the second term is known as the model complexity penalty term. This assumes a model's complexity is determined by the number of free variables in the mathematical model. The minimization of the average sum of squared error objective function does not account for model complexity but only measures how well a given model approximates the measured data. Thus minimizing the BIC as opposed to the SSE function will mainly reduce the number of free model parameters.

The next section presents the simulation results obtained by the FEM-PBIL algorithm for the two objective functions.

2.10 Simulations

In all simulations the number of generations, gen, is set to 30 and the population size, n, is set to 500. To confirm the consistency of the proposed PBIL algorithm three simulation runs of each model were run for each of the two objective functions. All models used the updating parameter settings shown in Table 2.5.

Table 2.5 Prior mean and step values for potential updating parameters

Parameter	**Mean (μ)**	**Step (σ)**	**Limits Min / Max**			**Parameter**	**Mean (μ)**	**Step (σ)**	**Limits Min / Max**	
$E_{1\text{-}4}$ *(Pa)*	7.2e10	3.16e9	6.0e10	8.2e10		I_{5min} (m^4)	8.3e-9	3.2e-11	7.63e-9	9.9e-9
G_6 *(Pa)*	2.8e10	3.16e9	1.8e10	3.2e10		I_{5max} (m^4)	8.3e-7	3.2e-9	7.63e-7	9.9e-7
ρ_4, ρ (kg/m^3)	2700	31.62	2.4e3	3.0e3		I_{6min} (m^4)	8.3e-9	3.2e-11	7.63e-9	9.9e-9
M_r, M_l, M_t *(kg)*	0.18	1e-3	0.15	0.20		$I_{6max}$$(m^4)$	8.3e-7	3.2e-9	7.63e-7	9.9e-7
I_{1min} (m^4)	1.56e-6	3.16e-9	1.27e-6	1.96e-6		J_5 (m^4)	3.1e-8	3.2e-10	2.8e-8	3.8e-8
$I_{2min}$$(m^4)$	8.33e-9	1e-11	7.63e-9	9.9e-9		J_6 (m^4)	3.1e-8	3.2e-10	2.8e-8	3.8e-8
Springs (S_1) *N/m*	1e12	1e-11	9.6e11	1.45e12		W_t *(m)*	9.99e-3	1e-4	9.0e-3	1.1e-2

Figure 2.4a shows the results of three different simulation runs of model B_1 using SSE as the objective function and Figure 2.4b shows three simulation runs of model B_1 using BIC as the objective function. These figures show the evolution of the best updating parameter combination (BUPC) within 30 generations. The horizontal axis is the number of model updating parameters, counting left to right as listed in both Tables 2.6a and 2.6b for each model. The vertical axis lists the generations in which the current best-updating-parameter combination is achieved. The colours white or black indicate whether the particular updating parameter was selected for updating or not respectively. Since Figure 2.4 presents model B_1 results, the number of potential-updating-parameters is six, as shown on the horizontal axis.

Each graph shows the current BUPC and the final BUPC. The current BUPC is the parameter combination that achieves a better error than the previous best error so far in the algorithm iterations. This means within any one generation a number of current BUPC can be found. The final BUPC is the model's optimal parameter combination given the algorithm stopping criterion (i.e. maximum number of gen = 30 in this case).

2.10.1 SSE as the Objective Function

The first simulation in Figure 2.4a selected the final BUPC, in generation (gen) number 26, to be parameters 1, 3, 4. The second simulation lists the final BUPC as 1, 2, 4 in gen 28 and the third run lists the final BUPC as parameters 1, 5 which was found in gen 17. These final BUPC did not improve the objective function value in the subsequent 4, 2 and 13 remaining generations respectively. When using the SSE as the objective function the algorithm does not converge to the same final BUPC on different simulation runs.

Secondly the most frequently selected updating parameter is sometimes not one of the final best updating parameters (results not shown). Thirdly the algorithm still obtains new current BUPC at later generations (generations close to the algorithm stopping criterion) when compared to the BIC function case.

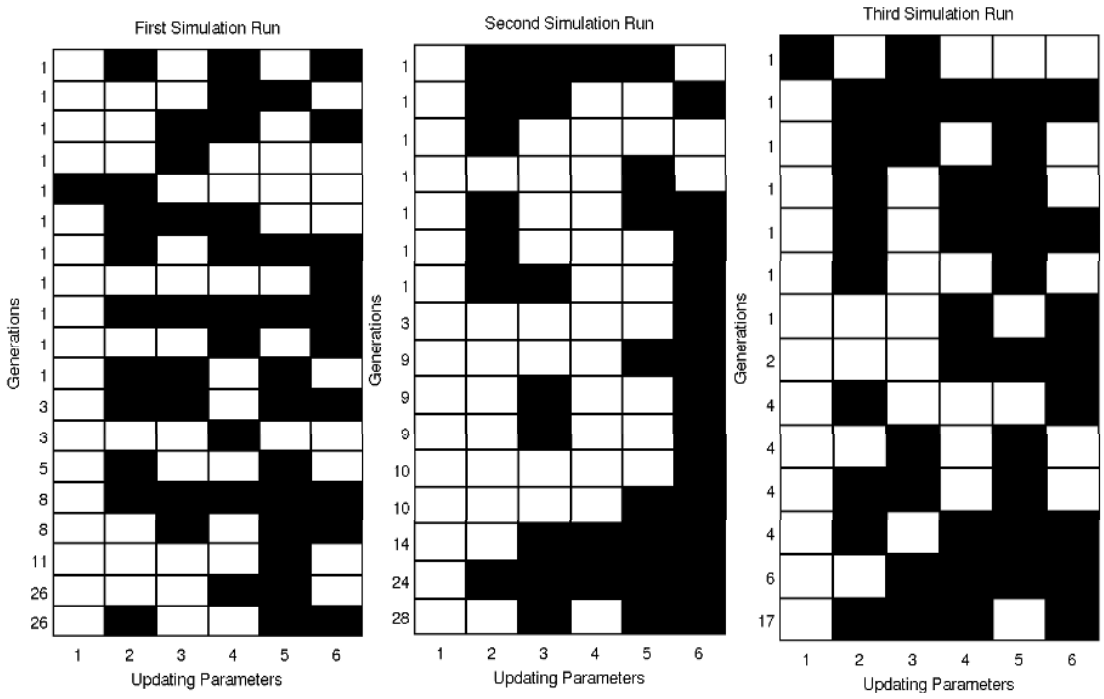


Figure 2.4a BUPC evolution for B_1 using DLR data and SSE function

2.10.2 BIC as the Objective function

All simulations in Figure 2.4b found the final BUP to be updating parameter number 1.When using the BIC error function the updating parameter combination that becomes the final BUPC is identified earlier in the algorithm generations. Even though this parameter is identified earlier on, the parameter updating process still improves the FE model error in subsequent generations hence the additional 11 current BUPC after the initial identification of this parameter. When using the BIC function the most frequently selected parameter is always the final BUP. Selecting the BUPC using the BIC function results in a fewer updating parameters. This is not surprising as the BIC objective function penalizes the number of model parameters as well as the SSE term.

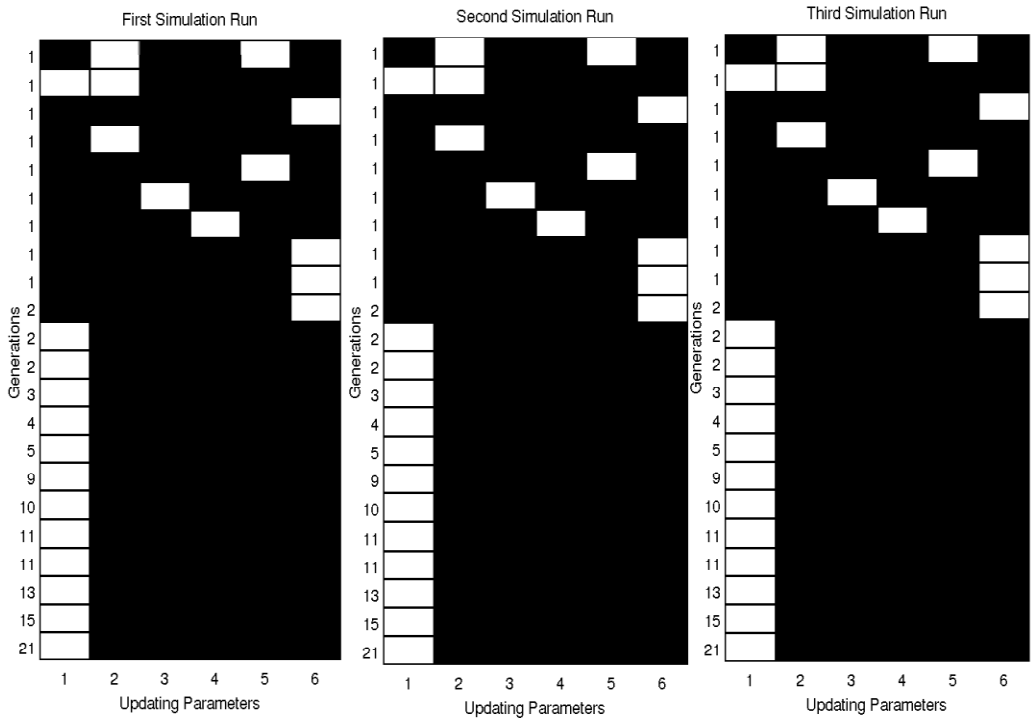


Figure 2.4b BUPC evolution for B_1 using DLR data and BIC function

2.10.3 Results Summary

Table 2.6 shows the combined results of the three simulations for each model, on the BIC objective function and each measured dataset. Table 2.7 shows similar results to Table 2.6 but using the SSE as the objective function. All simulations were on both IC and DLR data. In these tables the Average Percentage Selection Frequency (APSF) of each parameter over the three simulation runs is shown as a percentage. The average selection frequency is the number of times a particular parameter is selected as one of the current best updating parameters over the total number of counts that a current BUPC is found.

Table 2.6 APSF for each parameter for BIC function

Bayesian Information Criterion (BIC)			E_2	E_5	E_6	M_l	M_r	M_t	S_1	W_t
	Model P_1	***IC***	2 %	15 %	9 %	46 %	52 %	4 %	6 %	2 %
		DLR	20 %	13 %	16 %	9 %	22 %	16 %	4 %	78 %
			ρ	I_{2min}	$I_{5min}J_5$	$I_{5max}J_5$	$I_{6MIN}J_6$	$I_{6MAX}J_6$		
	Model B_1	***IC***	82 %	9 %	9 %	9 %	0 %	9 %		
		DLR	59 %	14 %	5 %	5 %	9 %	18 %		
			E_1I_{1min}	E_2I_{2min}	E_5I_{5min}	E_5I_{5max}	G_5J_5	G_6J_6	ρ	
	Model B_3	***IC***	12 %	23 %	18 %	8 %	20 %	16 %	74 %	
		DLR	7 %	17 %	28 %	17 %	17 %	0 %	67 %	

For example in Figure 2.4a parameter number 1 (ρ) was selected 18 out of 19 counts in the first simulation, 16/16 in the second, 13/14 in the third, as the current BUP thus an average selection frequency of 96% (see Table 2.7, DLR data, model B_1). Comparing Table 2.6 and 2.7, the most frequent BUP for model B_1 is independent of objective function and measured data. The most frequent BUP for model B_3 is consistent for both BIC and SSE functions and across both measured data.

Table 2.8 shows the final BUPC that produced the best results from the three simulation runs for each model. The final BUPs are identified by their numbering, according to their listing sequence in Tables 2.6 and 2.7. For example, model P_1 identified parameter number eight (W_t) as the final BUP when using the BIC objective function for DLR data while under the SSE function, W_t was identified as the BUP for both data. When the algorithm fitness was set to the BIC function, all models resulted in a single BUP. This is evident from the differences in parameter selection frequency magnitude of one parameter over all the others as shown in Tables 2.6 and 2.7.

The single BUP that is found when using the BIC function is always present (and is at a high selection frequency) in the final BUPC of the SSE function across all models. There was some variation in the final BUPC for each model that used the SSE function in the three simulations. Thus the parameters listed in Table 2.8 are those that produced the lowest error in the three runs. When using the BIC the final BUPC was consistent across all simulations and these are listed in Table 2.8.

Table 2.7 APSF for each parameter for SSE function

Squared Sum of Errors (SSE)			E_2	E_5	E_6	M_l	M_r	M_t	S_1	W_t
	Model P_1	*IC*	33 %	37%	56 %	85 %	87 %	41 %	57 %	80 %
		DLR	48 %	32 %	32 %	64 %	87%	36%	46 %	84 %
			ρ	I_{2min}	$I_{5min}J_5$	$I_{5max}J_5$	$I_{6MIN}J_6$	$I_{6MAX}J_6$		
	Model B_1	*IC*	95 %	39 %	42 %	52 %	38 %	47 %		
		DLR	96 %	47 %	53 %	57 %	40 %	36 %		
			E_1I_{1min}	E_2I_{2min}	E_5I_{5min}	E_5I_{5max}	G_5J_5	G_6J_6	ρ	
	Model B_3	*IC*	25 %	53 %	34 %	54 %	59 %	68 %	100 %	
		DLR	33 %	44 %	74 %	32 %	66 %	60 %	100 %	

Table 2.8 Final BUPC for each FEM on all data for both objective functions

	Model P_1					**Model B_1**					**Model B_3**			
	DLR - SSE	**DLR - BIC**	**IC- SSE**	**IC- BIC**		**DLR - SSE**	**DLR - BIC**	**IC- SSE**	**IC- BIC**		**DLR - SSE**	**DLR - BIC**	**IC- SSE**	**IC- BIC**
Parameter Number	2,4,5,8	8	2,4,5,7,8	4		1,3,4	1	1,3,5,6	1		3,5,7	7	5,6,7	7

Table 2.9 shows the average (over the three simulations) and percentage errors for the three models on all the measured data for the different objective functions. Comparing the results from Table 2.9 and Table 2.3, model P_1 benefited the most from the model parameter selection and updating process. For the P_1 model the most uncertain parameters were the main wing thickness and the accelerometer masses on the wind drums. Model B_3 has the main wing density (ρ_4) as the most uncertain parameter for both measured data. In both cases the updated density value was lower than the standard material value. Model B_1 updated the GARTUER structure's overall density the most. The final updated density value was also lower than the standard material value of 2700 kg/m^3.

Overall the average model errors between the BIC and SSE objective functions for each model are within 1% even though all models selected one final BUP when using the BIC function. Furthermore when all the models were evaluated with the BIC function they always settled on the BUP earlier in the algorithm iterations. This can be considered as the algorithm's 'confidence' on the particular BUP more so when the number of candidate uncertain parameters is high. The ability to quickly (i.e. early in the evolution process) find the BUPC and to achieve a good error margin is a very attractive property for using the BIC as the objective function for FEM updating.

Table 2.9 PNFD and Avg. Errors on DLR, IC data for both objective functions

	Model P_1					**Model B_1**					**Model B_3**			
	DLR - SSE	**DLR - BIC**	**IC- SSE**	**IC- BIC**		**DLR - SSE**	**DLR - BIC**	**IC- SSE**	**IC- BIC**		**DLR - SSE**	**DLR - BIC**	**IC- SSE**	**IC- BIC**
Wn	**Δf (%)**	**Δf (%)**	**Δf (%)**	**Δf (%)**		**Δf (%)**	**Δf (%)**	**Δf (%)**	**Δf (%)**		**Δf (%)**	**Δf (%)**	**Δf (%)**	**Δf (%)**
1	13.76	13.24	12.35	10.87		7.81	8.66	9.69	9.24		8.89	9.04	11.29	11.05
2	0.58	0.43	-0.35	-1.69		2.01	2.99	4.20	3.72		3.71	3.82	6.45	6.34
3	-1.91	-4.00	-1.09	-2.47		4.97	5.68	9.30	9.03		5.99	6.13	10.68	10.68
4	-0.86	-2.93	-0.02	-3.55		4.60	5.29	8.94	8.75		5.57	5.93	10.46	10.59
5	-9.39	-9.43	-7.93	-8.34		3.18	4.15	5.00	4.58		4.24	4.24	6.58	6.16
6	5.40	5.41	5.06	3.99		4.59	5.57	6.91	6.38		4.67	4.67	7.53	6.65
7	-4.40	-4.79	-2.71	-3.12		-13.26	-12.15	-11.07	-11.66		-11.64	-11.67	-8.98	-9.29
8	0.39	-0.31	1.67	0.99		-3.97	3.00	-1.97	-2.47		-2.54	-2.59	-0.18	-0.45
9	-0.40	-0.49	0.32	-0.51		2.68	3.66	5.10	4.61		4.03	4.04	6.93	6.64

2.11 Chapter Summary

Two important aspects in FE model updating are considered. Firstly a method of automatically selecting the best updating/uncertain parameter combination from an initial potential-updating-parameter set is proposed. Secondly the updating of the selected parameters is proposed, both aspects within a population based evolution framework.

Three, two beam and one plate element, FE models of the GARTEUR SM-AG19 test bed are compared on two different measured datasets. The influence of two different objective functions, BIC and SSE, on the selection and updating of the uncertain parameters using the proposed algorithm is studied. The algorithm under the BIC objective function always results in a FE model with fewer updating parameters than the SSE function. This is an attractive aspect of the BIC cost function. The other attraction is the algorithm identifies the most uncertain model parameter early on in the algorithm iterations when the BIC function was employed. Using the SSE function produced results that varied up to the end of the algorithm counter. This is an undesirable characteristic of such objective functions because one is never certain whether running the algorithm for extra iterations would produce a vastly different BUPC.

The plate element model benefited the most from the proposed updating-parameter selection and updating algorithm. All models had consistent performance across both measured data sets and objective functions.

CHAPTER 3

Probabilistic Model Selection and Updating

How to efficiently update a FEM?

As seen from the previous chapter, a number of competing finite element models of one real structural system can be developed. Using the designers' understanding of the problem and its uncertainties this chapter proposes the *simultaneous* updating of all the proposed models. This way no model is a priori preferred for updating but each is objectively evaluated in a group setting to be the best or worst. This paradigm essentially casts the FEM updating as a *model selection* problem. This approach, it will be argued, is an efficient way of not only updating a FE model in a group setting but a convenient procedure of evaluating multiple models simultaneously.

3.1 Introduction

Given multiple initial FE models, each developed a priori from engineering judgment, a choice of the best model for a system has to be made. An evolution-based approach for optimal FEM selection is proposed. The main assumption is that all models occupy the same dimension in space, i.e. all models can be represented by the same *number* of uncertain parameters. The proposed method, particle swarm optimization (PSO), uses the evaluation of each model to influence

the performance of another. This procedure addresses the, “How can the chosen model be efficiently updated?” challenge set out in Chapter 1.

3.2 Chapter outline

Section 3.3 introduces the particle swarm optimization algorithm idea and presents its interpretation in a number of domains. Section 3.4 shows the graphical interpretation of the mechanics of the algorithm. Section 3.5 presents the technical details of the PSO algorithm. Section 3.6 describes the simulated structural beam model. Section 3.7 defines the candidate FE models used in the simulations. Section 3.8 explains the adopted FE model representations and the algorithm settings. Section 3.9 defines the objective function used in the simulations. Section 3.10 presents four simulation results of the PSO algorithm on the modelled beam. Section 3.11 concludes this chapter.

3.3 Particle Swarm Optimization (PSO)

Particle swarm optimization was first developed by Kennedy and Eberhart [64]. PSO is a population-based stochastic search algorithm inspired by the social-psychological behaviour of biological entities in nature when they are foraging for resources. The population/swarm of entities in nature could be that of birds, fish and or ants etc searching for food [64,65,66]. Each entity in the swarm is able to dynamically adapt individually and through group influence to the environment while in search of resources. The swarm adapts by stochastic moving towards previously good regions in the environment. This means the movement of the swarm in the search space has some random elements to it but this movement generally tends to converge to optimal point(s) in the search space.

3.3.1 Variations on PSO

The swarm behaviour metaphor has been adopted by the evolutionary computation community [66,67] where the biological entities are called particles, the swarm is called a population, the environment is the solution space and the resource is the solution to the problem. One of the main differences between evolutionary and classic swarm based algorithms is the way the particles interact.

In a typical evolutionary algorithm (e.g. Genetic algorithm [44]), particles combine and mutate within a population and over generations. In swarm based approaches for example, particles communicate instead of merging. There is no evidence of one method being superior to the other but consensus is that both these approaches are well suited to problems where the solution search space is too large to search exhaustively [43,59,66,67,68].

