EngOpt 2008 - International Conference on Engineering Optimization
Rio de Janeiro, Brazil, 01 - 05 June 2008.

Design Optimization of Structures Including Repetitive Patterns

D. Akçay Perdahcıoğlu*, M.H.M. Ellenbroek*, P.J.M. van der Hoogt*, A. de Boer*

*Institute of Mechanics, Processes and Control - Chair of Structural Dynamics and Acoustics
University of Twente
P.O. Box 217, 7500AE Enschede, The Netherlands.
(d.akcay@utwente.nl)

Abstract

It is becoming a common practice to use surrogate models instead of finite element (FE) models in most of the structural optimization problems. The main advantage of these surrogate models is to reduce computation time as well as to make design optimization of complex structures possible. For surrogate modeling, firstly input-target pairs (training set) are required which are obtained by running the FE model for varying values of the design parameter set. Then the relationship between these pairs is defined via curve fitting where the created curve is named as a surrogate model. Once the surrogate model is found, it replaces the FE model in the optimization problem. Finally the optimization is performed using suitably chosen algorithm(s). Since solving an FE model may take very long time for certain applications, gathering the training set is usually the most time consuming part in the overall optimization process. Therefore, in this research the merits of the Component Mode Synthesis (CMS) method are utilized to gather this set for structures including repetitive patterns (e.g. fan inlet case). The reduced FE model of only one repeating pattern is created using CMS and the obtained information is shared with the rest of the repeating patterns. Therefore, the model of an entire structure is obtained without modeling all the repetitive patterns. In the developed design optimization strategy Backpropagation Neural Networks are used for surrogate modeling. The optimization is performed using two techniques. Genetic Algorithms (GAs) are utilized to increase the chance of finding the location of the global optimum. Since the optimum attained by GAs may not be exact, Sequential Quadratic Programming is employed afterwards to improve the solution. An academic test problem is used to demonstrate the strategy.

Keywords: Structural Optimization, Repetitive patterns, Component Mode Synthesis, Surrogate Modeling, Backpropagation Neural Networks, Genetic Algorithms, Sequential Quadratic Programming.

1. Introduction

Currently structural designs such as cars, aircrafts and aerospace appliances are analyzed extensively using FE method, possibly years before the first prototype is built. The benefits of the FE method may include; increased accuracy, faster and less expensive design cycle, better comprehension of the structure behavior. Therefore, it is an indispensable tool for the complicated engineering analyses. The correct static analysis of a complex structure highly depends on the size of the mesh. Thus, most of the structural models for industrial applications are composed of fine meshes which may involve several millions of degrees of freedom (d.o.f). On the other hand, investigating the dynamic properties of that structures require only few deformation modes which could be calculated with coarse meshed FE models. Consequently, reducing these models for structural dynamic analysis is essential in order to prevent time and computer memory consumption. The so-called *Component Mode Synthesis* (CMS) technique has been utilized since 1960s for the dynamic analysis of complex structures. The idea behind this technique is: dividing the structure into a number of substructures, calculating the corresponding reduced order FE models and then assemble them for obtaining a reduced order FE model of the complete structure. This technique is commonly preferred in the industry because it allows modeling of each substructure by different design groups and also any design changes in a single substructure affect only the system matrices of that substructure. Hence; if a modification is required in any specific substructure of a complex structure (e.g. solid rocket boosters of a space shuttle), only the system matrices of that particular substructure are changed and coupled with the rest of the already calculated substructure matrices which causes a significant save in computation time. Most of the structural designs involve repeating patterns in their geometry for instance; wings of a plane, one slice of a fan inlet case, etc. Generating the system matrices of one repeating pattern and utilizing the copies of it for the identical parts is another attribute coming with CMS.

One of the common problems identified by structural dynamics is the harmonic excitation of a structure at one of the resonance frequencies by an external force. This may cause large strains and large stresses in a structure which may lead to failure by fatigue. In most of the situations it is not possible to control the frequency content of the external load excitation. Therefore, resonance conditions can only be avoided by changing the design in order to keep the resonance frequency away from the excitation frequency. In reality there are always other factors that have to be considered besides shifting the natural frequencies. These might be; additional constraints coming from practical design and performance requirements for instance minimum total mass, effect of the modifications on the other dynamic properties, restrictions on the physical properties of the structure such as bounded lengths or widths. Under the concept of design optimization, all these criterion can be tackled at the same time.

