Finite element model selection using Particle Swarm Optimization

Linda Mthembu¹, Tshilidzi Marwala², Michael I. Friswell³, Sondipon Adhikari⁴

¹Visiting Researcher, Department of Electronic and Computer Engineering, Faculty of Engineering and Built Environment, University of Johannesburg, PO Box 17011, Doornfontein 2028, South Africa.

²Executive Dean Faculty of Engineering and Built Environment, University of Johannesburg, PO Box 17011, Doornfontein, 2028, South Africa, South Africa.

³Professor of Aerospace Structures, School of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, United Kingdom.

⁴Chair of Aerospace Engineering, School of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, United Kingdom

Abstract

This paper proposes the application of particle swarm optimization (PSO) to the problem of finite element model (FEM) selection. This problem arises when a choice of the best model for a system has to be made from set of competing models, each developed a priori from engineering judgment. PSO is a population-based stochastic search algorithm inspired by the behaviour of biological entities in nature when they are foraging for resources. Each potentially correct model is represented as a particle that exhibits both individualistic and group behaviour. Each particle moves within the model search space looking for the best solution by updating the parameters values that define it. The most important step in the particle swarm algorithm is the method of representing models which should take into account the number, location and variables of parameters to be updated. One example structural system is used to show the applicability of PSO in finding an optimal FEM. An optimal model is defined as the model that has the least number of updated parameters and has the smallest parameter variable variation from the mean material properties. Two different objective functions are used to compare performance of the PSO algorithm.

Nomenclature

FEM                  Finite element model.
FEMU                Finite element model updating.
PSO                 Particle Swarm Optimization.
d               Model Dimension.
m_{id}                i-th Finite element model position at parameter d.
v_{id}                i-th Finite element model velocity at parameter d.
p_{id}                Best position for the i-th Finite element model at parameter d.
p_{gd}                Global best finite element model at parameter d.
w_{k}                The k-th dimension inertia weight.
M_{max}, M_{min}        Maximum and minimum model position respectively.
V_{max}, V_{min}        Maximum and minimum model velocity respectively.
E                  Young's Modulus.
AIC                Akaike Information Criterion.
SSE                Sum of Squared Errors.
µ        Mean.
msrd_{data}        Measured structural results.
fem_{results}        Finite element model results.
1. Introduction

The finite element model updating (FEMU) problem arises due to the mismatch between the initial finite element model results and the measure real system results [7, 9]. The modeller’s problem is then to determine which aspects or feature of the initial model are uncertain or incorrectly modelled. This is effectively a system-identification problem [15]. In classic system-identification the real measured system/data is approximated by a set of mathematical equations, usually polynomial equations whose parameters or order is unknown and is to be identified. In this paper the system is a structural system which is described by a finite element model and what needs to be identified are the uncertain parameters of the initial FE model [8, 9].

In the literature there are two main directions to finite element model updating (FEMU); direct and indirect (iterative) methods [9]. In the direct model updating paradigm [4, 9] the measured modal data are directly equated to the model modal values thus freeing up the system matrices (the mass, damping and or stiffness matrix elements) for updating. This approach often results in unrealistic system matrix element magnitudes, for example, physically unrealisable mass elements. In the indirect or iterative model updating approach the updating problem is formulated as an optimization problem, often approached by the use of least squares, maximum likelihood, eigenvalue sensitivity, genetic algorithm optimization and Bayesian approaches [1, 2, 8, 9, 12, 13, 16, 17, 18, 20, 21 and 26]. This paper proposes an iterative scheme to model updating.

This paper continues addressing the FEMU problem with the approach proposed in [20]. This approach is based on directly answering three standard questions in FEM updating; (a) which aspects of the models do we need to update? (b) how are we to update these models? (c) is the updated model the best one? In [20, 21] a particular Bayesian approach to model updating was presented. The methods considered updating FE models using Bayesian stochastic techniques from which one can determine the best model in a given group. The approaches presented in [20, 21] and in this paper effectively propose performing finite element model updating within the model selection framework. This whole approach is based on the premise that different analysts would differ (for example the updating parameters of the GARTEUR structure in [14] ) on which aspects of the initial finite element model are incorrectly modelled to answer question (a), and how to proceed with the model updating (when answering question (b)). This disagreement between researchers on the first two questions often makes it difficult to compare FEM updating results of the same structure let alone different updating techniques. To answer questions (b) and (c) we assume that a number of pre-existing potentially correct finite element models of a particular structure have been developed. These models could have been generated through engineering judgement or competing analysis by different people/techniques. Given the problem described in the previous paragraph, the problem is then to develop a method of both updating all the models and then selecting the best performing one. Ideally one would like an all-in-one procedure of both updating and selection. This would make it easier to compare the model updating results as they would have been updated ‘similarly’. The particle swarm optimization framework proposed in this article allows this simultaneous updating of all competing models and the identification of the best model in the given group.

2. Particle Swarm optimization (PSO)

Particle swarm optimization was first developed by [10]. 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 [5, 10, 11]. 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 tends to converge to optimal points in the search space.

The swarm behaviour metaphors has been adopted by the evolutionary computation community [5, 11, 13] 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 [13]), 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 these methods are well suited to problems where the solution search space is too large to search exhaustively [6, 11, 24, 26]. The practical interpretation of this analogy is that a