3.3.2 PSO in other contexts

The mathematical interpretation of this analogy is that instead of starting with one point searching for a minimum or maximum of a function it is advantageous to start with a number of points at different positions on the function space. These initial points will stochastically 'move' through the problem search space searching for the minimum/maximum solution point. This means *each* considered point/particle can potentially find the optimal solution to the problem and indeed the more points the quicker a solution can be found.

In the FEM updating context and approach proposed in this paper; each particle is a potentially correct initial model to the finite element modeling problem. Here we assume the first thesis challenge of the number and identify of updating parameters has been settled. The FE model search space[3] is defined by the number of updating parameters in the models; specifically the maximum number of updating parameters defines the complete search space. Obviously if the models do not have the same number of free parameters then some models will only search a subset of the full search space nonetheless they will be embedded in the full space (See Section 3.8 for the chosen particle representation). PSO has been used to update one FEM in [69] and use to design an induction cooker in [70]. Recently it has been used for fault diagnosis of a turbo pump motor [71]. In [72] a variation of PSO is used to derive a set of optimal parameters for the pattern recognition classifier [54,55] that uses wavelet kernel functions. In [73] PSO is used as a wrapper [54,55] to reduce the number of measured variables in a dataset.

[3] The concept of FEM search space is a hypothetical one. All models are all searching for the best approximation of one real system even though they might have different updating parameters. The updating parameters represent the different positions in this space.

In [74] PSO has been used to design ultrasonic motors. In [75] PSO is used for image watermark extraction. With regards to FEM updating, the approach followed in this thesis is simpler and more informative than the approach followed in [69] where only one FE model was updated.

3.4 Graphical Interpretation

Figure 3.1 shows a populations of four potentially correct FE models; $M_A(\alpha_1,\alpha_2,\alpha_3,\alpha_4,\alpha_5)$, $M_B(\beta_1,\beta_2,\beta_3, \beta_4)$, $M_C(\lambda_1,\lambda_2,\lambda_3)$ and $M_D(\Phi_1,\Phi_2)$ to be updated. The algorithm is set to run over five iterations (I_1-I_5). Since model $M_A(\alpha_1,\alpha_2,\alpha_3,\alpha_4,\alpha_5)$ has the highest number of updating parameters the hypothetical search space has five dimensions. All models have to be expanded to have five dimensions thus the zeroes in the other four models. Initially model $M_A(\alpha_1,\alpha_2,\alpha_3,\alpha_4,\alpha_5)$ has the best approximation on the problem. The PSO algorithm marks this model as the global best. Other models are influenced to 'move' towards $M_A(\alpha_{11},\alpha_{21},\alpha_{31},\alpha_{41},\alpha_{51})$ thus their arrows point towards where this model is.

In the second iteration each model has updated its parameter values; this is noted by the iteration counter next to the parameter number subscript. After the second iteration model $M_C(\lambda_{12},\lambda_{22},\lambda_{32})$ has a better approximation accuracy. All models thus change and move towards this current global best model. This is called global or social behaviour. Similar to PBIL in the previous chapter there is some element of individual dynamism and the models do not fully move towards the leading model but also consider their previous best positions before they move. This is called local behaviour. This swarm behaviour has a tendency of focusing the search; previously spread out models are now concentrated on certain regions in the search space. After five algorithm iterations model $M_C(\lambda_{15},\lambda_{25},\lambda_{35})$ has the best accuracy on the problem.

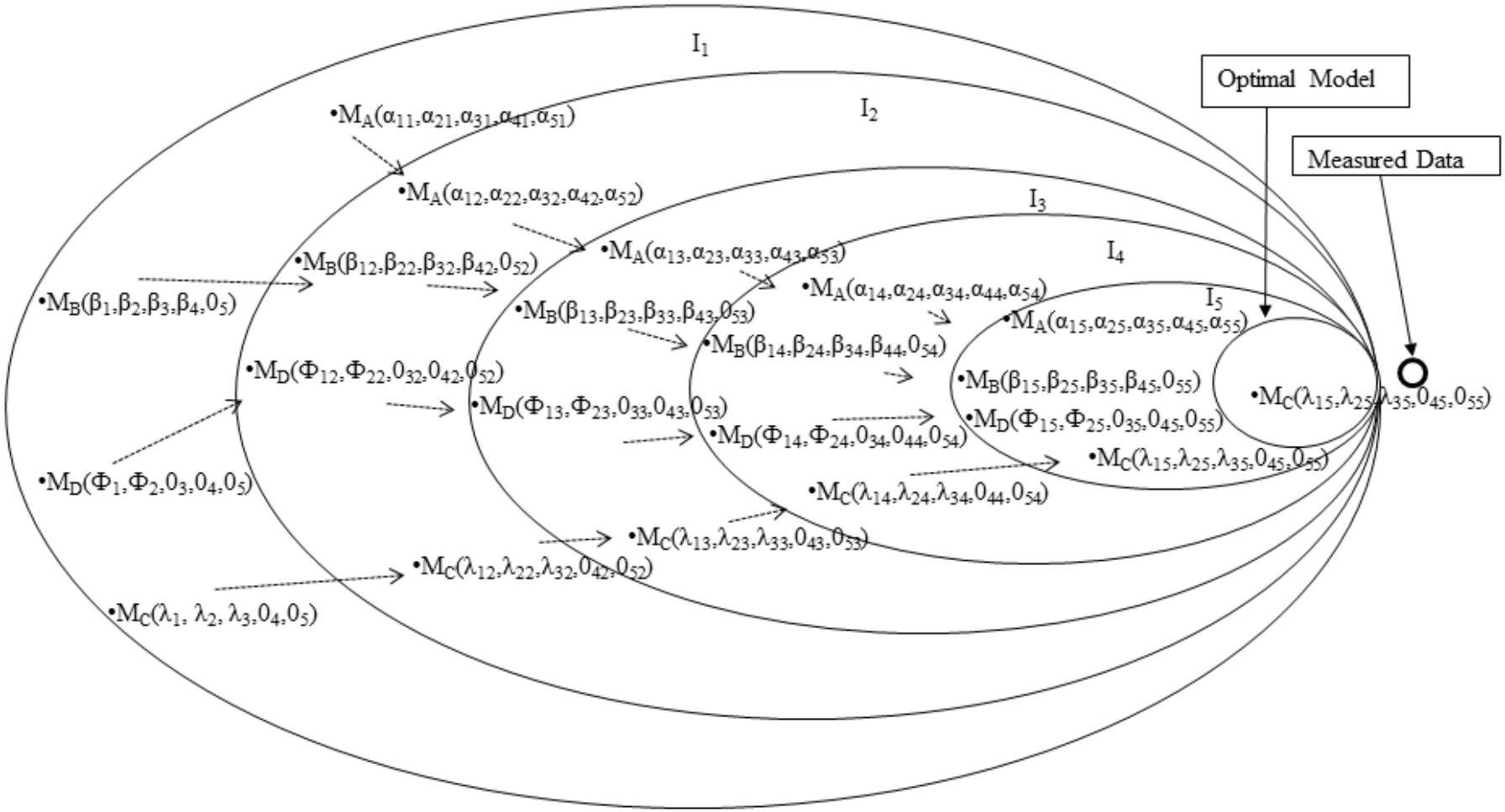


Figure 3.1 General PSO updating scheme

3.5 PSO Operators

In the PSO algorithm each particle is described by its vector position in the search space as in equation 3.1 below [66,68]:

$$m_i = \{m_{i1}, m_{i2}, m_{i3}, ...m_{id}\} \tag{3.1}$$

where i is an arbitrary particle and d is the problem dimension defined by the maximum number of updating parameters for the models. The particle position features (m_{id}'s) are the potential solution variables. To evaluate each particle on the problem one substitutes the particle position to the model/ function. The velocity of the i-th particle is represented by:

$$v_i = \{v_{i1}, v_{i2}, v_{i3}, ...v_{id}\} \tag{3.2}$$

Each particle also stores its (local) best ever position as it searches the problem space. This is represented by:

$$p_i = \{p_{i1}, p_{i2}, p_{i3}, ...p_{id}\} \tag{3.3}$$

The swarm also has a record of the best ever position by any particle, this is known as the global (socially) best solution which is represented by:

$$p_g = \{p_{g1}, p_{g2}, p_{g3}, ...p_{gd}\} \tag{3.4}$$

Each iteration of the PSO algorithm updates the position and velocity of each particle. The particle velocity is updated through the following equation [45,94]:

$$v_{ik}(t) = w_k v_k(t-1) + c_1 r_1 (p_{ik} - m_{ik}) + c_2 r_2 (p_{gk} - m_{ik}) \tag{3.5}$$

and the particle position is updated using equation 3.6 [45,94]:

$$m_{ik}(t) = m_{ik}(t-1) + v_{ik}(t) \tag{3.6}$$

where $\{i \in 1...m\}$ and $\{k \in 1...d\}$ which means each particle's position and velocity parameters /dimensions are updated on each iteration of the algorithm. See

Section 3.8.2 for the PSO algorithm pseudo code. In equation 3.5, c_1 and $c_2 \in \mathbb{R}$ are constants weighting that normally vary between 2 and 4 [66]. The random constants $r_1, r_2 \in U[0,1]$ introduce randomness to the search process. In order to prevent the tendency of the particle position and velocity to explode in magnitude, M_{max}, M_{min}, V_{max} and V_{min} are defined for each particle dimension.

Thus if

$$m_{ik} > M_{\max} \text{ then } m_{ik} = M_{\max}$$

$$m_{ik} < M_{\min} \text{ then } m_{ik} = M_{\min}$$

Similarly if

$$v_{ik} > V_{\max} \text{ then } v_{ik} = V_{\max}$$

$$v_{ik} < V_{\min} \text{ then } v_{ik} = V_{\min}$$

The setting of these limits on each dimension of the problem would depend on the analyst's understanding of the problem. The types of constraints are also very much dependent on the problem, for example some particle dimensions might be known to be constrained to be positive values, and in that case the absolute $|M_{max}|$ or $|V_{max}|$ might be applicable [66,68].

A modification to the original PSO algorithm by [67,76] introduced w_k, the inertia weight variable. This variable controls the influence of the previous velocity on the current velocity value. An adaptive inertia weight is often used to improve the algorithm's search from an initially explorative (global) search to a more local search as this variable decreases. This also has a tendency to improve the algorithm's convergence rate [66]. This variable is specified by the starting weight (w_{start}), w_f is the fraction of iterations over which the inertia weight is decreased and w_{end} the final inertia weight value The initial w_k in equation 3.5 is w_{start} and it is decrease by $w_k = w_k - w_{dec}$ where:

$$w_{dec} = \frac{w_{start} - w_{end}}{N - w_f} \tag{3.7}$$

from the first iteration up to iteration N x w_f, thereafter w_k is w_{end}. The (p_{ik}- m_{ik}) term in equation 3.5 measures how far each particle is currently from its personal best position (local) and (p_{gk}- m_{ik}) measures how far each particle is from the global (social) best particle in the swarm. This means the middle term in equation 3.5 tends to control the particle's velocity based on the particle's own best position while the last term allows the particle to be influenced by the best performing particle in the swarm.

In the next section we present the finite element models and propose a particular representation of these in the particle swarm context.

3.6 Modelled Structure H-beam

The finite element models updated in this chapter were all developed from the unsymmetrical H-beam previously used in [7,28]. The details of the modelled beam are described in the next section but more in detail in appendix A1.

3.6.1 Unsymmetrical H-Beam

A simple unsymmetrical H-beam shown in Figure 3.2 is modelled. This unsymmetrical H-beam is suspended on rubber bands The measured natural frequencies of interest of this structure occur at; 53.9Hz, 117.3Hz, 208.4Hz, 254Hz and 445Hz which correspond to modes 7, 8, 10, 11 and 13 respectively. The aluminium beam material has a Young's Modulus of 7.2×10^{10} Pa, the beam length is 600mm with a width of 32.2mm and a section thickness of 9.8 mm. The left edge has a length of 400mm, the right edge length 200mm and a density of 2700 kg/m^3.

Figure 3.2 shows that the beam is divided into elements numbered from one to twelve. Each finite element model used standard isotropic material properties and Euler Bernoulli beam elements to approximate the beam sections of the structure. The beam is free to move in all six degrees of freedom.

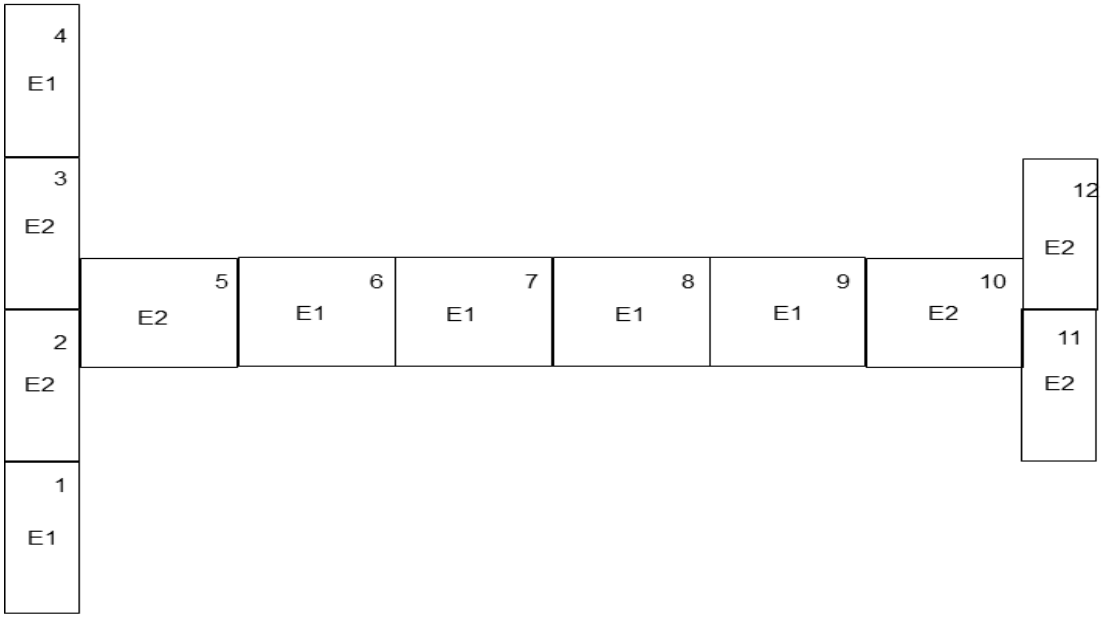


Figure 3.2 Model M_2 of a 12 element Unsymmetrical H-Beam

3.7 Candidate FE Models

All models in this example assume the only uncertain beam property is its Young's Modulus (E). To design different models of the beam, beam elements are grouped differently. The beam is modelled by eight competing models, m_i, i = 1...8. Model m_1 assumes the whole beam's Young's modulus is the updating parameter to be updated from the average given material value. Model m_2 has two parameter, E_1 and E_2; the elements numbered 1,4, 6,7,8,9 (all forming parameter E_1) are to be varied equally while elements 2, 3, 5, 10, 11, 12 (all E_2) are to be varied equally (see Figure 3.2 for the element numberings).

Model m_2 models the elements connected near the structural joints as one parameter and those away from the joints as another. Model m_8 assumes the left edge together with the first horizontal element, the horizontal section and right edge together with the last horizontal element are best updated differently, thus the three parameter arrangement. Table 3.1 lists the rest of the models and their parameterizations.

Table 3.1 Unsymmetrical H-Beam FEM parameters

Model Identity	No. of Model Parameters	Parameter Labels	Element grouping
m_1	1	E_1	{1-12}
m_2	2	E_1 & E_2	{1,4,6-9} & { 2,3,5,10-12}
m_3	3	E_1 E_2 E_3	{1,4,6-9}, {2,3,11,12} & {5,10}
m_4	4	E_1 E_2 E_3 E_4	{1,4,6-9}, {2,3} {11,12} & {5,10}
m_5	5	E_1 E_2 E_3 E_4 E_5	{1,4,6-9}, {2,3} {11,12},{5} & {10}
m_6	2	E_1 E_2	{1,2,3,4} & {5-12}
m_7	2	E_1 E_2	{1- 6} & { 7-12}
m_8	3	E_1 E_2 E_3	{1-5}, {6-9} & (10,11,12}

Perhaps the most important step in the implementation of the PSO algorithm is the choice of particle representation. This fundamentally dictates the problem search space and the ease of algorithm implementation. The next section presents the model representation adopted in the current finite element model updating procedure.

3.8 PSO Algorithm

3.8.1 Particle Representation

Each particle or finite element model ($m_{i...m}$) is described by the following E_i vector arrangement:

$m_1 = [E_1,0,0,0,0]$; $m_2 = [E_1,E_2,0,0,0]$; $m_3 = [E_1,E_2,E_3,0,0]$; $m_4 = [E_1,E_2,E_3,E_4,0]$;

$m_5 = [E_1,E_2,E_3,E_4,E_5]$; $m_6 = [E_1,E_2,0,0,0]$; $m_7 = [E_1,E_2,0,0,0]$; $m_8 = [E_1,E_2,E_3,0,0]$;

where in each case the updating parameter is sampled from a normal distribution defined as:

$$E_{i...5} = \mu + q\sqrt{0.5e20}\text{N.m}^{-2} \tag{3.8}$$

and the mean

$$\mu = 7.2x10^{10} Pa$$

The q variable samples random numbers from a normal distribution between $[-\infty, \infty]$. The parameter location or grouping of the models as described in Table 3.1 nullifies the concern that models m_2, m_6 and m_7 seem to be described by the same parameter vector. This choice of model representation sets the problem search space to five dimensions. Even though all the models search the five dimensional space, each is actually constrained to only searching a particular manifold of the space. This contextually means each subspace is assumed to have an optimum somewhere which the particle is supposed to find guided by individualistic and social performance. This means if model m_5 finds the best solution within the group (thus it is p_g) at some coordinate/parameter values, all the other particles will adapt towards model m_5's parameter values. In this particle representation, all the particle parameters will incrementally change towards model m_5's values, even the zeros in the particle vector description. But since, for example, model m_2, m_6 and m_7 are only dependent on the first and second updating values, it does not matter what happens to the zero features! Each model will somewhat also resist moving towards model m_5's coordinates by also incrementally moving towards their own previous best positions.

3.8.2 PSO Pseudo-Code

In this section the FEM-PSO pseudo-code algorithm together with the parameter settings used in the simulations are presented in Table 3.2. In Table 3.2 if the original PSO algorithm is implemented i.e. the inertia variable is eliminated, the last *If* statement condition is not executed and the velocity equation does not have w_k on the first term.

Table 3.2 The FEM-PSO Algorithm Pseudo Code.

% Set constants
I = 1000; *% The number of iterations*
$c_1 = c_2 = 2$; *% Individual and Group Influence*
m = 8; *% Number of potentially correct models*
$w_{start} = 1.2$; *% Initial inertia weight*
$w_f = 0.5$; *% Inertia decrement factor*
wend = 0.4; *% Final inertia weight*

%Initialise

w_{dec} *% Using equation 7.*
$m_{i..m}$ *% Randomly initialise the models*
$p_{i...m}$ *% Randomly initialise particle best solution e.g. $p_i=m_i$*

% Compute

$F(m_{i..m})$ - *% Calc FEM Fitness objective function in section 4.4*
Identify P_g & Start Iteration count =1

Repeat
while *iteration < N do*
for *all $m_{i..m}$ do*
for *each dimension d*
Calculate particle velocity(v_i) using equation 5
Update particle position (m_i) using equation 6
end *for*
Compute $F(m_i)$ & Update p_i if $m_i(t) > p_i(t-1)$
end *for*
Update p_g if any $p_i > p_g(t-1)$
if *iteration < $|N w_f|$*
$w_k = w_k - w_{dec}$
end *if*
iteration = iteration +1;
end *while*
return p_g

3.8.3 Model parameter constraints

In our FEMU problem the constraints placed on the particle velocity and position in the algorithm were as follows; the maximum parameter magnitude M_{max} for each dimension was set at 7.5e10 $N.m^{-2}$ and the minimum, M_{min}, was set at 6.5e10 $N.m^{-2}$. The maximum velocity magnitude was set to the difference between M_{max} and M_{min}. The minimum velocity was set to 1e9 $N.m^{-2}$. This means the velocity in this algorithm was tracing a factor of the standard deviation of the parameter values (i.e. the second term in equation 3.8) and the particle position was determining the mean parameter value (E_i) in equation 3.8.