In [1], the idea of integrating CMS into the design optimization scheme was in its infancy. The strategy and the employed techniques are improved a lot from then to now. The essence has been given to certain applications where the benefits of using CMS may cause a significant reduction in the computation time. In this research, attention is focused on the optimization of structures which have problems in their dynamic properties and repeating patterns in their geometries. The Craig-Bampton method is employed as an CMS method for the calculation of the corresponding reduced order FE models. Since it is still very time consuming to employ these models directly in the optimization problem, they are replaced by their surrogate models for the sake of computational efficiency. For surrogate modeling, firstly input-target pairs (training set) are required. Therefore after creating a sample set (inputs) based on the selected design parameters, the CMS reduced system matrices of the substructures involving the selected design parameters are computed for each element of the sample set. If there exists similar substructures, only one of them is taken into account and its system matrices are shared between the other similar ones. Next, all the reduced system matrices are assembled according to each element of the input set, solved and the targets are attained. Then the relationship between input-target pairs is defined via curve fitting where the created curve is named as a surrogate model. In our strategy Backpropagation Neural Networks (NNs) are used for surrogate modeling. Once the surrogate model is found, it replaces the reduced FE model in the optimization problem. Finally the optimization is performed using suitably chosen algorithm(s). In this research, optimization is performed using two techniques. Genetic Algorithms (GAs) are utilized to increase the chance of finding the location of the global optimum. Since the optimum attained by GAs may not be exact, Sequential Quadratic Programming (SQP) is employed afterwards to improve the solution. In other words, GAs are employed to provide an initial point for SQP which may lead to an exact global optimum.

This paper is built up as follows: In section (2), Component Mode Synthesis and the Craig-Bampton method are explained in details. In section (3) and (4); Neural Networks and the employed optimization strategies; Genetic Algorithms and Sequential Quadratic Programming; are pointed out. The suggested optimization strategy is introduced in section (5). Next, the strategy is demonstrated on an academic test problem and finally in section (7), conclusions are presented.

2. Component Mode Synthesis and Craig-Bampton Method

CMS has proven to be an efficient method for dynamic analysis of complex structures because of its economic and executive properties. It involves breaking up a large structure into several substructures (components), obtaining reduced order system matrices of each component and then assembling these matrices for attaining reduced order system matrices of the entire structure. All substructure calculations are independent from each other, therefore design changes in one component has no effect on the models of the other components.

In mathematical words, the technique can be explained as follows:

Let us assume that an FE model of a structure is constructed on a domain Ω and it is divided into N non-overlapping substructures where each component is defined on the sub-domain Ω^c. Thus, except the nodes on the interface boundaries, each node belongs to one and only one component. The linear dynamic behavior of an undamped component, labeled c, is governed by the local equilibrium equations,

$$\mathbf{M}^c\ddot{\mathbf{u}}^c + \mathbf{K}^c\mathbf{u}^c = \mathbf{f}^c + \mathbf{g}^c \quad c = 1, 2, \ldots, N \tag{1}$$

where $\mathbf{M}^c$, $\mathbf{K}^c$ and $\mathbf{u}^c$ are the mass matrix, stiffness matrix and vector of local d.o.f of the component, respectively. The vector $\mathbf{f^c}$ represents the external loads and the vector $\mathbf{g^c}$ stands for the interface forces between the component c and the neighboring components that assures dynamic equilibrium at the interfaces. The partitioned form of Eq.(1) can be written as follows:

$$\begin{bmatrix} \mathbf{M}^c_{ii} & \mathbf{M}^c_{ib} \\ \mathbf{M}^c_{bi} & \mathbf{M}^c_{bb} \end{bmatrix} \left\{ \begin{array}{c} \ddot{\mathbf{u}}^c_i \\ \ddot{\mathbf{u}}^c_b \end{array} \right\} + \begin{bmatrix} \mathbf{K}^c_{ii} & \mathbf{K}^c_{ib} \\ \mathbf{K}^c_{bi} & \mathbf{K}^c_{bb} \end{bmatrix} \left\{ \begin{array}{c} \mathbf{u}^c_i \\ \mathbf{u}^c_b \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{f}^c_i \\ \mathbf{f}^c_b \end{array} \right\} + \left\{ \begin{array}{c} \mathbf{g}^c_i \\ \mathbf{g}^c_b \end{array} \right\} \tag{2}$$

where i and b refer to *interior* and *boundary*, respectively.

It has been already discussed that in dynamic analyses using the information of all d.o.f is not necessary. Thus; in CMS for reducing the structure model, internal node displacements $\mathbf{u}^c_i$ of each substructure are replaced with their approximation. This is done by employing a *transformation matrix* $\mathbf{T}^c$ and a vector of internal nodes η^c such as

$$\left\{ \begin{array}{c} \mathbf{u}^c_i \\ \mathbf{u}^c_b \end{array} \right\} \approx \mathbf{T}^c \left\{ \begin{array}{c} \eta^c \\ \mathbf{u}^c_b \end{array} \right\} \tag{3}$$

and $dim(\eta^c) \ll dim(\mathbf{u}^c_i)$. $\mathbf{T}^c$ is built up using *reduction bases.*

In Craig-Bampton method [2], the reduction basis are obtained utilizing the *fixed interface normal modes* and the *constraint modes* of each component.

