

An artificial neural network residual kriging based surrogate model for curvilinearly stiffened panel optimization

Mohammed R. Sunny^{*2}, Sameer B. Mulani^{3a}, Subrata Sanyal^{4b}
and Rakesh K. Kapania^{1c}

¹Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, Blacksburg, USA

²Department of Aerospace Engineering, Indian Institute of Technology, Kharagpur 721302, India

³Department of Aerospace Engineering and Mechanics, The University of Alabama, Tuscaloosa, AL 35487, USA

⁴Measurement Science and Engineering Department, Naval Surface Warfare Center (NSWC), Corona Division, P.O. Box 5000, Corona, CA 92878, USA

(Received January 1, 2016, Revised May 26, 2016, Accepted June 7, 2016)

Abstract. We have performed a design optimization of a stiffened panel with curvilinear stiffeners using an artificial neural network (ANN) residual kriging based surrogate modeling approach. The ANN residual kriging based surrogate modeling involves two steps. In the first step, we approximate the objective function using ANN. In the next step we use kriging to model the residue. We optimize the panel in an iterative way. Each iteration involves two steps-shape optimization and size optimization. For both shape and size optimization, we use ANN residual kriging based surrogate model. At each optimization step, we do an initial sampling and fit an ANN residual kriging model for the objective function. Then we keep updating this surrogate model using an adaptive sampling algorithm until the minimum value of the objective function converges. The comparison of the design obtained using our optimization scheme with that obtained using a traditional genetic algorithm (GA) based optimization scheme shows satisfactory agreement. However, with this surrogate model based approach we reach optimum design with less computation effort as compared to the GA based approach which does not use any surrogate model.

Keywords: surrogate model; optimization; artificial neural network; kriging; stiffened panel

1. Introduction

Design optimization of aerospace systems usually involves high-fidelity analyses and computationally expensive optimization. Surrogate based analysis and optimization (SBAO) is an effective means to tackle this issue. The basic idea of using surrogate models (also called meta

*Corresponding author, Assistant Professor, E-mail: sunny@aero.iitkgp.ernet.in

^aAssistant Professor, E-mail: sbmulani@eng.ua.edu

^bChief Scientist, E-mail: subrata123@msn.com

^cMitchell Professor of Aerospace and Ocean Engineering, E-mail: rkapania@vt.edu

models) is to replace the high fidelity, expensive analysis code with a less expensive approximate model. To obtain an accurate approximation, it is necessary to have a good training set and a proper surrogate. Much of the research on surrogates has been done in these two fields, viz., design of experiments and selection and validation of surrogate models. While Queipo *et al.* (2005), Forrester and Keane (2009), and Jones (2001) have reviewed the state of the art in surrogate model based design optimization in general; Ahmed and Qin (2009) have presented a review of the literature related to its application in the field of aerodynamics. Surrogate models can be broadly classified into two categories-non interpolating and interpolating. Examples of non interpolating (also known as smoothing) surrogate models are polynomial response surface (PRS) and artificial neural networks (ANN). Both of these models are fitted by minimizing the sum of the squared errors at a finite number of selected data points. At the data points, one may not get exact match between the predicted and actual value. Examples of interpolating surrogate models include radial basis neural network (RBNN) and kriging. Response surfaces created by an interpolating surrogate models passes exactly through the data points used for fitting the model.

The development of PRS was initiated by Box and Wilson (1951), followed by a comprehensive account of the methods of using PRS by Myers and Montgomery (1995). However, it has been shown to be unsuitable for highly nonlinear and irregular performance problems (Venter *et al.* 1998, Vavalle and Qin 2007). Several investigations have been carried out to increase the accuracy of the PRS for multi disciplinary optimization (Balabanov *et al.* 1999), by resizing the design space. To overcome these limitations of PRS, alternative surrogates like kriging and radial basis functions were proposed. Guinta (1997) has shown that quadratic regression polynomial gave better results than kriging. However, Wang *et al.* (2008) concluded that kriging is superior in nonlinear problems than PRS.