3.9 Objective Functions

A number of fitness or objective functions are available in the scientific literature. In the FEMU problem Occam's razor is very much applicable. Occam's razor is also known as the law of parsimony and can be expressed as "Entities are not to be multiplied beyond necessity". Here these statements can be interpreted to say one seeks a model with the fewest updating parameters that will produce FE model results closest to measured lab results. In this paper we compare two objective/fitness functions; the Akaike Information Criterion (AIC) [60,61,68,76] and the Squared Sum of Errors (SSE). The AIC proposes that one should exchange the complexity of the model with its goodness of fit to the measured data [61,76]. The AIC function is given by the following equation [68]:

$$AIC = n\log(\sigma^2) + 2d \tag{3.9}$$

where,

$$\sigma^2 = \frac{\sum_{i=1}^{n}(msrd_{data} - fem_{results})^2}{n} \tag{3.10}$$

(also known as the standard deviation) and d is the number of model parameters, n is the number of measured modes, $msrd_{data}$ is the measured data and $fem_{results}$ are the finite element model results.

The squared sum of errors is given by

$$SSE = \frac{\sum_{i=1}^{n} (msrd_{data} - fem_{results})^2}{2} \tag{3.11}$$

As it can be seen from equations 3.9 to 3.11, the first term of equation 3.9 is effectively the SSE and is commonly referred to as the data-fit term [51,77,78,79] and the second term is known as the model complexity penalty term. This assumes a model's complexity is determined by the number of free variables. This is not always the correct way to define model complexity as argued in [77,80]. In [80] it is argued that there is not necessarily a relationship between number for parameters and model complexity. The observed data can determine the model complexity. A complex model can have few parameters.

The minimization/maximization of the squared sum of error objective function does not account for model complexity but only measures how well a given model fits the data. In [78] it is argued that model selection should not only be based on data-fit term. We would expect the implementation of the AIC function as the objective function in the PSO algorithm to be biased to models with fewer parameters.

3.10 Simulation Results

The simulations presented in this section are all modelled using Version 6.0 of the Structural Dynamics Toolbox (SDT®) for MATLAB®. A number of simulations were run using different setting of the PSO algorithm parameters. Initially the number of iterations in all settings was set to I =1000 but it was found that the algorithm consistently converged before Max = 500 iterations. In each of the experiments the convergence figures will only focus on the main convergence part of the graph, where necessary the figure will be expanded to show 500 iterations.

Table 3.3 PSO simulation parameter settings

PSO Parameter	Simulation No.1	Simulation No. 2	Simulation No. 3	Simulation No. 4
C_1 Local influence	2	2	2	2
C_2 Global Influence	2	2	2	2
w	0	0	Adaptive	Adaptive
Objective function	AIC	SSE	AIC	SSE

3.10.1 Simulation number 1

In this simulation the original PSO algorithm is implemented on the FEM updating problem, this means there is no inertia variable as shown in Table 3.3. Figure 3.3 shows the convergence plot of the AIC objective function over the 200 iterations of the algorithm. Figure 3.4 shows the variation of the best particle (p_g) in the swarm (global best model - GBM) over 10 iterations

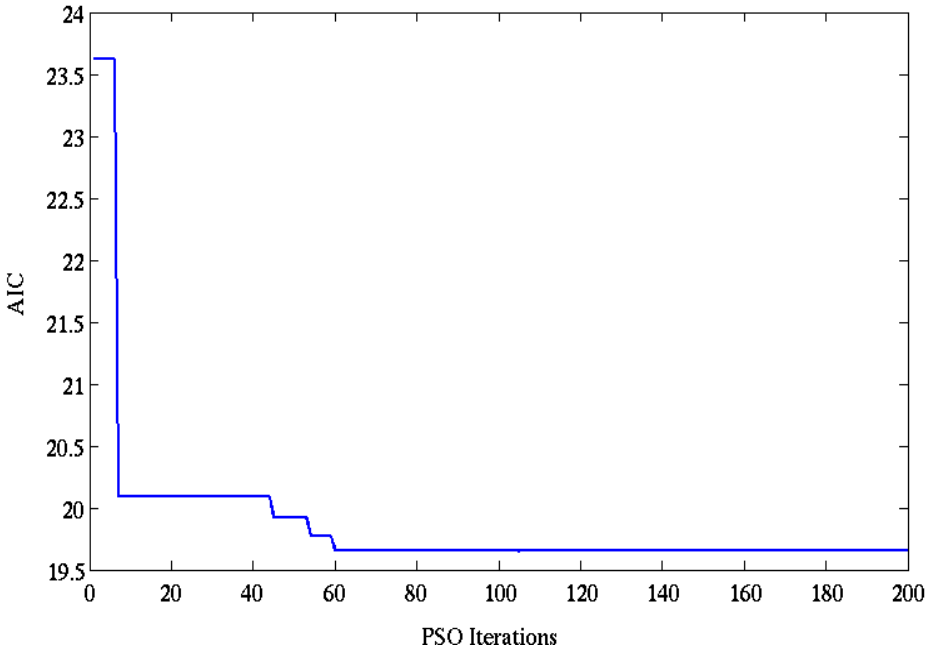


Figure 3.3 AIC convergence vs. PSO iterations

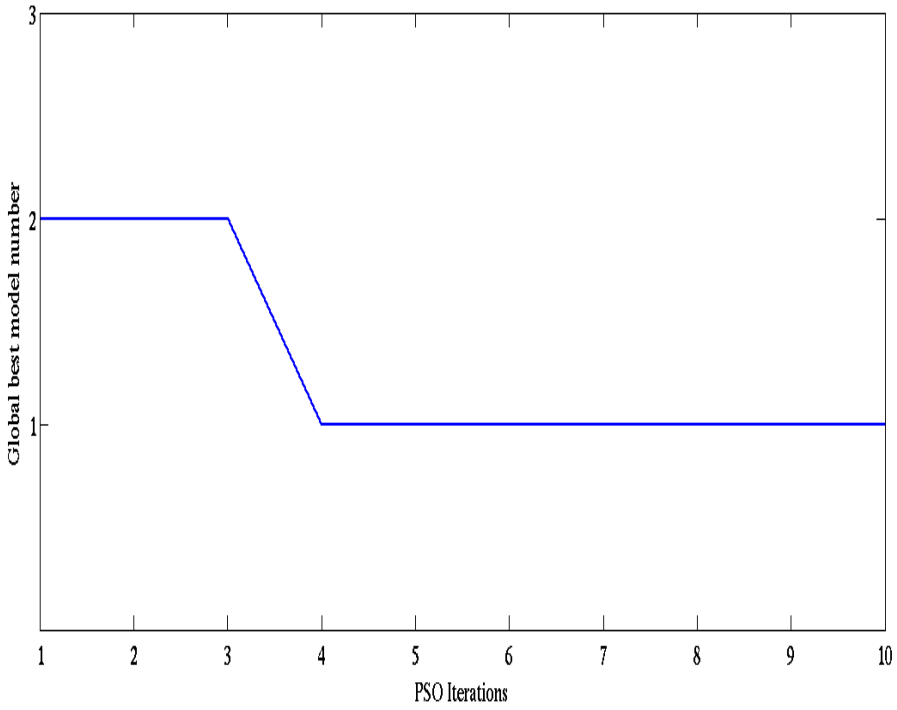


Figure 3.4 AIC GBM number vs. PSO iterations

Figure 3.3 illustrates that the PSO algorithm rapidly converged close to the ultimate minimum error within the first 70 algorithm iterations. The global best model in this simulation started off as model m_2 but after 3 iterations it changed to model m_1 and remains unchanged for the rest of the simulation. The model order, based on the minimum of the objective function for these PSO settings was m_1, m_6, m_2, m_7, m_8, m_3, m_4 and then m_5.

3.10.2 Simulation number 2

This simulation is the same as simulation 1 except the objective function has been changed to SSE. Figure 3.5 shows the convergence plot of the SSE objective function over the first 100 algorithm iterations. Figure 3.6 illustrates the convergence behaviour of global best model over 100 iterations.

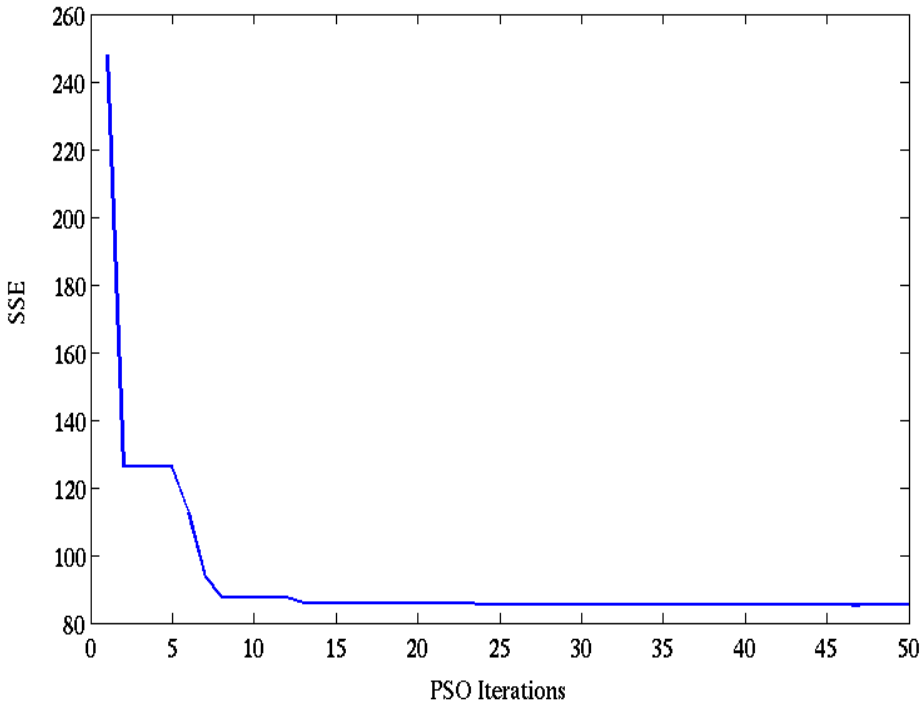


Figure 3.5 SSE convergence vs. PSO iterations

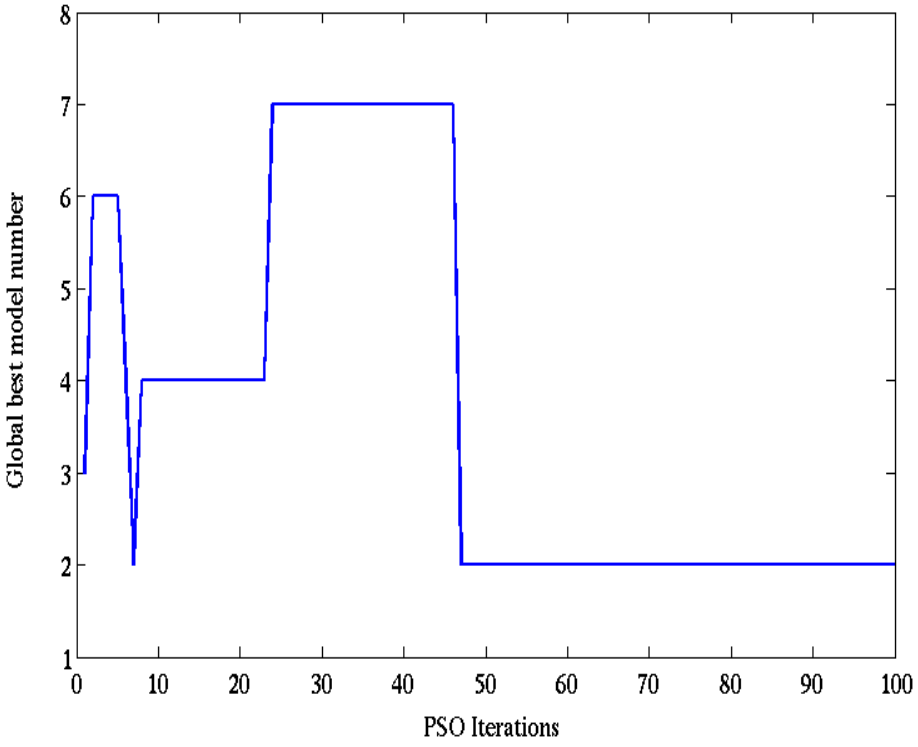


Figure 3.6 SSE GBM number vs. PSO iterations

The objective function (in Figure 3.5) did not improve much after 100 iterations even though the global model changed. A relatively small improvement occurred (not shown) due to model m_6 becoming the global best model after 300 iterations. It is clear from Figure 3.6 that the objective function had a significant role in the updating of the model parameters. Initially model m_3 was the p_g then the global best changed to being m_6, m_2, m_4, m_7, m_2 then finally model m_6 was again the p_g (not shown). The final objective function based model order in this simulation was m_6, m_1, m_2, m_5, m_4, m_8, m_3 and then m_7. This is different to the results in simulation 1 where the less complex models attained lowest errors.

3.10.3 Simulation number 3

In simulation number 3 the PSO algorithm has been changed by introducing the adaptive inertia weight. As mentioned in Section 3.5 the inertia parameter allows the algorithm to initially explore a wider search area and near the end to exploit the local search space. This is evident from Figure 3.7, where the search

converged much earlier in the iteration count than in figure 3.3 whilst using the same objective function. The weighted AIC error also decreased with more units than the un-weighted function in Figure 3.3.

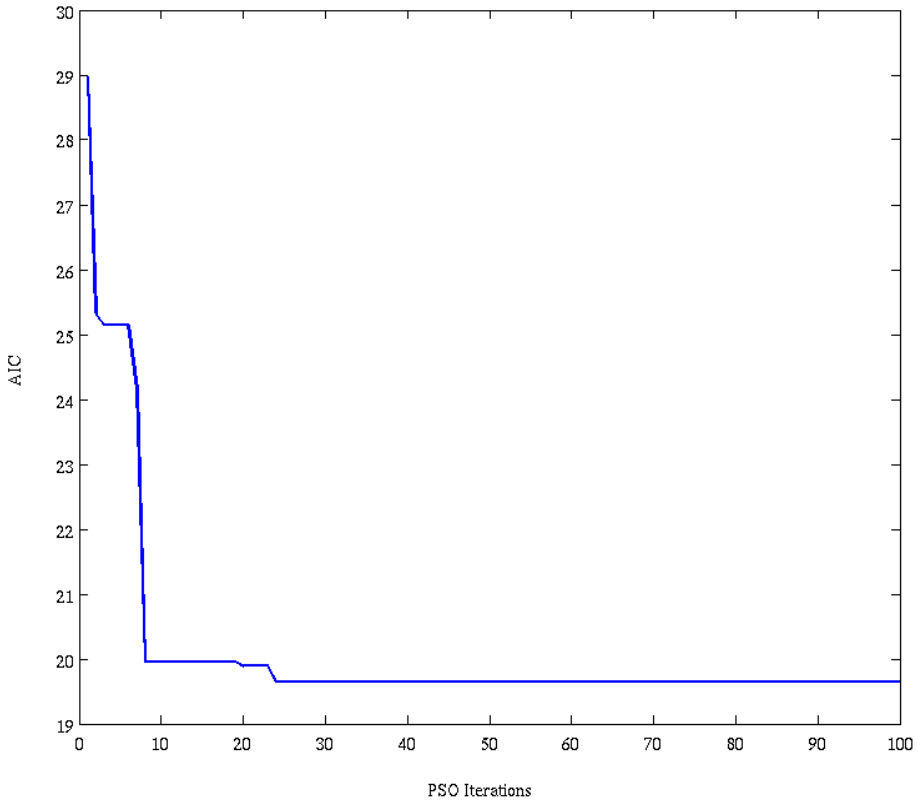


Figure 3.7 Weighted – AIC convergence vs. PSO iterations

Figure 3.8 also supports the initial exploration to local exploitation concept because different models were initially the global best, p_g, as opposed to the case in Figure 3.4 but later on in the algorithm a firm favourite was converged on.

The final model order in this simulation was m_1, m_6, m_7, m_2, m_3, m_8, m_4 and then m_5. Perhaps the AIC objective function is too critical of the model complexity as it seems to always select models according to it.

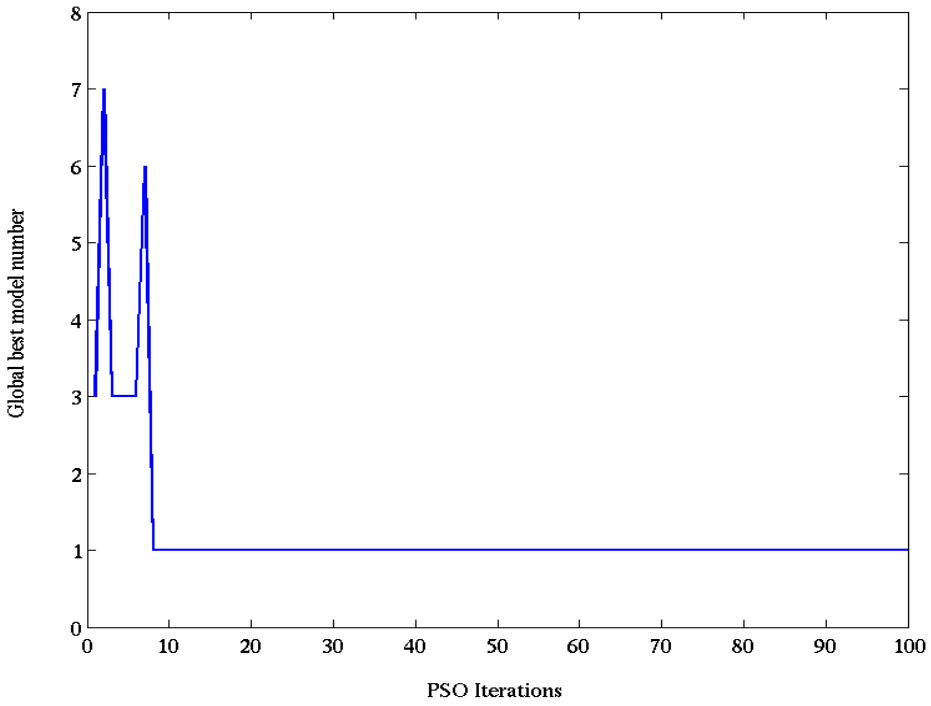


Figure 3.8 Weighted – AIC GBM numbers vs. PSO iterations

3.10.4 Simulation number 4

In simulation number 4 the PSO algorithm has the adaptive inertia weight but uses the SSE as the objective function. Figures 3.9 – 3.10 show the results of this simulation. The inertia adaptive weight does not seem to be that influential under the SSE objective function. The adaptive inertia weight has the opposite effect to the SSE as to the AIC function.

The final model order in this simulation was m_4, m_1, m_3, m_6, m_2, m_7, m_5 and then m_8. Clearly the SSE objective function is not concerned with the complexity of the finite element model. The best model in this case is the most complex. It is not easy to directly compare the objective function magnitudes the algorithms converge to. This would have allowed for better analysis of why different models behave so differently under different objective functions. More conclusive decisions on the choice of objective function in this type of updating methodology can only be made with further experiments on different types of objective functions.