The fixed interface normal modes are calculated by restraining all d.o.f. at the interface and solving the usual eigenvalue problem

$$(\mathbf{K}^c_{ii} - \omega^2_j \mathbf{M}^c_{ii})\{\phi^c_i\}_j = 0 \qquad j = 1, 2, \ldots, F \tag{4}$$

where ω_j, $\{\phi^c_i\}_j$ stand for the eigenvalue and the corresponding eigenvector of the j^{th} normal mode, F is the number of truncated normal modes. The fixed interface normal modes of a component c are

$$\phi^c = \begin{bmatrix} \{\phi^c_i\}_1 & \{\phi^c_i\}_2 & \ldots & \{\phi^c_i\}_F \\ \mathbf{0}_b & \mathbf{0}_b & \ldots & \mathbf{0}_b \end{bmatrix} = \begin{bmatrix} \phi_j \\ \mathbf{0}_b \end{bmatrix}^c \qquad j = 1, 2, \ldots, F. \tag{5}$$

The constraint modes are calculated by statically imposing a unit displacement to the interface d.o.f. one by one while keeping the displacement of other interface d.o.f. zero and the interior d.o.f. of the substructure force free such that,

$$\begin{bmatrix} \mathbf{K}^c_{ii} & \mathbf{K}^c_{ib} \\ \mathbf{K}^c_{bi} & \mathbf{K}^c_{bb} \end{bmatrix} \begin{bmatrix} \psi^c_{ib} \\ \mathbf{I}^c_{bb} \end{bmatrix} = \left\{ \begin{array}{c} \mathbf{0}^c_{ib} \\ \mathbf{R}^c_{bb} \end{array} \right\} \tag{6}$$

where $\mathbf{R}^c_{bb}$ stands for the unknown reaction forces. The constraint mode matrix ψ of component c is defined as

$$\psi^c = \begin{bmatrix} \psi^c_{ib} \\ \mathbf{I}_{bb} \end{bmatrix} = \begin{bmatrix} -{\mathbf{K}^c_{ii}}^{-1}\mathbf{K}^c_{ib} \\ \mathbf{I}_{bb} \end{bmatrix}. \tag{7}$$

Therefore, the *Craig-Bampton transformation matrix* $\mathbf{T}^c_{CB}$ for component c is

$$\mathbf{T}^c_{CB} = \begin{bmatrix} \phi_j & \psi_{ib} \\ \mathbf{0}_b & \mathbf{I}_{bb} \end{bmatrix}^c \tag{8}$$

and the *Craig-Bampton reduced stiffness and mass matrices* are; $\mathbf{K}^c_{CB} = {\mathbf{T}^c_{CB}}^T \mathbf{K}^c \mathbf{T}^c_{CB}$, $\mathbf{M}^c_{CB} = {\mathbf{T}^c_{CB}}^T \mathbf{M}^c \mathbf{T}^c_{CB}$, respectively. The external loads and the internal forces are $\mathbf{f}^c_{CB} = {\mathbf{T}^c_{CB}}^T \mathbf{f}^c$, $\mathbf{g}^c_{CB} = {\mathbf{T}^c_{CB}}^T \mathbf{g}^c$, respectively.

After reducing the system matrices of each substructure, the next step is the assembly of all these matrices. The substructures can be interpreted as macro elements for the assembly. The local reduced d.o.f of a component c is related to the reduced d.o.f $\mathbf{u}_s$ of the entire structure by

$$\left\{ \begin{array}{c} \eta^c \\ \mathbf{u}^c_b \end{array} \right\} = \mathbf{B}^c \mathbf{u}_s. \tag{9}$$

The matrix $\mathbf{B^c}$ is a boolean matrix which relates the boundary d.o.f $\mathbf{u}^c_b$ and the interior d.o.f η^c of component c to the corresponding d.o.f of the entire structure. Hence, Eq.(9) is a *compatibility condition* between each substructure c, $c = 1, 2, \ldots, N$ and the structure. Using this condition, the local equilibrium equations, Eq.(1), including the Craig-Bampton reduced system matrices can be assembled as:

$$\mathbf{M}_s\ddot{\mathbf{u}}_s + \mathbf{K}_s\mathbf{u}_s = \mathbf{f}_s \tag{10}$$

where

$$\mathbf{M}_s = \sum_{c=1}^{N} \mathbf{B}^{c^T}\mathbf{M}^c_{CB}\mathbf{B}^c \quad , \quad \mathbf{K}_s = \sum_{c=1}^{N} \mathbf{B}^{c^T}\mathbf{K}^c_{CB}\mathbf{B}^c \quad , \quad \mathbf{f}_s = \sum_{c=1}^{N} \mathbf{B}^{c^T}\mathbf{f}^c_{CB}$$

are the reduced mass and stiffness matrices and the external load vector of the entire structure. It is important to point out that the interface forces $\mathbf{g}^c_{CB}$ are all cancelled out after assembly. This assembly is called *primal assembly* where the substructures are assembled using the compatibility of the interface nodes.

3. Neural Network Surrogate Models

The Artificial Neural Network (ANN) structure is inspired by the working principle of the brain. The neurons considered in ANN are simple abstractions of biological neurons and they are used to predict the relations between a particular input-target data set. As it is deduced from [3], a two layer NN having a nonlinear transfer function with sufficient number of neurons in the hidden layer and a linear transfer function in the output layer can be trained to approximate any function. This ability to approximate functions to any desired degree of accuracy makes NNs attractive tools for surrogate modeling. A two layer NN structure is illustrated in Figure 1.