An ANN model is a nonparametric regression method. This method utilizes the functional concept of neurons in the brain (Daberkow and Mavris 1998). An ANN is composed of neurons which can be represented as nonlinear transfer functions of the inputs. The main advantages associated with ANN are that: i) it can learn from example and perform the required task, ii) it can create its own organization or representation of the information provided to it, iii) it can retain some of its capabilities even with major network damage. ANN based surrogate models have been successfully used by researchers like Welch *et al.* (2003), Zaabab *et al.* (1995) etc. Daberkow and Mavris (1998) critically evaluated the use of ANNs as meta models in aircraft design application, and contrasted this with the RSM method. They found that RSM worked better than ANNs for a problem with few variables and small ranges, but where a large number of variables or extended ranges are involved, ANNs yield a better approximation. Kiranyaz *et al.* (2009) devised a novel method for automatic generation of ANNs based on a multi dimensional Particle Swarm Optimization. However as ANN is a universal approximator and models the overall nonlinear pattern, in case of the optimization problems where the variation of the objective function with the design variables show very high nonlinearity, an ANN by itself may not be sufficient.

Kriging was formally developed by Matheron (Matheron 1963), and was named after a South African mining engineer. Kriging considers the value of a function as a sum of a general trend (could even be a constant) and a systematic departure. Demonstration of the efficiency of kriging for modeling and optimization of deterministic functions can be found in Sacks *et al.* (1993) and Sacks *et al.* (1989). Jones and Schonlau (1998) investigated the efficacy of kriging by employing the DACE stochastic model for constructing a global optimization algorithm. Attempts have also been made to increase the performance of kriging by using approaches like Analysis of Variance (ANOVA) (Booker *et al.* 1998) and gradient data (co-kriging) (Toal and Keane 2011). Kriging,

along with Gaussian functions, is a special type of Radial Basis Function (RBF) method, which uses linear combination of radially symmetric functions to approximate response functions (Keane 2004). RBFs have a special feature that their response decreases (or increases) monotonically with distance from a central point. Michler and Heinrich (2012) have used RBFs to create surrogate models for simulating a fighter aircraft in trimmed state.

Demyanov *et al.* (1998) and Shen *et al.* (2004) have proposed the use of a hybridized version of ANN and kriging to utilize the benefits of both these models. Demyanov *et al.* (1998) used the ANN residual kriging model to model climatic data. There are various methods for selecting training points while developing a surrogate model. These include classical methods like full-factorial design (FFD), partial factorial design (PFD), face-centered cubic (FCC), central composite design (CCD) and D-optimal design. They are easy to implement, but number of points increases rapidly with factors and levels. To counter these difficulties, Space Filling Designs are used. In these designs, points tend to uniformly cover the entire design space. They are used where deterministic errors are expected such as in computer experiments. Latin Hypercube Sampling (LHS) (McKay *et al.* 1979) and Orthogonal arrays (OA) (Hedayat *et al.* 1979) employ space filling designs. OA produces uniform design but can generate particular forms of point replication and is sometimes inflexible while LHS does not produce point replicates but is not uniform. To address these concerns, OA-based LHS (Leary *et al.* 2003, Ye 1998) and other optimal LHS schemes (Palmer and Tsui 2001) have been proposed. The process of selecting a particular surrogate is called validation, in which the generalization error estimates are used to assess the quality of the surrogate and select a model for analysis and optimization. Split sample (SS), Cross Validation (CV) (Myers and Montgomery 1995) and Bootstrapping (Efron 1983) are some of the different schemes that can be used to validate a model.

In our optimization problem, we have optimized the shape and size of a stiffened panel with curvilinear stiffeners to minimize its mass keeping its buckling parameter, crippling parameter and von Mises stress parameter within the permissible range. To perform this optimization, we divided the optimization process into two steps. In the first step, we optimize the shape to minimize the buckling parameter. In the second step, we fix the shape and minimize the mass. Variation of the buckling parameter with the design variables is observed to be highly nonlinear. This makes PRS and ANN inefficient for this problem. Following the recommendation of Demyanov *et al.* (1998), we have used ANN residual kriging for this problem. Among the different sampling methods, we found LHS to be sufficient for application to our problem. However, for higher number of sample points with higher number of design variables, we observed LHS and other sampling methods to be time consuming. So, we designed our own random sampling algorithm that meets our needs.