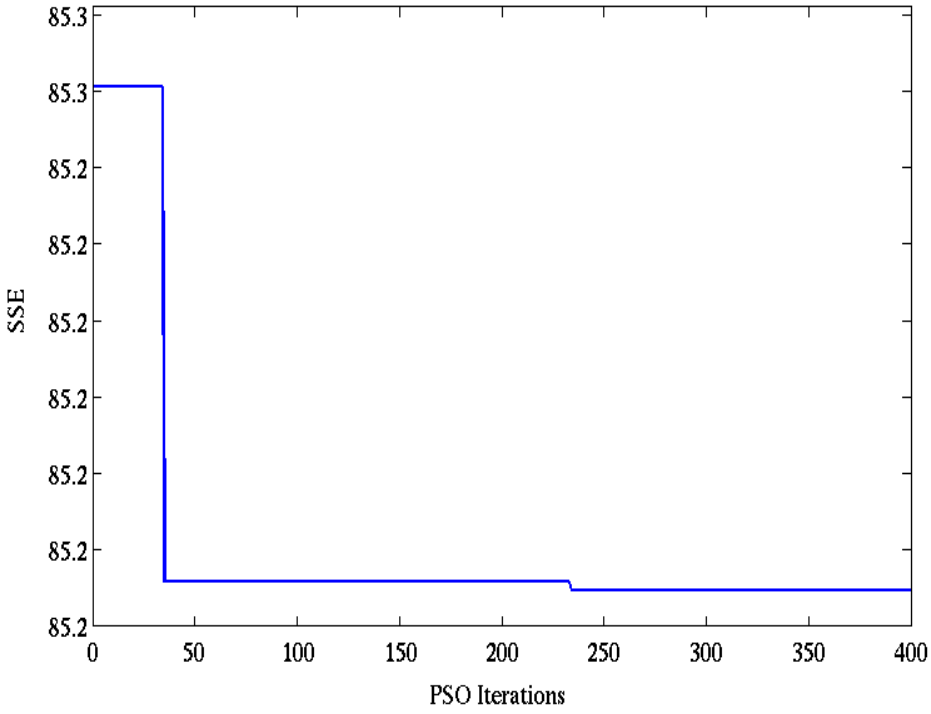


Figure 3.9 Weighted – SSE convergence vs. PSO iterations

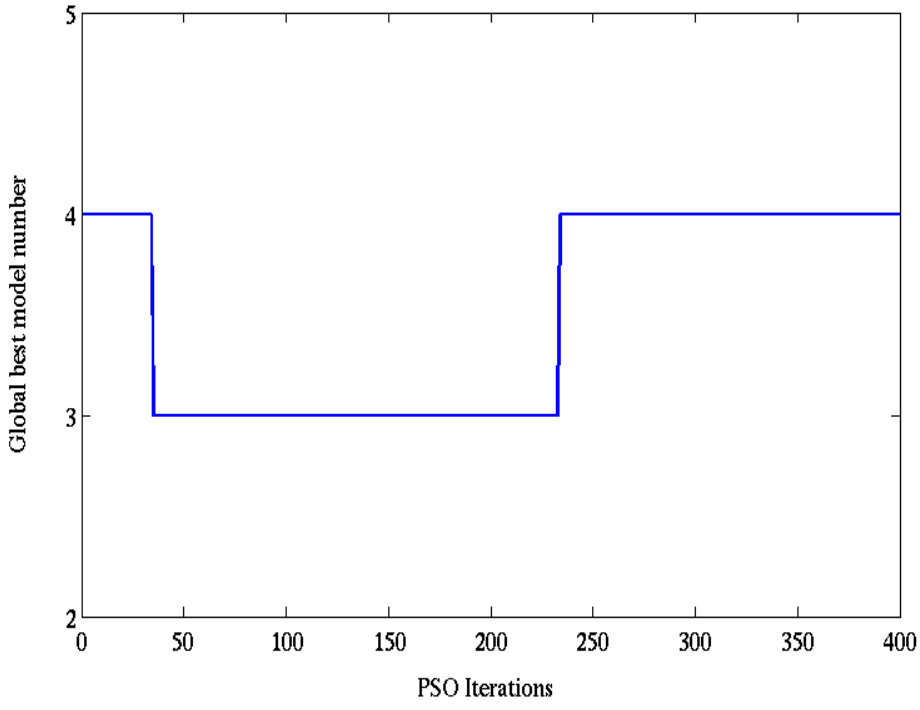


Figure 3.10 Weighted –SSE GBM number vs. PSO iterations

3.11 Chapter Summary

To summarize, we have argued that FEM updating should be performed in a multiple model framework. This is not only convenient but necessary as no FE model can be designed in isolation. A particle swarm based method of FEM updating and selection is proposed to implement this framework. The method, PSO, updates FEM parameters using a stochastic-population based procedure. Each potentially correct model of a structure is treated as an adaptive particle in the FEM updating problem space. This space is defined by the number of potentially updatable model parameters.

A number of simulations, using two different objective functions, are performed on eight competing FE models of a particular structure. The particle swarm based optimization approach to FEM updating offers the researcher an ability to simultaneously update and select the best model in a given group. One attractive aspect of this updating procedure is that the updating of the models affects each other. This means each candidate model has the potential to change its accuracy position as the group updating is performed.

An important aspect in using PSO for model updating is the choice of fitness function. This function is crucial in obtaining a reasonable best model. The limitation of the PSO updating procedure implemented in this chapter is that one cannot qualify the significance of the difference between the updated models. PSO only provides the model accuracy ordering. This limitation is addressed in the next chapter.

CHAPTER 4

Bayesian Model Selection and Updating

How to establish that your model is the best?

There are multitudes of FEM updating procedures in this field so one can potentially update a particular model in a number of ways. Furthermore there is no consensus on the best method or a way of objectively comparing such updating methods and indeed the updated FE models themselves. This is a fundamental problem faced by any FEM updating practitioner. As a first step this chapter attempts to address the objective comparison measure for updated FE models. This is essentially a model selection problem and Bayes' theorem is well suited to quantify model selection probabilities.

4.1 Introduction

Having addressed the first two FEM updating challenge of:

- Which aspects of the model need to be updated and which are actually updated?
- And proposed an efficient FE model updating scheme

The pertinent question becomes - With what certainty can we guarantee that our updated model is the best one? This we propose ought to be an essential and necessary statistic to establish before any FEM updating exercise can be considered complete.

Having accepted that the most probable model can best be established from a set of candidate models we can proceed to the model selection problem. In this regard we propose using the Bayesian evidence statistic to assess the probability of each updated model. This Bayesian measure makes it possible to evaluate the need for alternative updating parameters in the updating of any initial FE model. The model evidences are compared using the Bayes factor, which is the ratio of evidences. Jeffrey's scale is then used to quantify the significance in the model evidence differences. The Bayesian evidence is calculated by integrating the likelihood of the data given the model and its parameters over the a priori model parameter space using the Nested Sampling (NS) algorithm. This algorithm samples the likelihood distribution by using a hard likelihood-value constraint on the sampling region while providing the posterior samples of the updated model parameters as a by-product.

4.2 Chapter outline

The next section introduces the Bayesian inference approach in the context of finite element model updating. Section 4.4 expands on the marginal likelihood function and its interpretation as the evidence measure. Section 4.5 introduces nested sampling. Section 4.6 and 4.7 introduce Bayesian model selection and the FE models used in the experiments. Section 4.8 summarizes the chapter.

4.3 Bayesian Inference

Bayesian inference [26,51,77,81,82] allows one to quantify uncertainties in quantities of interest in a formal way. Bayesian inference is often implemented in two settings; parameter estimation [83,84,85] and model selection [78,86,87,88]. In [83] Bayesian inference is used to estimate Markov random field parameters for image reconstruction. In [84] it is used to estimate the parameters of

probability distributions. In [28] the parameter estimation context of the Bayesian framework was implemented to obtain the posterior probability distributions of FEM parameters. By obtaining the posterior probability of the updating parameters the probability distribution of the modal properties may be calculated. This gives the researcher a quantitative measure of the confidence intervals of the modal data and updating parameters of the model.

In general parameter estimation is concerned with the plausibility of a given model's parameters based on some observed measurements and this is often carried out using standard Bayes theorem and/or sampling methods e.g. Markov Chain Monte Carlo (MCMC) [77,89,90]. In contrast, model selection deals with the mathematical hypothesis of the ability of the model(s) to approximate a particular observed/measured quantity.

4.3.1 Parameter Estimation

In parameter estimation the mathematical model (in our case a particular FE model) is often assumed to be true. The model is then allowed to approximate the measured data and the plausibility of the model parameters can be inferred from their posterior probability. This probability is calculated via Bayes theorem as:

$$P(\theta \mid D,H) = \frac{P(D \mid \theta,H)\,P(\theta \mid H)}{P(D \mid H)} \tag{4.1}$$

where the left hand side is the posterior probability of the updating parameters for the true model H, given some data D, θ are the updating parameter(s). The prior probability of the model parameters is $P(\theta,H)$ and $P(D \mid \theta,H)$ is the likelihood of the model. The denominator $P(D \mid H)$ is called the marginal likelihood, or the evidence [51,77,80,82,91,92] of the model where the parameters have been marginalized out. Bayes theorem automatically incorporates the updating of the parameters by definition; it updates the prior probability distribution of the model parameter values with the likelihood (under the assumed parameters) of the model approximating the measured data. The likelihood function is the difference

between the model (using the assumed updating parameters) and the real system data. The updated parameter values are then revealed in the posterior probability distribution. This probability quantifies the plausibility of the chosen parameters to approximate the observed data. Bayes theorem assumes we have some a priori knowledge of the model parameter value distribution or the possible variation of the parameter values.

4.3.2 Model Comparison/Selection

In model selection a number of candidate finite element models for a system are formulated and compared to determine which model best approximates the measured data. In general, Bayesian inference provides a platform to evaluate which model(s) is (are) the most probable for given measured dataset(s).

In this thesis one measured data set is observed and the finite element model that best approximates this data is determined. The posterior probability of each model within a set of plausible models is given by Bayes theorem:

$$P(H_i \mid D) = \frac{P(D \mid H_i) P(H_i)}{P(D)} \tag{4.2}$$

where $P(H_i)$ is the prior probability of each model and $P(D)$ is the probability of the data. Since the denominator is independent of the models, we may neglect its influence so that Bayes theorem reduces to:

$$P(H_i \mid D) \propto P(D \mid H_i) P(H_i) \tag{4.3}$$

The first term on the right side of equation 4.3 is the denominator term (or evidence) from equation 4.1. On the assumption that each designed model is equally likely to match the data, the evidence term is then the deciding factor on which model is the most probable for a particular observed dataset. Model selection for FEM has been presented in [25,79]. In [79] the posterior distribution of the model was assumed to be a normal distribution, which is known to work for certain types of models [51,77]. The updating method proposed in this paper does

not place any prior assumption on the form of the posterior distribution of the model but has recently [89] been shown to perform better for normally distributed functions. In [79] a recently proposed MCMC type posterior probability sampling algorithm (TMCMC from [89]) was implemented. This algorithm estimates the model evidence by sampling the posterior probability distribution of the model by a sequence of non-normalized intermediate probability functions. This algorithm uses a number of free tuning parameters; namely a variable to balance the sampling steps, the number of intermediate probability distribution functions (PDF), the tempering parameter and the control parameter [79,89]. The algorithm proposed in this work is believed to be simpler to implement.

In the next section we explain the concept of Bayesian evidence. We then introduce an algorithm called Nested Sampling [93] to efficiently estimate the model evidence.

4.4 Bayesian Evidence

Equation 4.1 shows that the evidence term is a normalizing factor and can be written as [93]:

$$evidence = P(D \mid H_i) = \int P(D \mid \theta, H_i)\, P(\theta \mid H_i) d\theta \tag{4.4}$$

This equation may be interpreted as follows; given a unit of parameter space $d\theta$ with a model having a prior parameter probability of $P(\theta \mid H_i)$ over this space, the posterior probability distribution of the model over the same space will depend on how well the model with these parameters approximates the data $P(D \mid \theta, H_i)$. For the posterior distribution, the evidence is proportional to the two terms of the integrand in equation 4.4. The first term is the data- approximation (likelihood) component $P(D \mid \theta_{mp}, H_i)$ due to the most probable model parameter values, θ_{mp}.

4.4.1 The Likelihood function

We assume the likelihood function to be a normal function, with the function exponential E given by the difference between the measured and analytical natural frequencies at the measured modes. Thus this likelihood can be written as [28,38,77,81]:

$$P(D \mid \theta, H_i) = \frac{1}{(2\pi\sigma^2)^{(N_p N_m)/2}} \exp\left[-\frac{1}{2\sigma^2} \sum_{i}^{N_p} \sum_{j}^{N_m} \left(E^2\right) \right] \tag{4.5}$$

where E is the error matrix. Subscript j represents the jth modal property and i represents the ith measurement position on the structure. N_p represents the number of measured mode-shape coordinates and N_m is the number of measured modes. The uncertain error variance at each measured position on the structure is σ, and represents the uncertainty between the measured and analytical outputs.

4.4.2 Information Gain

The second term in equation 4.4, $P(\theta \mid H_i)d\theta$, is the model complexity penalty term (or the information gain). This information gain relates the prior probability of the model updating parameter values to the posterior distribution of the updated parameter values. In the posterior distribution the most probable updating parameter values occupy a smaller peaked region than in their initial prior distribution. This change factor is equal to the fraction of the posterior parameter space to the prior parameter space [51,77,90] and effectively measures how much information the model has extracted from the data. The information gain measure can be calculated as [90]:

$$I = -\int \ln\left[\frac{p(\theta_j \mid D, H_i)}{p(\theta_j \mid H_i)} \right] p(\theta_j \mid D, H_i) d\theta_j \tag{4.6}$$

equation 4.6 clearly shows that if the posterior parameter distribution is a substantially small fraction of an initially large prior distribution space, as is the case for complex models, then the natural logarithm of that ratio becomes large. This is the factor that penalizes the model complexity in the evidence calculation. The bigger the value of this factor the more information needs to be extracted from the measured data to update the suggested model parameters. The complexity of the model influences the value of the information gain and therefore affects the value of the Bayesian evidence.

4.4.3 The Evidence Reformulated

The evidence formulation of equation (4.4) can be re-written as [93,95,96]

$$Z = \int L(\theta)\ \pi(\theta)\, d\theta \tag{4.7}$$

where $L(\theta)$ is the likelihood and $\pi(\theta)$ is the prior distribution. Analytically evaluating this integral may be difficult or impossible if the product of the prior and likelihood is not simple. This often happens when, for example, the parameter space has high dimension, which requires the calculation of multidimensional likelihoods.

The most popular approach to approximate such integrals is to apply numerical techniques such as importance sampling and thermodynamic integration [32,61] although these encounter the problem that the prior parameters are distributed in regions where the likelihood function is not highly concentrated. Other sampling techniques, for example TMCMC [21,94] that are based on the Markov Chain Monte Carlo (MCMC) paradigm can be used, but these tend to only sample the peaks of the posterior distribution which can under-sample most of the narrow best- approximation regions [32]. Recently Skilling [93,95,96] proposed the nested sampling algorithm, which is able to efficiently estimate integrals of the form shown in equation 4.7. The algorithm works by transforming the multidimensional parameter space integral into a one dimensional integral where classical numerical approximation techniques to estimate the area under a function can be applied.

The astronomy and cosmology community [85,87,97,98] has been quick to adopt and successfully apply the nested sampling algorithm for quite a range of observed data and models. In [91] the Bayesian evidence framework is used to identify optimal damage features in spectral data. The Bayesian evidence has also been used doe financial model selection [92].

4.5 Sampling

4.5.1 Nested sampling

Nested sampling is a Monte Carlo, but not a Markov Chain, sampling method. The main idea behind the method is to divide the prior parameter space into 'equal mass' units and to order these by model likelihood. The total prior mass is denoted by X and each unit in this prior mass is $P(\theta \mid H)\ d\theta = \pi(\theta) d\theta = dX$. The likelihood function is written as [95]:

$$P(D \mid \theta, H) = L(\theta) = L(X) \tag{4.8}$$

in this space. This formulation transforms the likelihood into a function of a single parameter. The evidence integral from equation 4.6 then becomes:

$$evidence = Z = \int L(\theta)\ \pi(\theta)\, d\theta = \int_0^1 L\, dX \tag{4.9}$$

The algorithm supposes that the likelihood can be evaluated at all X_i such that $L_i = L(X_i)$, where X_i is a sequence of values that decreases from 1 to 0 such that $0 < X_N < ..... < X_2 < X_1 < X_0 = 1$ as illustrated on the right image of Figure 4.1.a. The left image shows 10 samples and the likelihood iso-contours in the 2-dimensional posterior probability parameter space. Figure 4.1.b shows these samples ordered by their likelihood in the 1D space. The red line shows the second highest likelihood sample in both spaces.The one dimensional integral function in equation 4.9 is easily estimated by any numerical method. For example using the trapezoid rule gives:

$$Z = \sum_{i=1}^{N} Z_i \tag{4.10}$$

where

$$Z_i = L_i b_i\,, \quad b_i = \frac{1}{2}(X_i - X_{i+1})$$

and L_i is the likelihood at that sample.

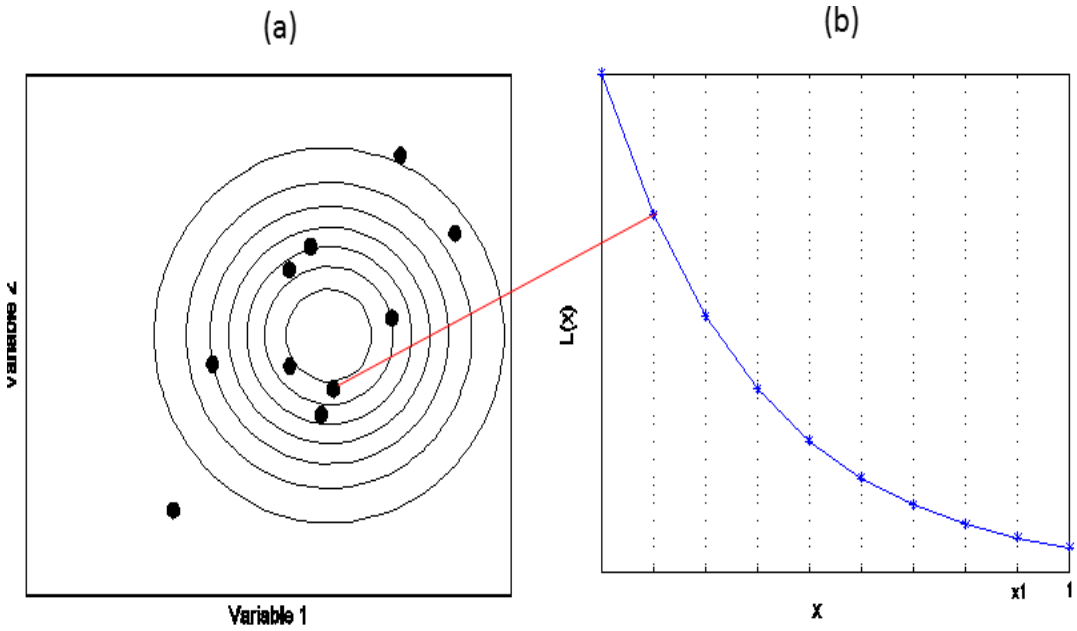


Figure 4.1 a) N_S =10 in 2D Iso- Contour. b) Likelihood ordering

In the context of finite element model updating, the algorithm achieves this approximation in the following manner:

1. Sample *N* updating parameter values (e.g. Young's Modulus values) from some prior probability distribution (based on the assumed distribution of the Young's modulus values in the structure). Evaluate their likelihoods.
2. From the N_S samples select the sample that results in the lowest likelihood, say $L_{i=1}$.
3. Increment the evidence by some summation rule, e.g. $Z_i = \frac{L_i}{2}(X_i - X_{i+1})$ for the trapezoid rule.
4. Discard the sample with the lowest likelihood (L_1) in the original *N* samples. Replace it with a new sample drawn uniformly from the prior parameter distribution within the remaining prior volume $[0, X_1]$. This new sample must satisfy the constraint that its likelihood is larger than the discarded sample's likelihood ($L_{new} > L_1$).
5. Repeat steps 2-4 until some stopping criterion is reached. This could be the desired precision on the evidence or some iteration count.

For further details and advances on nested sampling see [94,96,99]. The next section presents the experiments performed on two structures and introduces the candidate finite element models, together with the evaluation of their evidences.

4.6 Model Definition and Comparison

In FEM a mathematical model (H_i in Section 3) is defined by the identity and location of the updating parameters. In the first example models were designed to have a different number of free (updating) parameters at different positions along the same structure. The number of free-parameters in a model is one measure of the complexity of that mathematical model, although this has recently been questioned in the context of the evidence of a complex model [80].

Classically, complex models tend to over-approximate the data but in the Bayesian context this is not necessarily the case. In this paper we also wish to determine the relationship between the complexity of the finite element model and how well it matches the measured data, and, according to Occam's razor, the model with fewer parameters is preferred. In the Bayesian formulation of the model updating problem this condition is implicit. Jeffrey's scale [82,90,100,101] is often used to determine the significance of the difference between model evidences. This measure is calculated from the ratio of model evidences, also known as Bayes Factors. Table 4.1 shows the interpretation of this measure used in this paper.