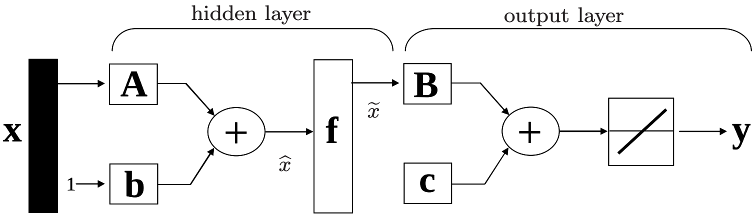


Figure 1: A two layer NN structure.

A mathematical description of a two layer NN can be given as

$$\begin{aligned}\widehat{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}\\ \widetilde{\mathbf{x}} &= f(\widehat{\mathbf{x}})\\ \mathbf{y} &= \mathbf{B}\widetilde{\mathbf{x}} + \mathbf{c},\end{aligned} \tag{11}$$

where $\mathbf{x} \in \mathbb{R}^{N_i\times 1}$, $\mathbf{y} \in \mathbb{R}^{N_h^2\times 1}$ represent the input-target vectors (training set), $\widetilde{\mathbf{x}} \in \mathbb{R}^{N_h^1\times 1}$ stands for the hidden layer outputs (at the same time an input for the output layer) and N_i, N_h^1, N_h^2 denote the number of input vector elements, hidden layer neurons and output vector elements, respectively. The number of neurons utilized in the hidden layer have an effect on the complexity of the network. The vector function $\mathbf{f} : \mathbb{R}^{N_i\times 1} \rightarrow \mathbb{R}^{N_h^1\times 1}$ used in the hidden layer stands for a set of nonlinear (sigmoid) transfer functions and allows the network to learn nonlinear and linear relationships between input-target pairs. The linear transfer functions employed in the output layer enables the network to produce values outside the range of sigmoid functions. The abbreviations $\mathbf{A} \in \mathbb{R}^{N_h^1\times N_i}$, $\mathbf{B} \in \mathbb{R}^{N_h^2\times N_h^1}$, $\mathbf{b} \in \mathbb{R}^{N_h^1\times 1}$ and $\mathbf{c} \in \mathbb{R}^{N_h^2\times 1}$ stand for the network parameters. The weights $\mathbf{A}$, $\mathbf{B}$ have an effect on the slope of the network output and the bias terms $\mathbf{b}$, $\mathbf{c}$ shift the entire network output on the coordinate axis [4].

Working principle of NNs are the same as the Least Squares Method (LSM). NNs are provided with a set of input-target pairs $\{\mathbf{p}_1, \mathbf{t}_1\}, \{\mathbf{p}_2, \mathbf{t}_2\}, ..., \{\mathbf{p}_Q, \mathbf{t}_Q\}$ where $\mathbf{p}_q$ is an input to the network and $\mathbf{t}_q$ is the corresponding target (input might be thickness and width, target might be one of the natural frequencies). First, input pairs are applied to the network and the corresponding network outputs are obtained. Then, these outputs are compared to the target values and the network parameters (weight and bias terms) are adjusted in order to minimize the mean square error between the network output and the target

$$\min_{\mathbf{A},\mathbf{B},\mathbf{b},\mathbf{c}} F_m = \sum_{q=1}^{Q} (\mathbf{t}_q - \mathbf{y}(p_q))^T (\mathbf{t}_q - \mathbf{y}(p_q)) \tag{12}$$

where Q is the total number of input-target pairs and y is a function of network parameters. Eq.(12) defines an unconstrained optimization problem and can be solved using any appropriate iterative algorithm. Most of the traditional numerical algorithms need the knowledge of the gradient. Thus, for the solution of Eq.(12), the partial derivatives of F_m with respect to the network parameters are required. Since F_m is an implicit function of the hidden layer parameters, the chain rule of calculus is used to calculate the derivatives which proceeds from the output layer through the hidden layer. The Backpropagation NNs take their name from this property.

As it is mentioned before, NN complexity is determined by the number of neurons utilized in the hidden layer. The increasing number of neurons leads to highly nonlinear NN structures which may cause *over-fitting*. Over-fitting occurs when the error on the training set is driven to a very small value but in the case of a new input-target pair involvement, the network becomes too poor to predict the new situation. Thus, the number of hidden layer neurons play a crucial role in learning process. When there is no information about the complexity of the underlying behavior, this number can not be estimated beforehand. In order to prevent finding it by trial and error, there are several developed techniques. In this study *regularization* is utilized which ensures that the surrogate model computed by the network is no more curved than necessary. This is achieved by modifying Eq.(12) with a penalty term F_p

$$\min \;\; F = \alpha F_m + \beta F_p. \tag{13}$$

where α and β stand for objective function parameters. One possible choice for the penalty term comes from the observation that an over-fitted function with regions of large curvature have large network parameters at these locations. If these parameters are penalized then it is possible to attain a smooth network response. In this study, *the sum of squares of the network parameters* is employed as a penalty term. Another challenge in Eq.(13) is the decision of the objective function parameters α and β. Their relative size determines the training process. If $\alpha \gg \beta$, the training algorithm minimizes the model error. If $\alpha \ll \beta$, the training algorithm smoothes the network response. The *Bayesian regularization* of Mackay [5] is used for the calculation of these parameters and Eq.(13) is solved by Levenberg-Marquardt method. The algorithm defined in [6] is utilized for this purpose in our research.

There are few points that are useful to take into account about NNs. Before training NNs, mapping the training data into the range $[-1, 1]$ enables to obtain better results. Additionally, in some situations the Backpropagation algorithm does not present the correct weights and biases for the optimum solution. That is because the nonlinear transfer functions in the hidden layer introduce many local minima into Eq.(13). The numerical techniques used to minimize this function are gradient based methods. Therefore, depending on the initial point, network solution can be trapped in one of the local minima. This can be prevented by reinitializing the network and retrain it several times until satisfactory convergence is obtained.

As a conclusion, unlike Response Surface Methodology and Kriging, NNs do not require any preliminary assumptions on the shape of the surrogate model. This is automatically done by utilized transfer functions and the hidden layer neurons in the network structure. Probable over-fitting caused by the improper hidden layer neuron number selection is prevented by regularization. In that sense, NNs are very flexible and effective tools for surrogate modeling if they are used in a proper way.

4. Optimization

Many structural optimization problems require the solution of non-convex nonlinear optimization problems where non-convexity may introduce multiple local optima. Pursue of global optimum is one of the main concerns of many researchers. Classical Nonlinear Programming (NLP) techniques may have the risk of being trapped in one of the local optima based on the selected initial point. Therefore in our strategy, Sequential Quadratic Programming (SQP); a widely used classical NLP technique; is utilized in combination with GAs. GA is employed to provide an initial point for SQP which may lead to a global optimum. Then SQP is called with that point to find an exact optimum solution.

4.1. Genetic Algorithms

Genetic Algorithm (GA) is a method for solving parameter optimization problems in the global sense by imitating the principles of natural evolution. The working principle of the method can be summarized as follows: First, GA is initialized with a random set of points (population). Next, the value of the objective function is calculated for each element of this set. Then GA selects some of these points based on their objective function values and creates a new set of points using them with some rules (mutation, crossover, etc.). Afterwards it replaces the previous population with the new one and follows the same procedure until there is no improvement in the population. The best point (one with minimum objective function value for a minimization problem) of the last population is the optimum solution.During its process, GA does not require any derivative information of an objective function.

The algorithm utilized in this paper solves problems including bound and linear constraints and unconstrained optimization problems by generating feasible points. The feasible points are computed either by making random changes to a single point (mutation) or by combining the vector entries of a pair of points (crossover).

Since a region restricted by bound and linear constraints define a convex set, the feasible crossover-point can be generated using the convex set definition such as:

$$\text{crossover-point} = \alpha x_m + (1-\alpha)x_n,\ \alpha \in [0,1]$$

where x_m, x_n are the selected points for crossover.

The mutation operator creates mutation-point by selecting a feasible direction in the design space and modifying the selected point on that direction with a sufficiently small step size.

When nonlinear constraints are involved into the optimization problem, they are introduced to the objective function with some parameters and a subproblem is created. Then GA solves this subproblem, modifies the parameters according to some rules and creates a new subproblem. This results in a new optimization problem. Until the stopping criteria is met this procedure is followed. For problems having nonlinear constraints, the *Composite Lagrangian Barrier-Augmented Lagrangian* (CLB-AL) algorithm of Conn et al. [7,8] provides a framework for the employed GA algorithm.

Unfortunately there is no convergence theory for GAs. Their solutions are based on estimations and might not be exact. On the other hand, a solution provided by GAs is likely to be close to a global optimum. It is also important to mention that, compared to the classical NLP techniques they are slow.

4.2. Sequential Quadratic Programming

In SQP, an NLP problem is attempted to solve using a sequence of Quadratic Programming (QP) subproblems. At each major iteration of SQP, an approximation is made for the QP problem parameters for generating a subproblem. Then the subproblem is solved and its solution is used to define a search direction for the next iteration point. The QP parameters are updated utilizing the new iteration point which generates a new subproblem. This procedure continues until a convergence to an optimum is obtained.

The construction of QP subproblems are the same for all SQP strategies. Available strategies only differ by selection of an QP solver and a merit function which promotes convergence from arbitrary starting points. In this study, null space active set method of Gill et. al. [9] is used for solving QP subproblems. The merit function is selected as in [10].

SQP is based on a strong convergence theory and its solutions are exact. Its disadvantage is that depending on the selected initial point it might be trapped in one of the local optima.

5. The Design Optimization Strategy

The design optimization strategy is illustrated in Figure 2. It starts with the *problem analysis* which firstly involves understanding the problem under consideration. Then selection of the design parameters and parameterization of the FE model for surrogate modeling, based on the obtained observations. Finally, decision of the objective and constraint functions of the optimization problem is made.