4.7 Applications of Bayesian Evidence

To investigate the applicability of the Bayesian evidence statistic to quantify and qualify different models we provide two examples. The first example considers a simple unsymmetrical H-beam structure used in chapter three and the second example considers the more complex GARTEUR SM-AG19 structure used in Chapter 2. In both examples we use measured data from real physical experiments.

Table 4.1 Significance ratios used in Jeffrey's scale

Z_j / Z_i	$\log_2(Z_j / Z_i)$	$\log_e(Z_j / Z_i)$	$\log_{10}(Z_j / Z_i)$	Evidence against H_i
1 to 3.2	0 to 1.7	0 to 1.2	0 to 0.5	Weak
3.2 to 10	1.7 to 3.3	1.2 to 2.3	0.5 to 1	Substantial
10 to 100	3.3 to 6.6	2.3 to 4.6	1 to 2	Strong
> 100	> 6.6	> 4.6	> 2	Decisive
> 1000	>10	>7	>3	Beyond Reasonable Doubt

4.7.1 Example 1: Unsymmetrical H-Beam

This structure is described in detail in Appendix A1 as before. To validate that the model evidence calculation can reveal the most plausible finite element model, eight illustrative models of the unsymmetrical beam are developed and the evidence of each is calculated. The objective is then to determine from the evidence, the most probable model from this set.

4.7.1.1 Candidate Mathematical Models

All models in this example assume the only uncertain property of the beam is its Young's modulus (E) value. Beam elements are grouped differently to form a particular number of parameters for each model. The beam is modelled by eight competing models, H_i, i = 1...8. Model H_1 assumes the whole beam's Young's modulus is the variable to be updated from the initial standard material value.

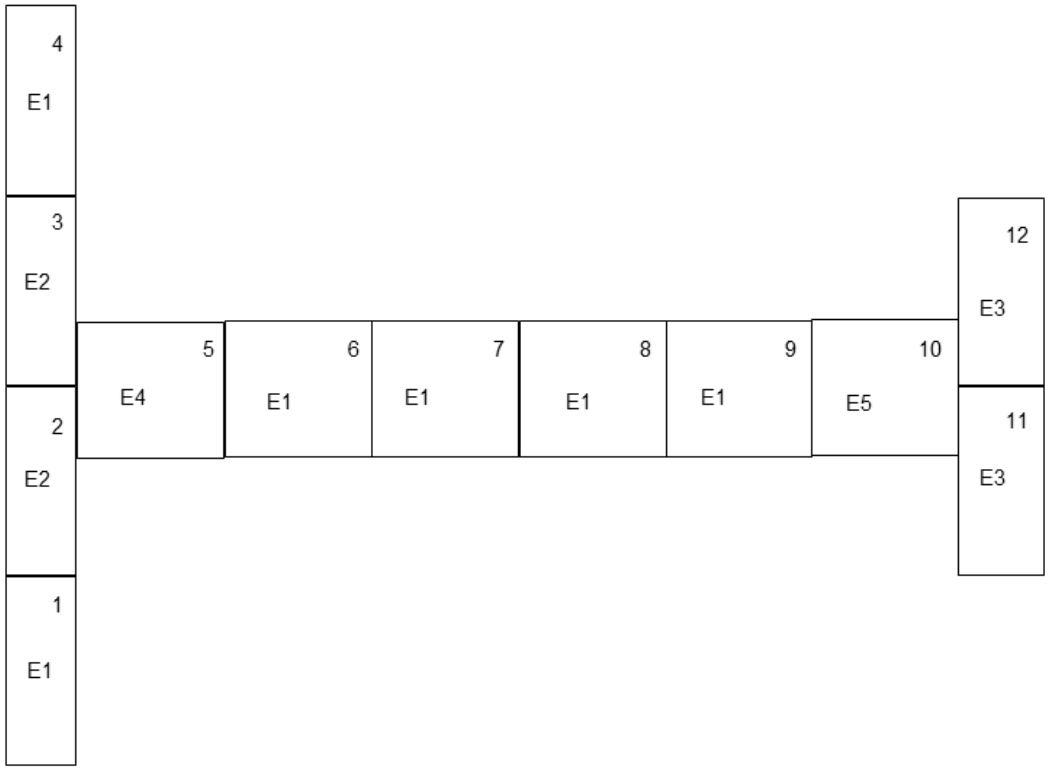


Figure 4.2 Model H_5 of a 12 element Unsymmetrical H-Beam

This means model H_1 has one parameter, E. Model H_5 has five parameters, E_1,E_2,E_3,E_4,E_5 the elements numbered 1, 4, 6-9 are grouped together to form parameter E_1, while elements numbered 2, 3 form E_2, elements numbered 11,12 form E_3 , element number 5 forms E_4 and element 10 forms E_5. The design of model H_6 assumes the dynamics of the whole left edge of the beam is different from the rest of the structure and this is modelled by grouping the first four elements together (to form E_1) and grouping the last eight in another parameter (E_2). Model H_8 assumes the left edge together with the first horizontal element, the horizontal section and right edge together with the last horizontal element are different and thus the model has three parameters. For completeness Table 4.2 lists the rest of the models and their parameterizations.

Table 4.2 Model parameters and labels for eight H-Beam FEMs

Model Identity	No. of Model Parameters	Parameter Labels	Element grouping
m_1	1	E_1	{1-12}
m_2	2	E_1 & E_2	{1,4,6-9} & { 2,3,5,10-12}
m_3	3	E_1 E_2 E_3	{1,4,6-9}, {2,3,11,12} & {5,10}
m_4	4	E_1 E_2 E_3 E_4	{1,4,6-9}, {2,3} {11,12} & {5,10}
m_5	5	E_1 E_2 E_3 E_4 E_5	{1,4,6-9}, {2,3} {11,12},{5} & {10}
m_6	2	E_1 E_2	{1,2,3,4} & {5-12}
m_7	2	E_1 E_2	{1- 6} & { 7-12}
m_8	3	E_1 E_2 E_3	{1-5}, {6-9} & (10,11,12}

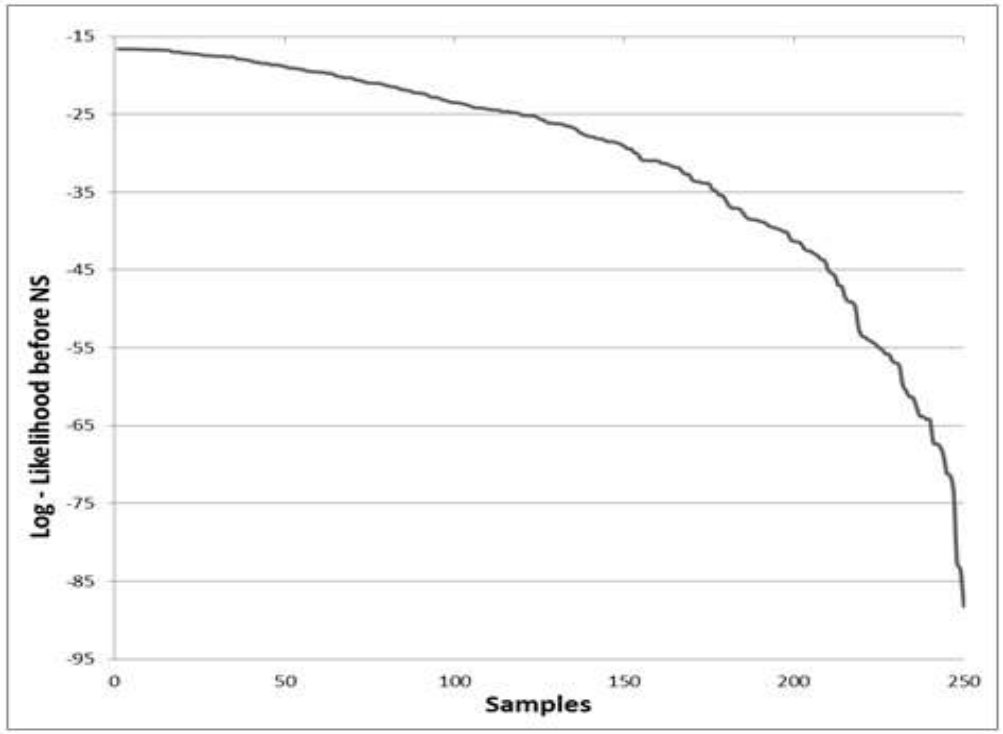
4.7.1.2 Prior Model and Nested Sampling Parameters

Each model was run through the stochastic nested sampling algorithm three times. The number of Young's modulus updating parameters sampled for each model was set to N=250 and the nested sampling algorithm's stopping criterion is set to a maximum of 300 iterations. The exploration for a new sample, point number 4 in Section 4.1, in the nested sampling algorithm had 20 MCMC steps. The updating variables were assumed to be identical but independent and were sampled from a normal prior distribution $P(\theta|H_i)$ with a mean of $7.2\text{x}10^{10}$ N.m^{-2} and a variance of approximately $0.55\text{x}10^{20}$.

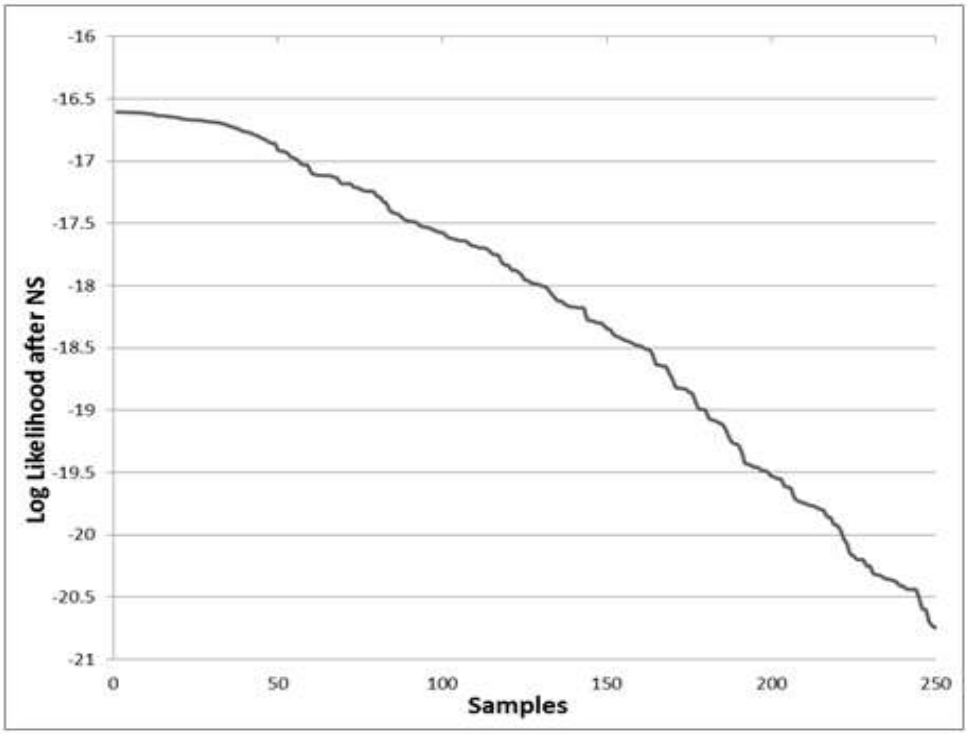
The inverse standard deviation of the error probability or likelihood function, $R(D|\theta,H_i)$, is the error between successive measured and initial model modal properties. It is set to uniformly vary between approximately 1 and 0.10.

4.7.1.4 Results and Discussions

Figures 4.3 and 4.4 plot the log likelihood values for Ns = 250 samples for model H_1 and H_5 before and after the implementation of nested sampling. In each model the lowest log-likelihood after nested sampling is much higher than before the algorithm was applied. Figures 4.3 and 4.4 show that, for both models, the nested sampling algorithm was able to sample approximately 70% percent more updating parameter values that minimize the difference between the measured data and the initial finite element model using the hard log-likelihood constraint as set out in the nested sampling algorithm point number 4.

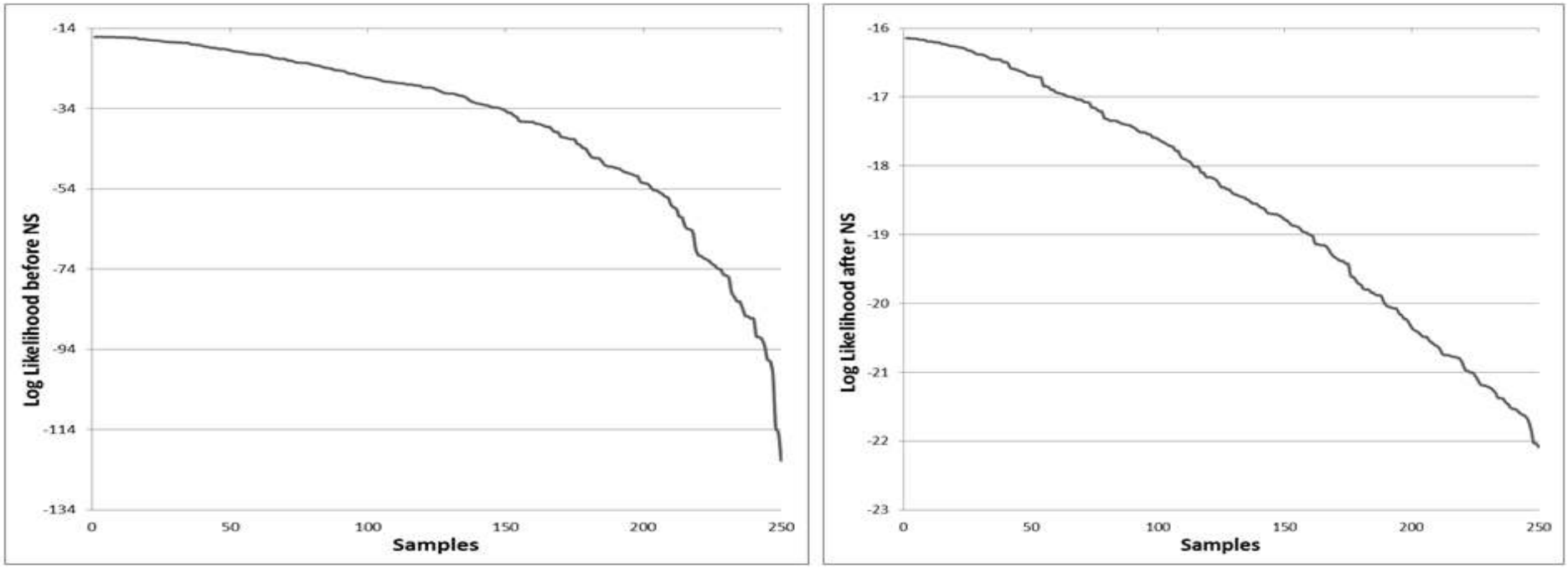


Log- Evidence before NS vs. Samples H_1



b) Log- Evidence after NS vs. Samples H_1

Figure 4.3 Log likelihood values for Ns = 250 from model H_1



c) Log- Evidence before NS vs. Samples H_5

d) Log- Evidence after NS vs. Samples H_5

Figure 4.4 Log likelihood values for Ns = 250 samples from model H_5

Tables 4.3 and 4.4 present the results of the stochastic simulations of the eight finite element beam models. These results are the posterior most probable parameter values based on the normal likelihood probability simulation from the three runs of the nested algorithm of each model in the set.

From the results in Table 4.3, model H_3 has, on average, the widest posterior distribution of all of the Young's modulus values and model H_1 has the smallest posterior distribution. Model H_5's Young's Modulus value of $7.16x10^{10}$ $N.m^{-2}$ is the closest to the standard material modulus while model H_1's modulus is the most varied at $6.82x10^{10}$ $N.m^{-2}$. In all of the models, the E_1 updating parameter has the lowest Young's modulus value and with quite small variances from the mean.

Not only is this the parameter that changes most, but also the parameter updating algorithm is confident in these E_1 posterior parameter updating values. The significance of this parameter's value is further supported by the relatively high averages for the E_2 updating parameters (which occupy the same location as E_1) in models H_6, H_7 and H_8. This suggests that in order for the finite element model to better approximate the real unsymmetrical H-beam the left side of the beam model should be less stiff than the standard aluminium beam. Table 4.5 shows the Bayesian log-evidence, data match and the information gain results of the nested sampling algorithm.

Table 4.3 Posterior mean and SD of updating parameter over three runs of NS

	MODEL H_1		**MODEL H_2**		**MODEL H_3**		**MODEL H_4**		**MODEL H_5**		**MODEL H_6**		**MODEL H_7**		**MODEL H_8**	
	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD	MEAN	STD
E1	6.8E+10	2.3E+09	6.8E+10	2.7E+09	6.6E+10	4.9E+09	6.8E+10	3.1E+09	6.7E+10	3.9E+09	6.8E+10	1.4E+09	6.8E+10	2.1E+09	6.8E+10	2.5E+09
E2	-	-	7.1E+10	6.0E+09	7.5E+10	6.9E+09	7.3E+10	6.8E+09	7.3E+10	5.9E+09	7.4E+10	7.1E+09	7.3E+10	5.9E+09	7.1E+10	7.5E+09
E3	-	-	-	-	7.0E+10	5.3E+09	7.1E+10	4.9E+09	7.4E+10	6.6E+09	-	-	-	-	7.2E+10	7.2E+09
E4	-	-	-	-	-	-	7.2E+10	7.0E+09	7.2E+10	5.9E+09	-	-	-	-	-	-
E5	-	-	-	-	-	-	-	-	7.3E+10	5.6E+09	-	-	-	-	-	-
$1/\sigma$ (rads)	0.13		0.11		0.18		0.14		0.14		0.13		0.15		0.14	

Table 4.4 Log Evidence, Data-Match and I-Gain for Unsymmetrical H-Beam Models

Model	Log Evidence	Data-Match	Information Gain
H_1	-25.43	-22.89	2.54
H_2	-23.39	-21.23	2.17
H_3	-31.22	-28.12	3.11
H_4	-25.94	-23.35	2.58
H_5	-25.38	-22.93	2.45
H_6	-25.22	-22.66	2.55
H_7	-27.36	-24.62	2.74
H_8	-25.95	-23.19	2.75

Table 4.4 shows that model H_2 has the highest log-evidence and model H_3 has the lowest log-evidence for the set of models. Model H_3 extracted the largest amount of information from the data to update its three parameters and was accordingly penalized for that. Surprisingly model H_5, which is the most complex model, extracted less information to update its five parameters and has a higher evidence measure than less complex models H_1, H_3, H_4, H_7 and H_8. Similarly models H_3, H_7 and H_8 extracted more information than the more complex model H_4.

This demonstrates that the Bayesian evidence measure not only penalizes the complexity of a model but also the amount of information the model extracts from the data to update its extra parameters. Models H_6 and H_7, which are similar, have a poorer data-match and they extract more information from the data than their complex equivalent model H_2, which has a higher evidence measure. In the similarly arranged models H_3, H_4 and H_5, the simplest model extracts the most information to update its parameters. Systematically splitting E_2 in model H_3 to become E_2 and E_3 in model H_4 and E_4 in model H_4 to become E_4 and E_5 in model H_5 has resulted in less information being extracted from the data.

The results demonstrate a common phenomenon that occurs when an extra variable is introduced whose effect on the model is similar to an existing parameter. This can often lead to over-fitting in classic regression but in Bayesian analysis this effect is counter-acted by a penalty that is imposed on any extra parameter through the information the model requires to update it. In order to evaluate/interpret the significance of the different Bayesian evidence measures we use Jeffrey's scale, with the results shown in Table 4.5.

Table 4.5 Jeffrey's scale and significance of Bayesian evidence

Model	Z_j / Z_i	$\log_e(Z_j / Z_i)$	Evidence against model H_i
H_3/H_2	1.33	0.28	Weak
H_3/H_8	1.24	0.22	Weak
H_3/H_6	1.23	0.21	Weak
H_6/H_2	1.07	0.06	Weak
H_7/H_2	1.16	0.16	Weak

It is clear from Table 4.5 that there is no real difference between all models. Jeffrey's scale highlights which models are more alike, for example model H_2 and model H_6 are more alike than H_2 is to H_7.

The Bayesian evidence statistic used in conjunction with Jeffrey's scale shows that an increased number of parameters will not necessarily result in a significantly better model. In the next section we present the second application example of Bayesian model updating using the nested sampling algorithm.

4.7.2 Example 2: GARTEUR SM-AG19 Testbed

This structure is described in detail in Appendix A2 as before. Three of the seven FE models, as presented in [54], are compared using our proposed Bayesian evidence statistic.