The second step in the strategy is the design of experiments. Here, a set of sample points are selected from the design space for surrogate modeling. In most of the situations there is no flexibility

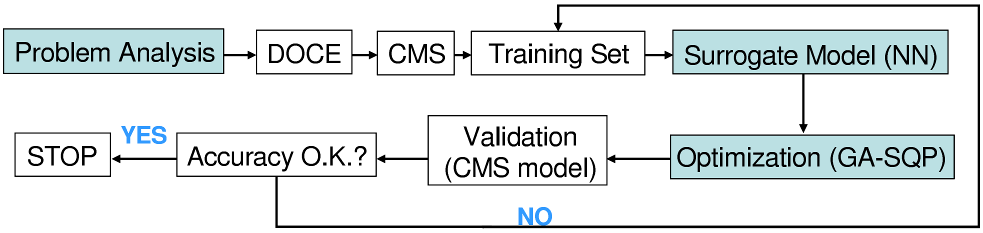


Figure 2: The design optimization strategy.

to select as many sample points as wanted. Therefore, in order to extract more information about the general response trend, it is required to select these limited amounts of points from good locations of the design space. At this point, it is crucial to make the distinction between the Classical Design of Experiments (CDOE) and the *Design of Computer Experiments* (DOCE). CDOE is based on laboratory experiments and random error exists in these experiments. On the other hand, DOCE is based on computer simulations which are deterministic. In other words, no matter how many times the same simulation is run, the results are always the same. Additionally, unlike CDOE, DOCE is based on the assumption that the true response trend is unknown. Thus for extracting more information about the trend, the main objective of DOCE is to distribute the sample points all over the design space. In this research this is done using the *Latin Hypercube Sampling* [11].

After generating the sample points, the next step is finding the response of the FE model for each of these points. This step could be the most time consuming step of the overall design optimization strategy based on the complexity of the FE model. For certain applications, using CMS may cause a lot of reduction in computation time. In this research, the attention is paid on structures which have repeating patterns in their design. Reduced order FE models of these structures are obtained using the Craig-Bampton technique. For repeating patterns only one repeating pattern is taken into account in the calculations.

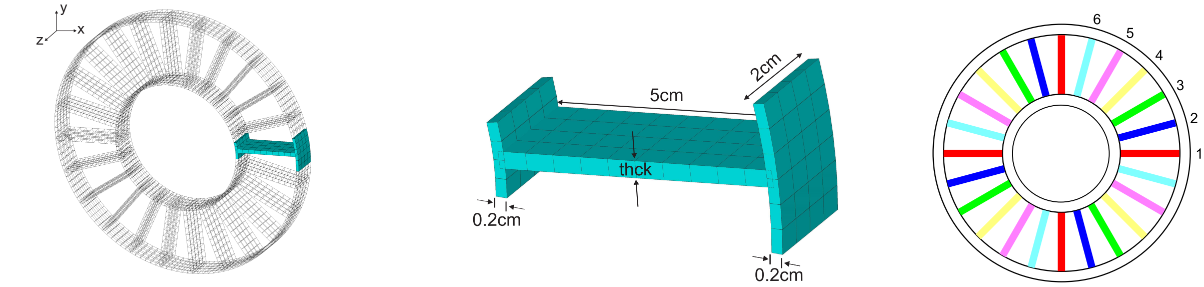
At the end of the previous step, the training set is generated. Hence, a surrogate model can be found using Backpropagation NNs. Then, it replaces the CMS based FE model in the optimization problem.

Next, the optimization is performed using first GA. Then its solution is provided as an initial point for SQP for finding an exact solution. Therefore, the chance of obtaining an exact global optimum solution is increased.

As it is mentioned earlier, there is no flexibility to select as many sample points as wanted at the beginning of the strategy. Since the obtained surrogate model is based on that limited amount of data, it may not represent the actual trend well. When the problem is optimized using that surrogate model, the attained results may not be trustable. Hence, it is very important to validate the response of the surrogate model with the response of the CMS based FE model. This is done at the end of the optimization step. The CMS based FE model is run for the optimum design parameters. If its response compromises with the response of the surrogate model, the scheme is stopped. Otherwise, it is an indication of a poor surrogate model. The optimum design parameters and the corresponding CMS based FE model response are added to the training set, NNs are trained again for obtaining a better surrogate model. The same procedure is followed until the error between the CMS based FE model result and the surrogate model result is small enough.

6. Demonstration of the Strategy

For the demonstration of the introduced strategy, a structure which resembles a fan inlet case is selected. The structure and its repeating component are illustrated in Figure 3a, the physical and the design parameters of the component are shown in Figure 3b. The thicknesses *thck* of the struts are selected as design parameters and the struts which have $n\frac{\pi}{2}$, $n = 0, \ldots, 3$ rotational distance between each other are assumed to have the same thickness values. In Figure 3c, the identical colors represent the struts that have the same thickness values. Since there are 24 struts on the structure, there exit 6 design parameters as total which are also summarized in Figure 3c via numbering. The structure is a free-free structure.



(a) The selected structure and its repeating component

(b) The physical and the design parameters of one component.

(c) The struts with same colors have the same thickness values.

Figure 3: The selected structure for the demonstration of the strategy.

For the same parameter values all the substructures are identical in the local coordinates. Thus, the Craig-Bampton transformation, stiffness and mass matrices of each substructure are all the same. Consequently, the reduced FE model of the entire structure can be obtained using the reduced FE model of one repeating component.

In this study, the reduced system matrices of a selected repeating component are generated using the Craig-Bampton method for different design parameter values in ANSYS. Assigning these matrices to the rest of the substructures via multiplying them with the corresponding rotation matrices, assembling the substructure system matrices for each design parameter configuration and solving the eigenvalue problem are performed in MATLAB. In the FE model, *Shell181* elements are used which are suitable for analyzing thin to moderately thick shell structures. Each element has 6 d.o.f at each node which are the translations and rotations on the x, y, z coordinates. The in-plane vibrations are the only concern for this problem. Therefore, the rotations on x, y axes and the translations on z axes are suppressed in the element. The selected material properties are as follows: Young's modulus (E) is 116 GPa., Poisson's ratio (ν) is 0.3 and the density (ρ) is 4.5 gr/cm^3.

In the initial design, the thicknesses of the struts are selected as $thck_i = 0.3$ cm, $i = 1, 2, \ldots, 6$. Thus, the total mass of the structure is 0.4936 kg and the 5^{th} natural frequency (2^{nd} bending frequency) is 702.23 Hz. with a mode shape illustrated in Figure 4a. Because of the fact that the structure is a free-free structure, the first three modes of the structure are rigid body modes.

For the structural optimization problem; the total mass of the entire structure is desired to be minimized by adjusting the thicknesses of the struts while increasing the 5^{th} natural frequency from 702.23 Hz. to 750 Hz. and preserving the 5^{th} mode shape of the initial design.

The optimization problem is formulated as follows:

$$\begin{aligned} \min_{\text{thck}_i} \quad & \rho V(\text{thck}_i) \\ \text{sbj. to} \quad & f_5 = 750 \\ & \text{MAC}_5 \geq 0.9 \\ & 0.1 \leq \text{thck}_i \leq 0.5 \qquad i = 1, \ldots, 6. \end{aligned} \tag{14}$$

In Eq.(14), V represents the volume of the entire structure which is a function of the design parameters thck$_i$, $i = 1, 2, \ldots, 6$. In order to keep the mode shape of the initial design the same, the *Modal Assurance Criteria* (MAC) is used to check the correlation between the 5^{th} eigenvector of the initial design and the 5^{th} eigenvector of the current design.

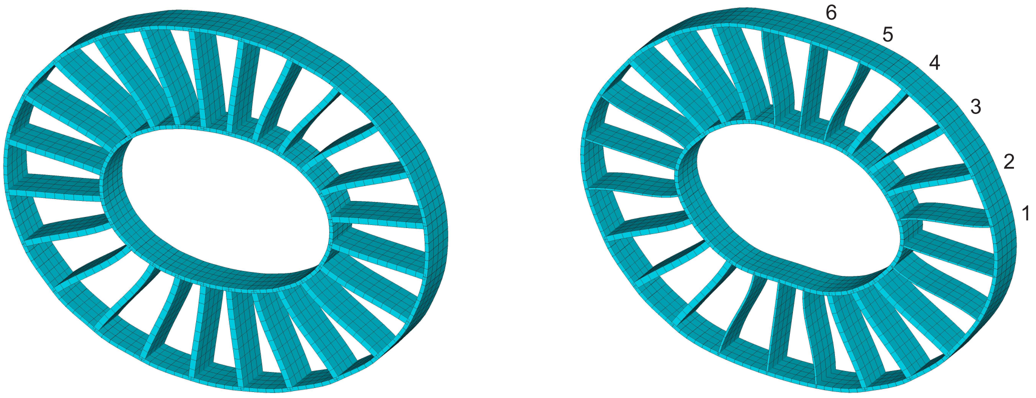
The MAC is a scalar value between 0 and 1, representing the correlation number between two mode shapes. A MAC value near 1 indicates a high degree of correlation between two mode shapes. If $\mathbf{u}$ and $\mathbf{v}$ are assumed to be two eigenvectors their MAC value is

$$\text{MAC} = \frac{(\mathbf{u} \cdot \mathbf{v})^2}{\|\mathbf{u}\|^2 \|\mathbf{v}\|^2} \tag{15}$$

where ”·” is the dot product. As it might be realized, MAC is nothing but the square of the cosine of the angle between two vectors.

Two NN surrogate models with 25 hidden layer neurons are employed in the optimization problem which take the place of f_5 and MAC_5. For generating a training set for surrogate modeling only one repeating component is used. First, a 60×1 DOCE set, $\mathbf{D}_1$, is generated for the varying thickness values of that component. Then, the Craig-Bampton stiffness and mass matrices are calculated for each element of this set. The DOCE set $\mathbf{D}_1$ and its corresponding system matrices are stored in a library for later use. The next step is sharing the obtained system matrices with the rest of the substructures. Since there are 6 design parameters in the overall structure, the 60×6 DOCE set, $\mathbf{D}_T$, is generated where each column of $\mathbf{D}_T$ is the permuted version of $\mathbf{D}_1$. Therefore each row of $\mathbf{D}_T$ represent one possible design configuration for the entire structure. Due to the fact that the system matrices of each configuration have already been calculated and stored in a library, it is only required to call the system matrices from the library, multiply them with the corresponding rotation matrices for locating them to their global coordinate positions, assemble them and solve an eigenvalue problem. At the end of the solution process a set of eigenvalues for the 5^{th} natural frequency and the corresponding eigenvectors are obtained. $\mathbf{D}_T$ and the frequency set are used for training the NN which is taking the place of f_5. The correlation between the computed eigenvectors and the 5^{th} eigenvector of the initial design is calculated using Eq.(15) and a set of MAC values is obtained. Afterwards, $\mathbf{D}_T$ and the MAC set are used for training the NN which is taking the place of MAC_5.