4.7.2.1 Candidate Mathematical Models

The three models used for comparison consist of one plate element model and two beam element models for the same GARTEUR SM-AG19 test structure. Table 4.6 details the location and type of uncertain parameter for each model.

The models are named according to their finite element types, e.g. model B_2 is an Euler-Bernoulli beam element model, and the model numbering is arbitrary. The lowest number of updating parameters in the three model set is from model B_2 with 6 parameters and the maximum is model P_2 and B_1 with 8 parameters each. As can be seen from Table 4.6 the updating parameters are quite varied across the three models. The updating parameter numbering is according to what property and which part of the test structure is updated e.g. E_1I_{1min}, means the Young's Modulus and minimum second moment of area of the fuselage are updated.

Table 4.6 FEMs and updating parameters for GARTEUR SM-AG19 structure

Model Name & Element Type	**Fuselage**	**Wing**	**R-Wing**	**L-Wing**	**Wing/Fuselage connection**	**V - Tail**	**Wing Thickness**	**R-Drum**	**L - Drum**	**Overall Density**	**Residual Type**
P_2 (Plate)	E_1 G_1	E_6 G_6			E_7 G_7 *(Steel)*	E_2 G_2	W_t	*Mass* (M_l)	*Mass* (M_r)	ρ	*Frequency*
B_1 (Beam)			I_{6MIN}, I_{6MAX},J_6	I_{5MIN}, I_{5MAX}, J_5		I_{2min}				ρ	*Frequency*
B_3 (Beam)	$E_1 I_{1min}$	E_4 I_{4MIN}, E_4 I_{4MAX}, ρ_4	G_6J_6	G_5J_5		E_2 I_{2min}					*Frequency*

$EI_{min,max}$: minimum and maximum bending stiffness. GJ : torsional rigidity. E: Young's modulus. G: shear modulus. ρ: mass density. M_R/M_L: right and left mass. W_T: wing thickness. Frequency: natural frequency.

4.7.2.2 Standard Properties

In order to quantify the model updating capabilities of the proposed nested sampling algorithm, we ran the three models with standard material property and measured geometric values for the uncertain parameters. Table 4.7 shows the FEM natural frequency, percentage frequency difference and average error results for each model compared to the two measured data sets.

Table 4.7 NF, PNFD and Avg. Errors for P_2, B_1 and B_2 using SMP and GM

P_2			**B_1**			**B_2**		
STD_{wn}	DLR	IC	**STD_{wn}**	DLR	IC	**STD_{wn}**	DLR	IC
	Δf (%)	Δf (%)		Δf (%)	Δf (%)		Δf (%)	Δf (%)
6.55	-2.61	-0.10	5.77	9.64	11.85	5.77	9.63	11.85
18.59	-15.48	-12.34	15.44	4.10	6.70	15.44	4.09	6.70
39.97	-20.65	-14.66	30.98	6.49	11.13	31.11	6.11	10.77
40.54	-20.91	-14.85	31.49	6.08	10.79	31.66	5.59	10.32
41.16	-15.45	-12.67	33.78	5.25	7.53	33.82	5.13	7.42
52.33	-8.16	-5.05	45.15	6.67	9.35	45.16	6.66	9.34
53.12	-7.47	-4.93	54.82	-10.90	-8.27	54.82	-10.90	-8.28
57.10	-3.66	-1.25	56.14	-1.93	0.44	56.14	-1.93	0.44
69.94	-10.95	-7.67	60.02	4.79	7.60	60.02	4.79	7.60
Avg Error	4.29	3.08		2.23	2.93		2.19	2.89

STD_{wn} is the natural frequency of each model using standard material properties and geometric measurements. Δf (%) is the percentage difference between the STD_{wn} and measured natural frequency data from each institute. **Note**: The Avg Error is the sum average of the absolute error between the above two natural frequencies.

In both sets of measured data and model comparisons, model B_2 has marginally lower average errors to B_1 and model P_2 has the highest average error. Clearly before updating, the plate model P_1 is a poor model of the given GARTEUR structure using this measured data. The main difference between B_1 and B_2 is that model B_1 has a split main-wing which uses separate updating parameters for the left (I_{5MIN} I_{5MAX} J_5) and right wing (I_{6MIN} I_{6MAX} J_6) while B_2 updates the whole wing under one parameter (I_{5MIN} I_{5MAX} J_5). Model B_1 only has one material

property (ρ - overall density) to update and the rest of the parameters are geometric in nature.

4.7.2.3 Prior Model and Nested Sampling Parameters

All model updating parameters shown in Table 4.8 except for the masses (M_R, M_L) and wing thickness (W_T) are sampled from a normal prior distribution. The masses and thickness are sampled from a uniform distribution between 0.15kg and 0.2kg and 9.5mm to 11mm respectively. Each model's nested sampling algorithm was run at the following settings: number of samples, N = 250; maximum number of iterations is 1500; and the number of MCMC steps is 15. The mean and variance values for all of the updating parameters are shown in Table 4.8. The inverse variance of the error probability function $p(D|\theta,M)$ is set to vary uniformly between 0.5 and 0.25. These values determine the assumed average variances of the difference between the measured and model natural frequencies for all models.

Table 4.8 Prior mean and variances for updating parameters

Parameter	Mean	Variance	Parameter	Mean	Variance
$E_{1\text{-}6}$	7.2e10 Pa	1e19	I_{2min}	8.33e-9 m^4	2e-20
E_7	21 e10 Pa	3e20	I_{5min}	8.33e-9 m^4	2e-10
$G_{1\text{-}6}$	2.8e10 Pa	1e19	I_{5max}	8.33e-7 m^4	2e-15
G_7	8.17E10 Pa	1e19	I_{6min}	8.33e-9 m^4	2e-20
ρ	2700 kg/m^3	2e4	I_{6max}	8.33e-7 m^4	2e-15
ρ_5	2700 kg/m^3	2e4	J_5	3.12e-8 m^4	2e-19
M_R & M_L	0.15-0.20 kg	Uniform distribution	J_6	3.12e-8 m^4	2e-19
I_{1min}	1.56e-6 m^4	2e-14	W_T	9.95-11 mm	Uniform distribution

In the next section we present the results of the application of the nested sampling algorithm to the GARTEUR SM-AG19 Testbed using the three models.

4.7.2.4 Results and Discussions

The Nested Sampling (NS) updating algorithm improved all of the models of the given structure. There is a maximum of a 70% and a minimum of a 11% improvement in the average natural frequency errors after NS. The maximum parameter value variations for all models are less than 8% of the standard values. The updated finite element model results are presented in Table 4.9. After nested sampling model P_1 achieves the lowest average error. This is not surprising as this model has the most updating parameters but practically this model will be computationally expensive to run. A better method to evaluate the updated models is to look at the model evidence statistic.

Table 4.10 shows the model evidences. According to these evidences and the Jeffrey's scale in Table 4.1, none of the models has a better than weak Bayesian evidence over the others for both sets of measured data. Model B_2 has the highest evidence and the highest information gain for the DLR data. This means the model used the prior parameter space efficiently to find the best posterior parameter values. This model's mean posterior values produced the highest model approximation to the measured DLR data; this is shown by the 0.10 data-match value. With regards to the IC data, model P_2 has the highest evidence but the lowest information gain. Again this model did not use the prior space efficiently; this is further confirmed by the data-match value being the worst in the model sets.

Table 4.9 NF, PNFD and Avg. Errors for P_2, B_1 and B_2 using NS outputs for updating

	Model Performance					
DLR Measured Data	**P_2**	**DLR**	**B_1**	**DLR**	**B_2**	**DLR**
ω_n	**ω_n**	**Δf (%)**	**ω_n**	**Δf (%)**	**ω_n**	**Δf (%)**
6.38	5.75	9.89	5.88	7.84	5.94	6.92
16.10	16.67	-3.56	15.73	2.29	15.84	1.60
33.13	34.50	-4.13	31.50	4.91	32.05	3.25
33.53	34.57	-3.11	32.03	4.48	32.71	2.46
35.65	39.00	-9.40	34.37	3.61	34.69	2.70
48.38	47.42	1.98	46.05	4.81	46.61	3.66
49.43	51.14	-3.46	55.03	-11.32	54.28	-9.81
55.08	54.60	0.88	56.15	-1.95	55.41	-0.60
63.04	64.54	-2.39	60.73	3.66	60.83	3.50
Avg Error		1.29		1.84		1.41

	Model Performance					
IC Measured Data	**P_2**	**IC**	**B_1**	**IC**	**B_2**	**IC**
ω_n	**ω_n**	**Δf (%)**	**ω_n**	**Δf (%)**	**ω_n**	**Δf (%)**
6.54	5.86	10.39	5.99	8.48	6.00	8.24
16.55	16.96	-2.47	16.03	3.14	15.96	3.57
34.86	35.23	-1.06	32.10	7.91	32.77	5.99
35.3	35.31	-0.03	32.67	7.45	33.69	4.55
36.53	39.47	-8.05	35.06	4.04	35.23	3.56
49.81	48.17	3.29	46.95	5.74	47.26	5.11
50.63	51.76	-2.23	55.98	-10.57	55.08	-8.78
56.39	55.35	1.85	57.17	-1.38	56.07	0.58
64.96	65.48	-0.81	61.77	4.92	60.98	6.13
Avg Error		0.972		2.24		1.94

Table 4.10 FEM Log Evidences, Data-Match and I-Gain for GARTEUR SM-AG19 models

Model	Log Evidence	Data -Match	Information Gain
P_2 (DLR)	-3.60	-3.07	0.53
B_1 (DLR)	-1.68	-0.92	0.76
B_2 (DLR)	-1.29	0.10	1.39
P_2 (IC)	-3.17	-2.90	0.27
B_1 (IC)	-3.92	-2.68	1.24
B_2 (IC)	-3.73	-0.83	1.90

As concluded in [39] it is difficult to assert the best model for the GARTEUR structure. In this experiment the proposed Bayesian evidence statistic reveals that no model that is significantly better than the others. Furthermore this Bayesian measure provides quantitative information for why this is so. Model B_2 is attractive in its use of parameter space but model B_1 is attractive in its use of fewer updating parameters for comparable performance. The plate model is not optimal for any of the measured data, especially with the 8 updating parameters, but there is no clear evidence against it. It should also be noted that the three models chosen, and the parameters selected, were the best models in the opinion of the seven groups involved in the original benchmark exercise [9,10,38,39,102].

4.8 Chapter Summary

In conclusion, we have argued that model selection should be integrated into the finite element model updating problem. Specifically, a Bayesian model updating perspective to the problem of finite element model updating is introduced. This paradigm is shown to be well established in other areas of science with little work available in the finite element updating literature.

Bayesian inference is able to simultaneously update the models whilst also incorporating a statistic to evaluate the evidence for each model. The theory of

Bayesian inference is presented together with a method to stochastically approximate the Bayesian evidence statistic. This method, nested sampling, was recently introduced as an efficient way of estimating integrals of the form needed to determine the model evidence.

We demonstrated the applicability of Bayesian model updating and the estimation of the evidence through two structural examples. In each example a number of competing finite element models are proposed to approximate the structure. Each model is unique and has a specific number of parameters and parameter identities. The method is initially demonstrated with a simple unsymmetrical structure, followed by a complex structure modelled by three completely different finite element type models. Bayesian inference is then applied to update each model. The evidence of each model is estimated from the updating parameter priors and the likelihood of the parameters after the data has been taken into consideration. The evidence calculation revealed that model complexity is not necessarily proportional to the model evidence. Two important factors affect the model evidence; these are how the model approximates the data and the amount of information each model extracts from the data to update its parameters. This is largely influenced by the choice of model parameters.

The Bayesian model evidence calculation allows the engineer to determine if there is any advantage in updating certain parameters in his/her model and what evidence one has that the chosen model is better than other models. To put the model evidences in context we further showed the significance of the model Bayesian evidence differences by using Jeffrey's scale. This statistic allows the engineer to determine if the model evidence differences are significant enough to warrant discarding some models.

Chapter 5

Summary and Future Work

5.1 Summary

Mathematical models form the basis of any scientific exercise. They are designed to describe, analyze and predict the behaviour of non-trivial complex systems. To narrow the field of models the thesis focuses on a specific type of mathematical models; finite element models (FEMs). The main limitation of any model lies is the definition of a model; it is an approximation of the real system. This means the analytical model results will not match the real system measurements. The question then becomes what can be done to the model to improve/update its approximation accuracy. This is the problem of finite element model updating.

With the definition limitation and the realization that any system can be modelled in a multitude of ways we described three fundamental challenges faced by any model. The first challenge is intrinsic to the model and asks what aspects of the model are uncertain for the model to result to a particular accuracy. The second challenge is how can such a model be efficiently updated and the last challenge is how do we know that any updated model is the best for a particular system. These challenges formed the basis of the approach followed in this thesis.

A graphical perspective to model updating highlighted the goal, challenges and limitations of the endeavor. Each challenge was noted to be general enough not to be specific to any one model. This allowed for a multi-model framework approach to FEM updating. Thus all the methods developed to address these challenges are also not specific to any one FEM but are applicable to all.

The thesis began by addressing the first challenge in Chapter 1. A method to search for an optimal updating parameter set for any FE model was developed. This method considers all possible uncertain parameters and collates them to form a vector. The algorithm, population based incremental learning (PBIL), searches this vector for the best parameter combination using an evolution based strategy. The uncertain parameter vector is probabilistically sampled for a good parameter combination and each combination is substituted to the model and evaluated on an objective function. Three, two beam and one plate element, models for one system are simulated on two objective functions and two measurement datasets. The Bayesian Information criterion (BIC) objective function arrives at the best updating parameter combination (BUPC) early on the algorithm iteration. Furthermore the most frequently selected parameter combination under the BIC is always the final BUPC. This is not the case for the sum of squared errors (SSE) objective function. The beam models were consistent on the best updating parameter across both datasets and functions. The plate model consistently benefited the most from the PBIL parameters selection and updating across all datasets and objective functions. The beam models slightly improved their average errors under this algorithm.

Having identified and updated the most uncertain parameters of a model in Chapter 2, we required a most efficient way of updating a FE model especially in a context of other models of the same system. To this end we developed an efficient method to simultaneously update a number of competing FE models. We argued that optimizing one model in seclusion is neither optimal nor informative with respect to other models. The proposed method, particle swarm optimization (PSO), is based on the behaviour of multiple biological entities foraging for food.

We implemented and analyzed the performance of this method on eight candidate FEMs of an unsymmetrical H-beam. The method was tested using two error functions; the Akaike Information criterion (AIC) and Sum of Squared errors (SSE). Using the AIC as the objective function always resulted in the simpler model having the best performance. This was not necessarily the case with the SSE function. The introduction of the adaptive inertia weight to the basic algorithm improved the convergence rate in both the AIC and SSE objective function experiments. This constant allowed the algorithm to initially explore wide updating parameter space and to later focus on the local parameter space. The model accuracy ordering under the AIC function was consistent before and after the introduction of the adaptive inertia weight. The SSE was not consistent in its choice of best model under such settings. Using PSO to update FE models proved to be efficient especially in the multi-model context.

In the last chapter we attempted to address one limitation of the PSO type algorithm when it is used to compare updated FE models. The proposed method simultaneously updates and quantifies the evidence for any one model. This allows for an objective measure of how good one updated model is over another. This is of fundamental importance in our multi-model setting. This method is called nested sampling. Nested sampling samples the FEM updating parameter space using a hard constraint on new samples. This allows for incremental improvements in model likelihood. This model updating is carried out within a Bayesian framework with the model parameter posteriors as a by-product. This nested sampling and Bayesian approach allows for the proposed model marginal likelihood or evidence to be evaluated. Jeffrey's scale then uses this Bayesian statistic to qualify model differences after the updating.

NS was implemented on two mechanical structures each with multiple models. Two different measurement dataset were used for the second structure. For one structure some models were clearly better than others and this could be explained by the proposed method. For the second structure it was clear from the method results that the models are similar.

5.2 Conclusions

Throughout all the thesis experiments one common conclusion is that the choices of the objective function is very important in any optimization scheme. Understanding the behaviour of a chosen function under different settings, e.g. different measurement datasets, is paramount. The FEM updating problem cannot be settled if the error function is not compared to other functions.

Fundamentally this thesis argues that any FEM updating problem is practically posed in a multi-model setting. It is concludes that updating one model in isolation is not informative in this setting. This thesis is therefore; finite element models of the same system should either be:

1. Updated simultaneously or
2. Be updated in the context of other models.

The first point does not favor any model over others and the second forces the need for a quantitative and qualitative measure of best model for the system.

5.3 Future Work

5.3.1 Multi Model Framework

Within the multi-model updating framework an objective evaluation or comparison of well-established updating procedures should be carried out. This work could be carried out on a few well know methods using one common structure, a priori chosen set of updating parameters and one objective function. This study would be useful in a number or ways, it would:

- Reveal which updating procedures are FE model specific and why.
- Which updating methods are universal and why.

5.3.2 Objective Function

A good choice of objective function was shown to be critical in obtaining a certain type of updating parameter set. Future work could study multiple objective functions and their influence on FEM updating especially the resultant updating parameters.

Limitation should not be placed on the number of functions perhaps addressing a multi-parameter problem like FEMU is better addressed in a multi-objective function setting. Each function could minimize/maximize a particular type of parameter e.g. material and the other function could target geometric type parameters. This multi-objective function optimization should act as a weighting for the different parameter types.

5.3.3 Sampling Updating Parameter Space

Achieving an optimal updating parameter set in the shorted time depends not only on the updating procedure but on how the parameter space is sampled. A common assumption is to sample from a Gaussian distribution. This is not always optimal and other distributions can be more representative for certain types of parameters.

Understanding a prior how these classes of (e.g. structural) parameters are distributed should minimize algorithm sampling time. This issue is most pertinent when the number of updating parameters (the search space) is large and efficient sampling techniques are required.

5.3.4 Validating FE models

The FEM updating problem is often limited to approximating the given measured data. Closely matching this data is often seen as having validated the updated model. Updated models are rarely tested on ‘unseen’ data. This is mainly because measured data is expensive and often limited to a dozen measured modes.

To overcome the data shortage more data from the same distribution as the measured data should be generated to form the ‘unseen’ set. With the

understanding of the properties (e.g. distribution, noise etc.) of this new data reasonable validation can be performed on the updated FEM. This would not only validate the updated model but the posterior distribution of the updated parameters could also be validated within the unseen data distribution.

References

[1] Ewins D.J, **Modal Testing: Theory and Practice**, Publisher Letchworth, Research Studies Press, 1984.

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

[3] Mottershead J.E, Friswell M.I, **Model updating in structural dynamics – A Survey**, Journal of Sound and Vibration, 167 (2) (1993) 347-375.

[4] Carvallo J, Datta B.N, Gupta A, Lagadapati M, **A direct method for model updating with incomplete measured data and without spurious modes**, Mechanical Systems and Signal Processing 21 (7) 2715-2731.

[5] Datta B.N, **Finite element model updating, eigenstructure assignment and eigenvalue embedding techniques for vibrating systems**, Mechanical Systems and Signal Processing 16 (2002), 83-96.

[6] Li W-M, Hong J-Z, **New iterative method for model updating based on model reduction**, Mechanical Systems and Signal Processing, 25(1) (2011), 180-192.

[7] Marwala T, **Finite element model updating using computational intelligence techniques: Applications to structural dynamics**, 2010 Heidelberg: Springer ISBN 978-1-84996-322-0.

[8] Adhikari S, Friswell M.I, **Distributed parameter model updating using the Karhunen-Loeve expansion**, Mechanical Systems and Signal Processing 24(2) (2010) 325-339.

[9] Bohle K, Fritzen C.-P, **Results obtained by minimizing natural frequency and mac-value errors of a plate model**, Mechanical Systems and Signal Processing 17 (1) (2003), 55-64.

[10] Goge D, Link M, **Results obtained by minimizing natural frequency and mode shape errors of a beam model**, Mechanical Systems and Signal Processing 17 (1), 2003, 21-27.

[11] Horton B, Gurgenci H, Veidt M, Friswell M.I, **Finite element model updating of the welded joints in a hollow section H-frame**, International Conference on Applications of Modal Analysis: Recent Advances in Modal Analysis Practice, 1999, 309-335.

[12] Jaishi B, Ren W-X, **Finite element model updating based on eigenvalue and strain energy residuals using multiobjective optimization technique**, Mechanical Systems and Signal Processing 21 (5) (2007) 2295-2317.

[13] Mottershead J.E, Mares C, Friswell M.I, James S, **Selection and updating of parameters for an aluminium space-frame model**, Mechanical Systems and Signal Processing, 14 (2000), 923-944.

[14] Mottershead J.E, Link M, Friswell M.I, **The sensitivity method in finite element model updating: A tutorial**, Mechanical Systems and Signal Processing, 25 (7) (2011) 2275-2296.

[15] Hua X.G, Ni Y.Q, Ko J.M, **Adaptive regularization parameter optimization in output-error based finite element model updating**, Mechanical Systems and Signal Processing 23 (3) (2009) 563-579.

[16] Marwala T, **Finite element model updating using response surface method**, Proceedings of the 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference, Palm Springs, California, USA, April 2004, AIAA Paper, 5165-5173, 2004-2005.

[17] Panigrahi S.K, Chakraverty S, Mishra B.K. **Vibration based damage detection in a uniform strength beam using genetic algorithm**, Meccanica, Springer, 44, (2009), 697 – 710.

[18] Titurus B, Friswell M.I, R**egularization in model updating**, International Journal of Numerical Methods in Engineering 75(4) (2008) 440-478.

[19] Fonseca J.R, Friswell M.I, Mottershead J.E, Lees A.W, **Uncertainty identification by the maximum likelihood method**, Journal of Sound and Vibration 288 (2005) 587-599.

[20] Horibe T, Takahashi K, **Crack Identification in beam using Genetic Algorithm and three dimensional P-FEM**, Journal of Solid Mechanics and Materials Engineering 1 (7) (2007) 886-894.

[21] Levin M.I, Lieven N.A.J, **Dynamic finite element model updating using simulated annealing and Genetic Algorithms**, Mechanical Systems and Signal Processing 12 (1) (1998) 91–120.

[22] Marwala T, **Finite element model updating using wavelet data and Genetic Algorithm**, Journal of Aircraft, 39 (4) (2002) 709-711.

[23] Passaro A, Júnior JR, Abe NM, Sasaki M, Santos J C, **Finite-element and Genetic Algorithm design of multi-segmented electro-optic**, Sensor for Pulsed High-Voltage Measurement, In 6th World Congresses of Structural and Multidisciplinary Optimization, 2005, Rio de Janeiro, Brazil.

[24] Perera R, Ruiz A, **A multistage FE updating procedure for damage identification in large-scale structures based on multiobjective evolutionary optimization**, Mechanical Systems and Signal Processing 22 (4) (2008) 970-991.

[25] Beck J.L, Yuen K-V, **Model selection using response measurements: Bayesian probabilistic approach**, Journal of Engineering Mechanics Vol. 130, pages 192-203, 2004.

[26] Beck J.L, Katafygiostis L.S, **Updating models and their uncertainties. Part I: Bayesian statistical framework**, Journal of Engineering Mechanics 124 (4) (1998) 455-461.

[27] Mares C, Dratz B, Mottershead J.E, Friswell M.I, **Model updating using Bayesian estimation**, ISAM 2006, Leuven, Belgium, pages 2607-2616, 18-20 September 2006.

[28] Marwala T, Sibisi S, **Finite Element model updating using Bayesian framework and modal properties**, Journal of Aircraft, 42 (1) January-February, 2005.

[29] Chen S, Billings S.A, Grant P.M, **Nonlinear system identification using neural networks**, International Journal of Control 51(6) (1990) 1191-1214.

[30] Ferariu L, Marcu T, **Evolutionary design of dynamic neural networks applied to system identification**, b'02 IFAC Congress, 2002, Barcelona, Spain

[31] Franklin G. F, Powell, J. D, Workman, M.L. **Digital control of dynamic systems**, 3rd. Addison-Wesley Longman Publishing Co., Inc. ISBN 0201331535, 1998.

[32] Hirotugu A, **A new look at the statistical model identification,** IEEE Transactions on Automatic Control 19 (6) (1974), 716–723.

[33] Hwang W.S, Lee D.H, **System identification of structural acoustics using the scale correction**, Mechanical Systems and Signal Processing 20 (2) (2006), 389-402.

[34] Ljung L, **System identification: Theory for the user**, Englewood Cliffs, NJ: Prentice hall, 1987 ISBN 0-13-881640.

[35] Jonsson M.T, Runarsson T.P, **Verifying finite element assumptions using structural system identification**, Proceedings of the Third IEEE Conference on Control Applications, (3) 1994 ,1739 – 1741.

[36] Leonard J.W, Khouri B.R, **System identification using a standard finite element program**, Engineering Structures 7 (3) (1985), 190-197.

[37] Ling X, Haldar A, **Element level system identification with unknown input with Rayleigh damping**, Journal of Engineering Mechanics, 130 (8) (2004), 877- 885.

[38] Balmes E, **Predicted variability and differences between tests of a single structure**, International Modal Analysis Conference (IMAC) 3242 (1) (1998), 558-564.

[39] Link M, Friswell MI, **Generation of validated structural dynamic models - Results of a benchmark study utilizing the GARTEUR SM-AG19 Testbed**, Mechanical Systems and Signal Processing, COST Action Special Issue, 17(1) January 2003, 9-20.

[40] Kim G-H, Park Y-S, **An Automated Parameter Selection Procedure for Finite Element Model Updating and its Applications**, Journal of Sound and Vibration 309 (3-5) (2008) 778-793.

[41] Mares C, Mottershead J.E, Friswell M.I, **Selection and updating of parameters for the GARTEUR SM-AG19 Testbed**, 3rd International Conference on Identification in Engineering Systems, Swansea, pages 130-141, April 2002.

[42] Baluja S, **Population-Based incremental learning: A method for integrating genetic search based function optimization and competitive learning**, Technical Report CMU-CS-94-163, Carnegie Mellon University,1994.

[43] Forrest S, Mitchell M, **What makes a problem hard for a Genetic Algorithm? Some anomalous results and their explanations**, Machine Learning 13 (1993) 285-319.

[44] Goldberg, D.E, **Genetic Algorithms in search, optimization and machine learning**, Reading, MA: Addison Wesley, 1989 ISBN: 0201157675.

[45] Holland J.H, **Adaptation in natural and artificial systems**, Ann Arbor, MI: University of Michigan Press, 1975 ISBN-13:978-0-262-08213-6.

[46] Mitchell M, **Introduction to Genetic Algorithms**, 1996, MIT Press ISBN 978-0-262-63185-3.

[47] Gosling T, Jin N, Tsang E, **Population based incremental learning versus Genetic Algorithms: Iterated prisoners dilemma**, Technical Report CSM – 401, University of Essex, England, (March 2004).

[48] Chaves-Gonzalez, J.M, Vega-Rodriguez, M.A, Dominguez-Gonzalez D, Gomez-Pulido J.A, Sanchez-Perez, J.M, **Population based incremental learning to solve the FAP problem**, 2nd International Conference on Advanced Engineering Computing and Applications in Science, Valencia 2008, 123-128.

[49] Rabil B.S, Sabourin R, Granger E, **Watermarking stack of grayscale face images as dynamic multi-objective optimization problem**, Proceedings of the 6th International Conference on the Advances in Mass Data Analysis of Images and Signals in Medicine, Biotechnology, Chemistry and Food Industry, 2011 New York, NY, USA, ISBN 978-3-942954-02-6.

[50] Yang S.Y, Ho S.L, Ni G.Z, Machado J.M, Wong, K.F, **Population-based incremental learning method for robust optimal solutions**, IEEE Transactions on Magnetics 46 (8) (2010) 3189 -3192.

[51] Bishop C, **Pattern recognition and machine learning**. Springer 2006.

[52] Liu J, Sun J, Wang S, **Pattern recognition: An overview**, International Journal of Computer Science and Network Security, 6 (6) (2006), 57-61.

[53] Ripley B, **Pattern recognition and neural networks**, Cambridge University Press, Cambridge, 1996.

[54] Guyon I, Elisseeff A, **An introduction to variable and feature selection**, Journal of Machine Learning Research 3 (2003), 1157-1182.

[55] Guyon I, Gunn S, Nikravesh M, Zadeh L (eds.), **Feature extraction, foundations and applications**, (Studies in Fuzziness and Soft Computing), Physica-Verlag, Springer, 2006

[56] Kira K, Rendell L.A, **A practical approach to feature selection**, Proceedings of the ninth international workshop on Machine learning, 249-256, 1992, Morgan Kaufmann Publishers Inc.

[57] Mitra P, Murthy C.A, Pal S.K, **Unsupervised feature selection using feature similarity**, IEEE Transactions on Pattern Analusis and Machine Intelligence, 24 (3) (2002), 301-312.

[58] Søndberg-Madsen N, Thomsen C, Peña J.M, **Unsupervised feature subset selection**, In Proceedings of the Workshop on Probabilistic Graphical Models for Classification, 2003, 71-82.

[59] Escalante H-J, Montes M, Sucar L-E, **Particle swarm model selection**, Journal of Machine Learning Research 10 (2009), 405-440.

[60] Acquah H de-Graft, **Comparison of Akaike information criterion (AIC) and Bayesian information criterion (BIC) in selection of an asymmetric price relationship**, Journal of Development and Agricultural Economics 2 (1) (20100, 1-6.

[61] Kuha, J, **AIC and BIC: comparisons of assumptions and performance,** Sociological Methods & Research, 33 (2) (2004), 188-229.

[62] Schwarz G, **Estimating the dimension of a model**, Annals of Statistics, 6 (2) (1978), 461-464.

[63] Weakliem DL, **A critique of the Bayesian information criterion for model selection**, Sociological Methods Research, February 27 (3) (1999) 3 359-397.

[64] Kennedy J, Eberhart R, **Particle swarm optimization**, In Proceedings of the International Conference on Neural Networks IV (1995) 1942-1948 IEEE.

[65] Dorigo M, Bhlum C, **Ant colony optimization theory: A survey**, Theoretical Computer Science, 344 (2005), 243-278.

[66] Kennedy J, Eberhart R, **Swarm intelligence**, Morgan Kaufmann Publishers, Inc., San Francisco, CA, 2001.

[67] Shi Y, Eberhart R.C, **Empirical study of particle swarm optimization**, In Proceedings of the Congress on Evolutionary Computation, pages 1945-1949, Piscataway, NJ, USA, 1999, IEEE.

[68] Voss M.S, Feng X, **ARMA model selection using particle swarm optimization and AIC criterion**, In Proceedings of the 15th IFAC Triennial World Congress on Automation Control, Barcelona, Spain, 2002.

[69] Marwala T, Finite **Element model updating Using Particle swarm optimization**, International Journal of Engineering Simulation 6 (2) (2005) 25-30.

[70] Hosseini, S.H.; Kashtiban, A.M.; Alizadeh, G, **Particle swarm optimization and finite-element based approach for induction heating cooker design**, International Joint Conference SICE-ICASE, 2006, 4624-4627.

[71] Yuan S-F, Chu F-L, **Fault diagnostics based on particle swarm optimization and support vector machines**, Mechanical Systems and Signal Processing, 21 (4) (2007) 1787-1798.

[72] Wang S, Meng B, **PSO algorithm for support vector machine**, 2010 Third International Symposium on Electronic Commerce and Security (ISECS), Guangzhou, 2010, 377-380.

[73] Tu C-J, Chuang L-Y, Chang J-Yang, Yang C-H, **Feature selection using PSO-SVM**, International Journal of Computer Science, 33 (1) (2007),111-116.

[74] Li S, Yang M, **Particle swarm optimization combined with finite element method for design of ultrasonic motors**, Sensors and Actuators A:Physical 148 (1) (2008), 285-289.

[75] Wang Z, Sun X, Zhang D, **A novel watermarking scheme based on PSO algorithm**, Bio-Inspired Computational Intelligence and Applications, Lecture Notes in Computer Science, 2007, Vol. 4688/2007, 307-314.

[76] You Z, Chen W, He G, Nan X**, Adaptive weight particle swarm optimization algorithm with constriction factor**, International Conference of Information Science and Management Engineering (ISME), 2010.

[77] MacKay D.J.C, **Information theory, inference and learning algorithms**, Cambridge University Press, 2003.

[78] Myung I.J, **The importance of complexity in model selection**, Journal of Mathematical Psychology, Vol. 44, pages 190-204, 2000.

[79] Muto M, Beck J.L, **Bayesian updating and model class selection for hysteretic structural models using stochastic simulation**, Journal of Vibration and Control, Vol. 14, No. 1-2, 2008, 7-34.

[80] Murray I, Ghahramani Z, **A note on the evidence and Bayesian Occam's razor**, Technical Report Gatsby Computational Neuroscience Unit GCNU-TR 2005-003. August 2005.

[81] Tipping M.E, **Bayesian inference: An introduction to principles and practice in machine learning**, In O. Bousquet, U. von Luxburg, and G. Rätsch (Eds.),Advanced Lectures on Machine Learning, 2004, pp. 41–62. Springer.

[82] Zellner A, Keuzenkamp H.A, McAleer M (eds), **Simplicity, inference and modelling (Keeping it sophisticatedly simple)**, Cambridge University Press, 2001.

[83] Saquib S.S, Bouman C.A, Sauer, K, **ML parameter estimation for Markov random fields with applications to Bayesian tomography**, IEEE Transactions on Image Processing, 7 (7) 1998, 1029-1044.

[84] Tong S, Koller D, **Active learning for parameter estimation in Bayesian networks**, Conference on Advances in Neural Information Processing Systems (NIPS 2000).

[85] Trotta R, Applications **of Bayesian model selection to cosmological parameters**, Monthly Notices of the Royal Astronomical Society 378 (2007), 72-82.

[86] Burnham, K. P, Anderson D.R, **Model selection and multimodel inference: A practical information-theoretic approach,** 2nd Edition, Springer, New York, NY, 2002.

[87] Mukherjee P, Parkinson D, Liddle A.R, **A nested sampling algorithm for cosmological model selection**, The Astrophysical Journal 638 (2006), L51-L54.

[88] Toni T, Stumpf M.P.H, **Simulations-based model selection for dynamic systems in systems and population biology**, Bioinformatics 26 (1) (2010), 104-110.

[89] Ching J.Y, Chen Y.C, **Transitional markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging**, Journal of Engineering Mechanics-ASCE 133 (2007), 816-832.

[90] Murray I, **Advances in Markov chain Monte Carlo methods**, PhD thesis, Gatsby Computational Neuroscience Unit (GCNU), University College London, 2007.

[91] Hensman J.J, Barthorpe R.J, **Feature extraction from spectral data using the Bayesian evidence framework**, Journal Key Engineering Materials, Damage Assessment of Structures VIII, 413- 414 (2009), 151-158.

[92] Shadbolt J, Taylor J.G (eds.), **Neural Networks and the financial markets: Predicting, combining and portfolio optimization – (Perspectives in Neural Computing)**, Springer, 2002.

[93] Skilling J, Nested sampling, In R. Fischer, R. Preuss, and U. von Toussaint, (eds.), **Bayesian inference and maximum entropy methods in science and engineering**, AIP Conference Proceedings 735 (2004), 395–405.

[94] Brewer B.J, Pártay L.B, Csányi G, **Diffusive nested sampling, Statistics and Computing**, 21 (4) (2011), 649-656.

[95] Skilling J, **Nested sampling for general Bayesian computation**, Bayesian Analysis 1 (2005), 833–860.

[96] Skilling J, **Nested sampling for Bayesian computations**, In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, Bayesian Statistics 8, Proceedings of the Valencia / ISBA 8th World Meeting on Bayesian Statistics, Oxford University Press, 2007.

[97] Feroz F, Hobson M.P, **Multimodal nested sampling: an efficient and robust alternative to MCMC methods for astronomical data analysis**, Monthly Notices of the Royal Astronomical Society 384 (2007), 449 – 463.

[98] Parkinson D, Mukherjee P, Liddle P, **A Bayesian model selection analysis of WMAP3**. Physical Review D 73 (2006), 1235231-123527.

[99] Chopin N, Robert C.P, **Properties of nested sampling**, Biometrika 97 (3) (2010), 741-755.

[100] Efron B, Gous A, **Scales of evidence for models selection: Fischer versus Jeffreys,** In Institute of Mathematical Statistics Lecture Notes- Monograph Series 38 (2001), 208 – 246.

[101] Jeffreys H, **An invariant form for the prior probability in estimation problems,** Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences **186** (1007) (1946), 453–461.

[102] Balmes E, Wright J, **GARTEUR group on ground vibration testing. Results from the test of a single structure by 12 laboratories in Europe**, International Modal Analysis Conference (IMAC) 3089 (2) (1997), 1246-1352.

Appendix

All the FE models are modelled using the Structural Dynamics Toolbox,

SDT® 6.2, for Matlab®.

Appendix A1

The Unsymmetrical H-Beam

The unsymmetrical H-shaped aluminium beam shown in figure A1.1 was used initially used by [37-39, 40, 41]. The H-beam structure had three thin cuts of 1mm that went half-way through the cross section. The cuts were introduced to elements 3, 4, 5. The structure with these cuts was used so that the initial FE model gives data that are far from the measured data, so as to test the proposed updating procedures on the FEMU problem. The structure was suspended on elastic rubber bands.

The beam structure was excited using an electromagnetic shaker and the response was measured using an accelerometer. The beam was divided into 12 elements. It was excited at point C and the acceleration was measured at 15 positions shown by open arrows. The structure was tested freely-suspended. The beam is free to move in all six degrees. A roving accelerometer was used for testing. The mass of the accelerometer was found to be negligible compared to the mass of the H-beam.

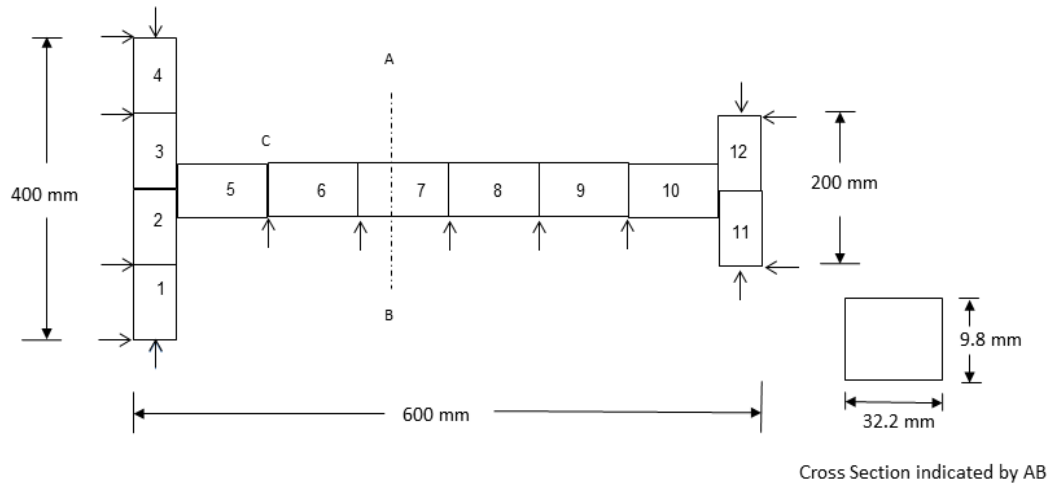


Figure A1.1 The 12 element Unsymmetrical H- Beam

The measured natural frequencies of interest of this structure occur at; 53.9Hz, 117.3Hz, 208.4Hz, 254Hz and 445Hz which correspond to modes 7, 8, 10, 11 and 13 respectively. This aluminium H-Beam had the properties shown in Table A1.1 below and these we called the standard material and geometric properties in the thesis.

Table A1.1 SMP and measured GP of Unsymmetrical H-beam

Material (Aluminium), Young's Modulus (E)	7.2×10^{10} N.m^{-2}
Length	600 mm
Width	32.2mm
Section Thickness	9.8 mm
Left Edge Length	400 mm
Right Edge length	200 mm
Density	2700 kg/m^3

All finite element model used standard isotropic material properties and Euler Bernoulli beam elements to approximate the beam sections of the structure.

Appendix A2

GARTEUR SM-AG19

In 1995-1996 this GARTEUR SM-AG19 structure was used as a benchmark study by 12 members of the GARTEUR Structures and Materials Action Group 19 [4,5,54]. One of the aims of the study was to compare the different computational model updating procedures on a single common test structure [3, 4, 8, 20, 32, 36]. The benchmark study also allowed participants to test a single representative structure using their own test equipment. Two experimental test results; data from DLR, Gottingen, Germany and data from the Imperial College of Science, Technology and Medicine (IC), U of K are considered in this thesis.