In the validation step, before generating the CMS based FE model for the optimum design parameter values, the system matrices of each substructure configuration is first looked for in the library. If they already exist in the library, they are called from there. Otherwise they are generated and saved in the library. Then all the system matrices of the optimum configuration are gathered in the global coordinates and solved. The obtained solutions are compared with the results of the two NNs. If the relative error between them is smaller than 0.005 for each case, the procedure is stopped else it is continued until the relative error is smaller than the desired value.



(a) Initial design

(b) Optimum design

Figure 4: The initial and the optimum designs and the corresponding 5^{th} mode shapes.

The results of the optimization problem are summarized in Table 1. The optimum design and its 5^{th} mode shape are illustrated in Figure 4b. As it might be realized, in the optimum design the thicknesses of the struts at the bending locations are reduced to their minimum limits for the sake of reducing the total mass and the thicknesses of the rest of the struts are adjusted in order to satisfy the defined constraints.

7. Conclusions

It is becoming a common practice to use surrogate models instead of FE models in the design optimization process. On the other hand, FE models are still required to gather a training set for surrogate modeling. For certain applications using CMS in FE modeling may cause a lot of reduction in computation time. In this research, the benefits of CMS is utilized for the optimization of structures which have repeating patterns. In the introduced design optimization strategy, the Craig-Bampton method is

Table 1: Summary of the Optimization Problem

Initial Design Parameters	[0.3 0.3 0.3 0.3 0.3 0.3]	MAC_5(NN)	0.9747
Optimum Design Parameters	[0.1 0.2823 0.3267 0.3176 0.1 0.1]	MAC_5(CMS)	0.9759
# of designs in the Library (Initial)	60	Final f_5(NN)	750 Hz.
# of designs in the Library (Final)	150	Final f_5(CMS)	749.88 Hz.
Initial Mass	0.4936 kg.	Optimum Mass	0.3904 kg.

used for reducing the FE model. Additionally, only one repeating pattern is modeled using that method and the calculated system matrices are utilized for the rest of the repeating patterns. Therefore, extra calculations for obtaining the system matrices of each repeating pattern is avoided which may cause a significant decrease in computation time. Backpropagation NNs with Bayesian regularization is employed for surrogate modeling. The strength of this method shows itself when there is no idea about the nonlinearity of the input-target relationship. The two step optimization strategy increases the chance of finding an exact global optimum. The introduced design optimization strategy is demonstrated on a problem where the structure has repeating patterns. The results indicate that the suggested strategy is performing well and very promising for real life applications.

8. References

[1] J.W. Wind, D. Akçay Perdahcıoğlu and A. de Boer, Distributed Multilevel Optimization for Complex Structures, *SMOJ - Structural and Multidisciplinary Optimization Journal*, 2007.

[2] R.R. Craig and M.C.C. Bampton, Coupling of Substructures for Dynamic Analysis, *AIAA - American Institute of Aeronautics and Astronautics*, 1968, 6(7), 1313-1319.

[3] K. Hornik, Multilayer Feedforward Networks are Universal Approximators, *Neural Networks*, 1989, 2, 359-366.

[4] Hagan M T, Demuth H B and Beale M. Neural Network Design. *Boston: PWS Publishing Company*, 1996.

[5] D.J.C. Mackay, Bayesian interpolation, *Neural Computation*, 1992, 4, 415-447.

[6] Foresee F D and Hagan M T. Gauss Newton Approximation to Bayesian Learning. *IEEE Trans. On Neural Networks - Proceedings of ICNN 97*, 1997, 3, 1930-1935.

[7] A.R. Conn, N.I.M. Gould and P.L. Toint, A Globally Convergent Augmented Lagrangian Algorithm for Optimization with General Constraints and Simple Bounds. *SIAM Journal of Numerical Analysis*, 1991, 28(2), 545-572.

[8] N.I.M. Gould, A.R. Conn and P.L. Toint, A globally Convergent Lagrangian Barrier Algorithm for Optimization with General Inequality Constraints and Simple Bounds. *Mathematics of Computation*, 1997, 66(217), 261-288.

[9] P.E. Gill, W. Murray, M.A. Saunders and M.H. Wright, Procedures for Optimization Problems with a Mixture of Bounds and General Constraints. *ACM Transactions on Mathematical Software*, 1984, 10(3), 282-298.

[10] S.P. Han, A Globally Convergent Method for Nonlinear Programming. *Journal of Optimization Theory and Applications*, 1977, 22(3), 297-309.

[11] A.A. Giunta, S.F. Wojtkiewicz Jr. and M.S. Eldred, Overview of Modern Design of Experiments Methods for Computational Simulations. *AIAA - American Institute of Aeronautics and Astronautics*, 2003, 2003-0649.