The aeroplane has a length of 1.5 m and a width of 3 m. The depth of the fuselage is 15cm with a thickness of 5cm. Figure A2.1 shows the two types of structural element choices; plate and beam, used to model the GARTEUR aeroplane in this thesis. In our models all element materials are standard isotropic. The beam model uses Euler–Bernoulli type elements and the plate model uses Kirchhoff shells. The natural frequencies measured by these two institutes are shown in Table A2.1. The difference between these is also shown in Table A2.1.

Table A2.1 Measured NF data from two institutes; DLR and IC

		Mode No.	1	2	3	4	5	6	7	8	9
Institute	**DLR**	**Frequency (Hz)**	6.38	16.10	33.13	33.53	35.65	48.38	49.43	55.08	63.04
	IC		6.54	16.55	34.86	35.30	36.53	49.81	50.63	56.39	64.96
Difference (%) IC/DLR			-2.50	-2.79	-5.22	-5.27	-2.46	-2.95	-2.42	-2.37	-3.04

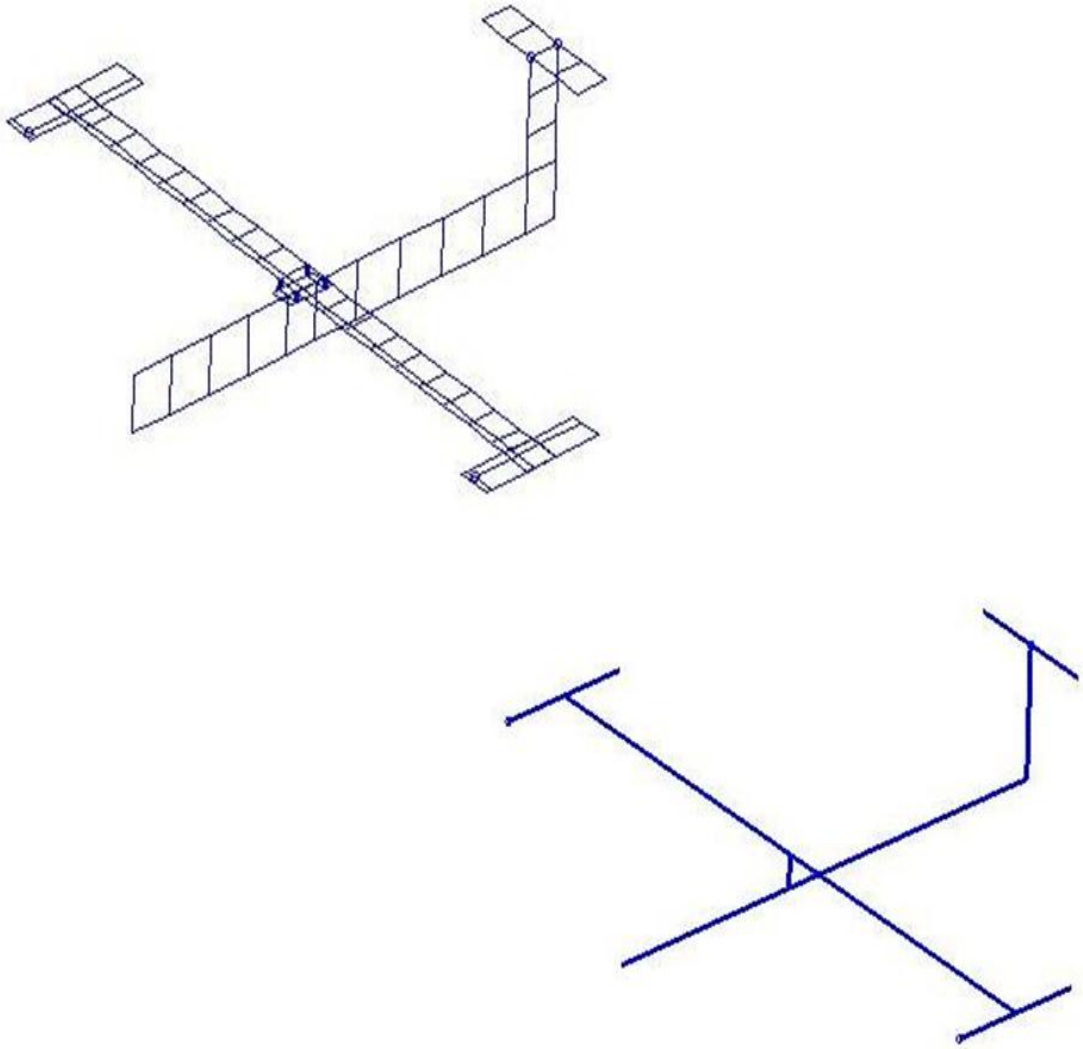


Figure A2.1 FEM of Garteur SM-AG19;Plate and Beam models

The following figures (A2.2 and A2.3) show the first nine modes of the PLATE model of the GARTEUR Structure using standard material and geometric data.

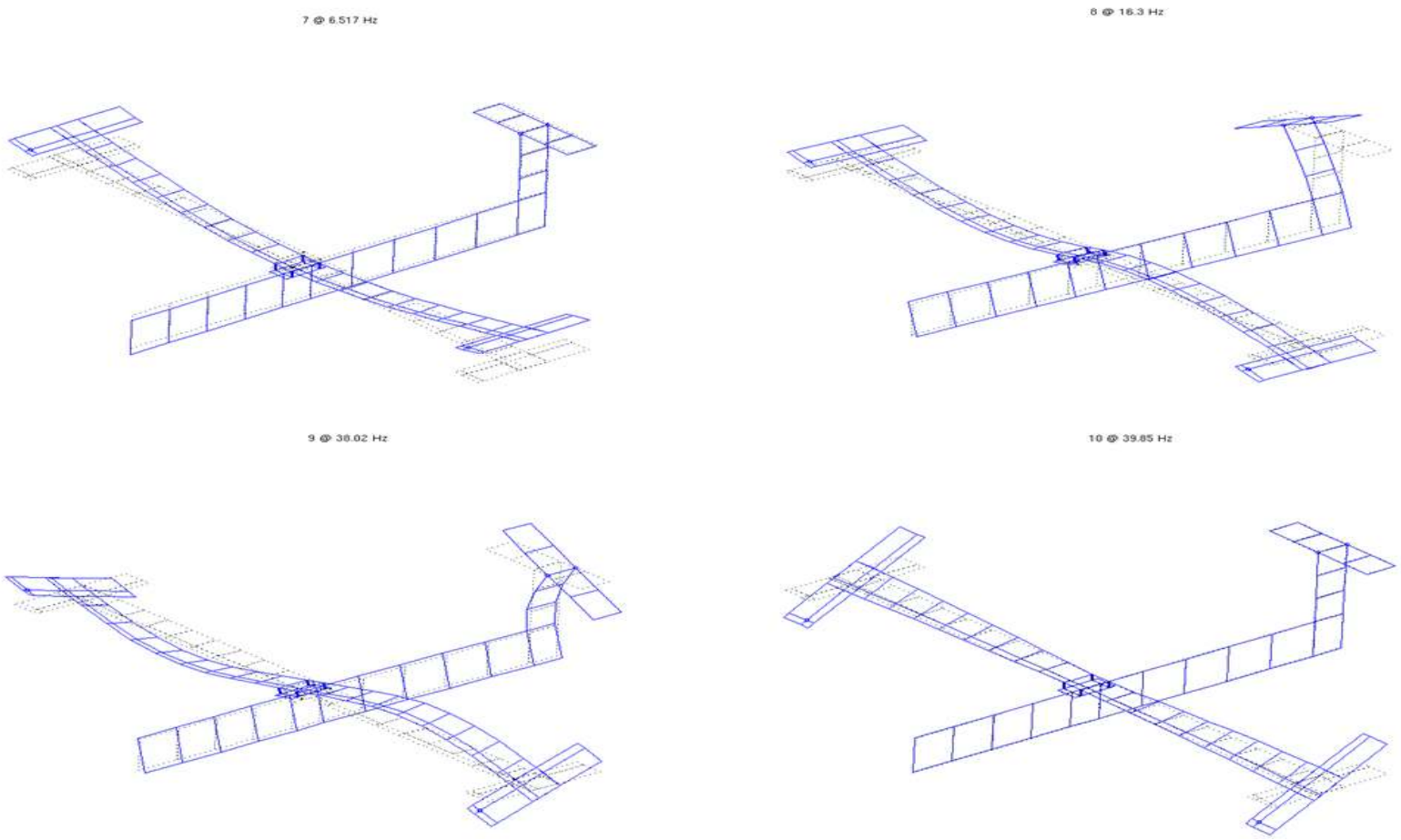


Figure A2.2 First four models of plate model

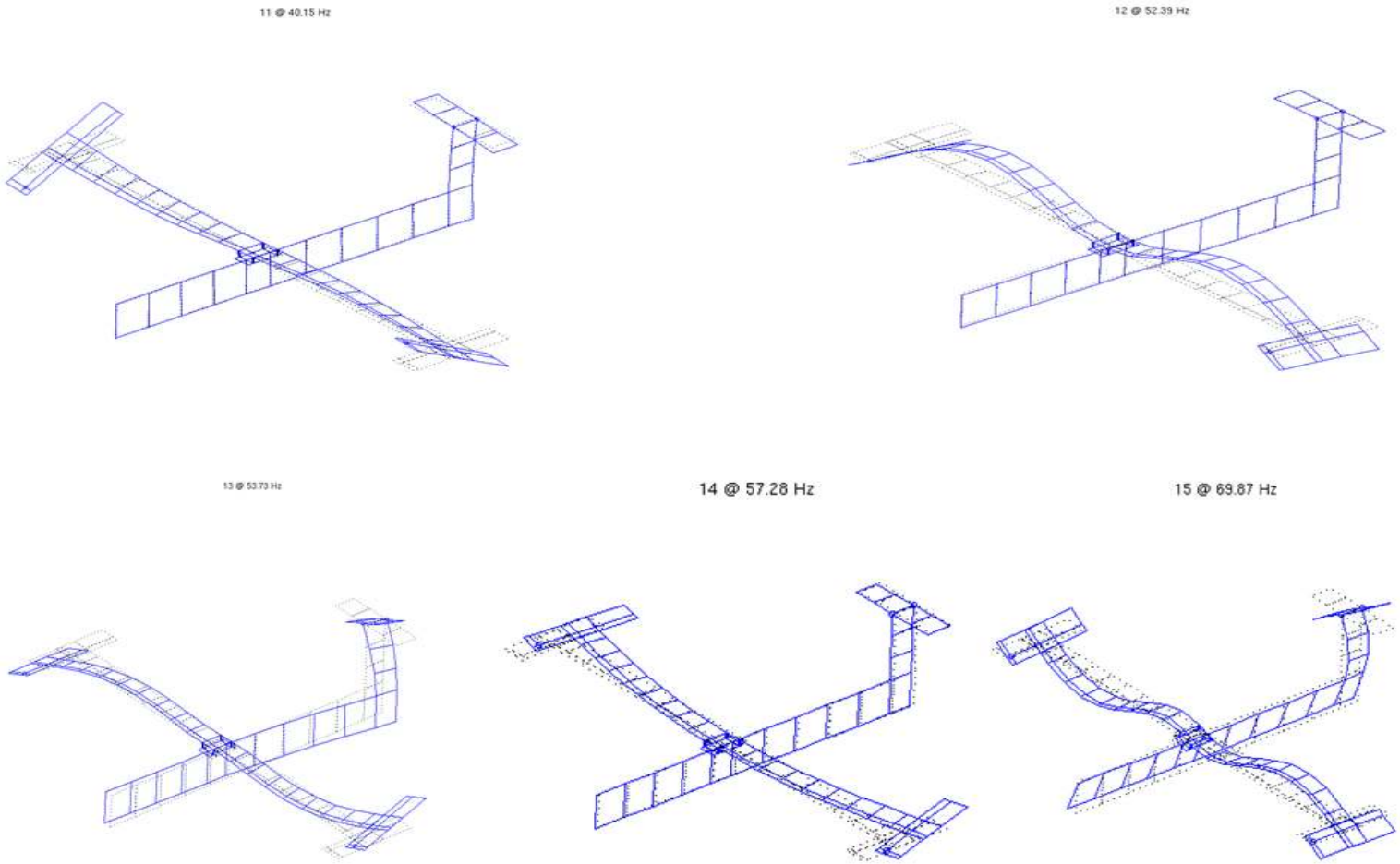


Figure A2.3 Fifth to ninth modes of plate model

The following two figures show the first nine modes of the BEAM model of the GARTEUR Structure using standard material and geometric data.

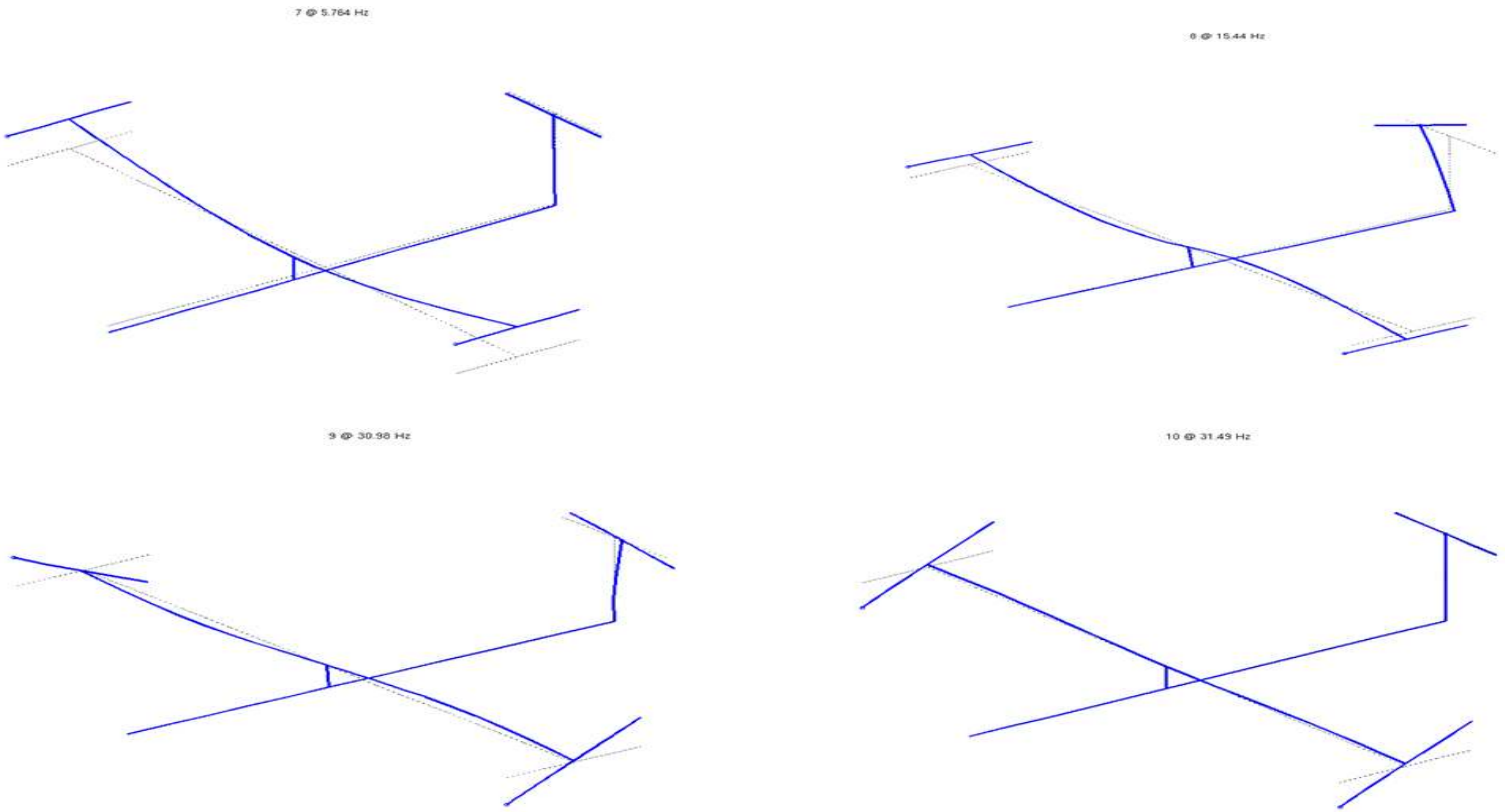


Figure A2.4 First four modes of beam model

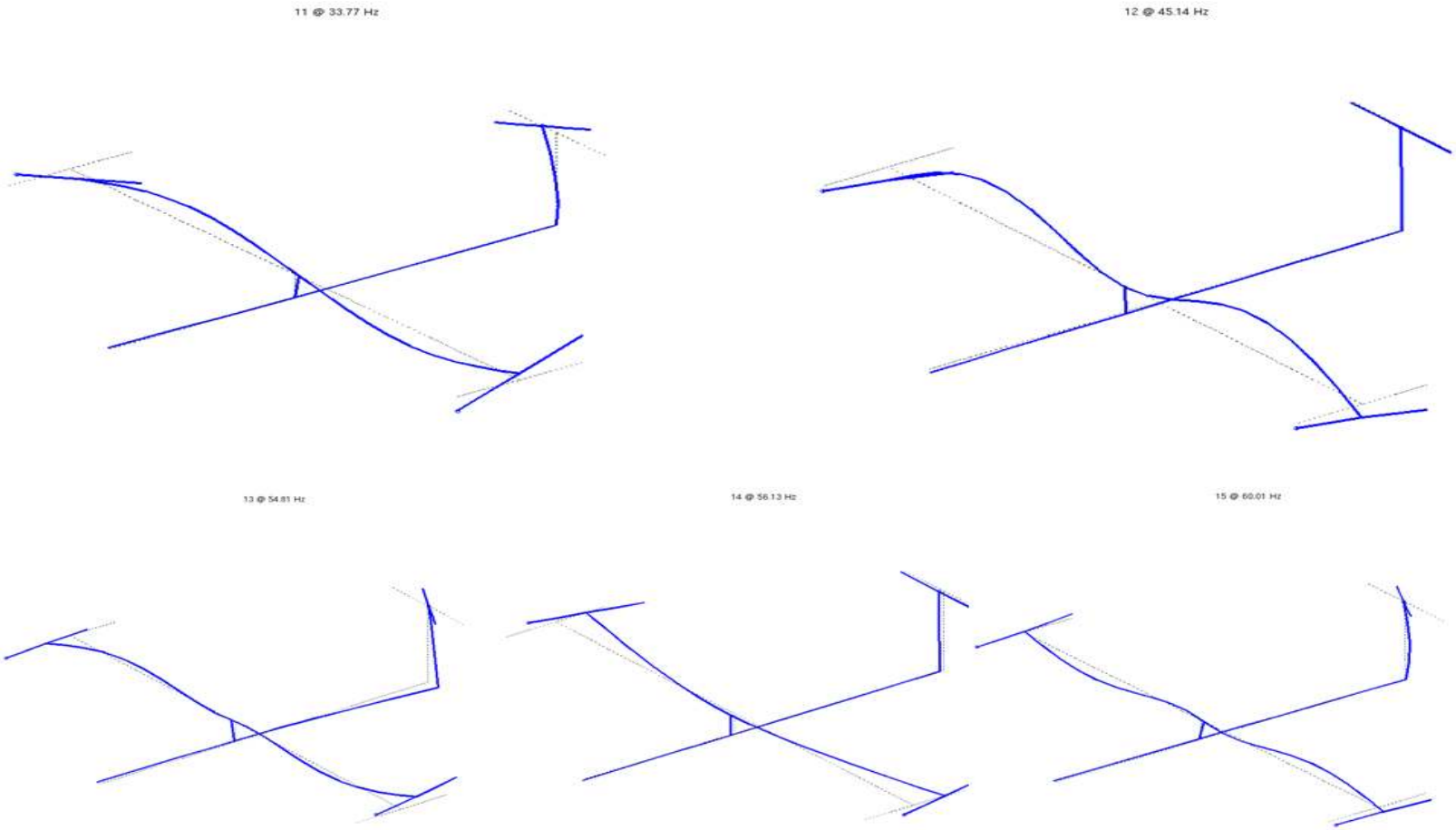


Figure A2.5 Fifth to ninth modes of beam model