This work is a part of our research on developing a framework for design optimization of a stiffened plate with curvilinear stiffeners. We call this framework EBF3PanelOpt. In this paper, a brief description of EBF3PanelOpt is given first. After that, the optimization problem under consideration has been explained. Next, descriptions of ANN, kriging, and ANN residual kriging have been given. After that, our two step optimization process based upon the artificial neural network residual kriging has been explained. We first evaluate the performance of the proposed approach by applying it to optimize the modified Rosenbrock's function to show the robustness of surrogate model based optimization scheme. Next, we considered two example problems involving mass minimization of a stiffened panel and solved using our proposed artificial neural network residual kriging based surrogate model and traditional GA. Comparison of the two methods has been shown to prove the efficiency of our proposed artificial neural network residual kriging based surrogate model.

2. Motivation

The motivation behind this research came from our need to increase the computational efficiency of the optimization process in EBF3PanelOpt. EBF3PanelOpt developed by the Unitized Structures Research Group at Virginia Polytechnic Institute and State University is a framework for analysis and design of stiffened plates with curvilinear stiffeners. Below is a brief description of the EBF3PanelOpt framework.

Msc.Patran (geometry modeling) and Msc.Nastran (finite element analysis) are integrated in the optimization framework, EBF3PanelOpt (Mulani *et al.* 2010, Mulani *et al.* 2013, Mulani *et al.* 2012) using an object oriented script written in Python. EBF3PanelOpt could be coupled with any optimizer to minimize the mass of the curvilinear blade-stiffened panel subjected to constraints on yielding, buckling, and crippling or local failure of the panel. In EBF3PanelOpt, the panel geometry/shape is defined using third-order interpolating B-spline with the help of eight points. The perimeter of panel is defined as one single curve by connecting four edges. The stiffener curve end-points are created by interpolating this perimeter curve. The stiffener curve is represented using third-order uniform rational B-spline using two end-points and a control point, so the stiffener always remains in the panel area. The control-point is defined using interpolation of the panel surface, so the control-point's x co-ordinate and y co-ordinate have values between 0 and 1. Apart from the definition of stiffener curve points, panel thickness, stiffener thickness, and stiffener height are used as design variables. For two and four blade stiffened panel with uniform panel thickness, the panel has 13 and 25 design variables, respectively.

For this work EBF3PanelOpt is utilized along with genetic algorithm (GA) for optimization. It was observed that optimization using GA when performed without any surrogate model requires very high number of evaluation. To address this issue, we developed a surrogate model based optimization scheme that requires less CPU time.

3. Description of the optimization problem

The optimization problem is formulated as

$$\begin{aligned} \min_{\mathbf{x}} f_0(\mathbf{x}) \\ f_i(\mathbf{x}) \leq 1, \quad i = 1, \dots, m \\ A_j \leq x_j \leq B_j, \quad j = 1, \dots, n. \end{aligned} \quad (1)$$

Here, the design variables x_i ($i=1, \dots, m$) define the shape, the heights of stiffeners, the thicknesses of the stiffeners, and the thickness of the plate. The objective function f_0 is the mass of the plate, constraints f_i ($i=1, \dots, 3$) are the buckling parameter, crippling parameter, and the von Mises stress parameter. Buckling parameter is defined as the reciprocal of the fundamental buckling eigenvalue, von Mises stress parameter is the ratio of the aggregated von Mises stress Kreiselmier-Steinhauser criteria and yield stress.

There are total $6N_{st}+1$ design variables, where N_{st} is the number of stiffeners. Among them, the first $4N_{st}$ variables x_i ($i=1, \dots, 4N_{st}$) define the shape of the stiffeners, whereas the rest $2N_{st}+1$ variables x_i ($i=4N_{st}+1, \dots, 6N_{st}+1$) define the thickness and height of the stiffeners and the thickness of the plate.

To describe the design variables x_i , let us take an example of a plate with 2 curvilinear

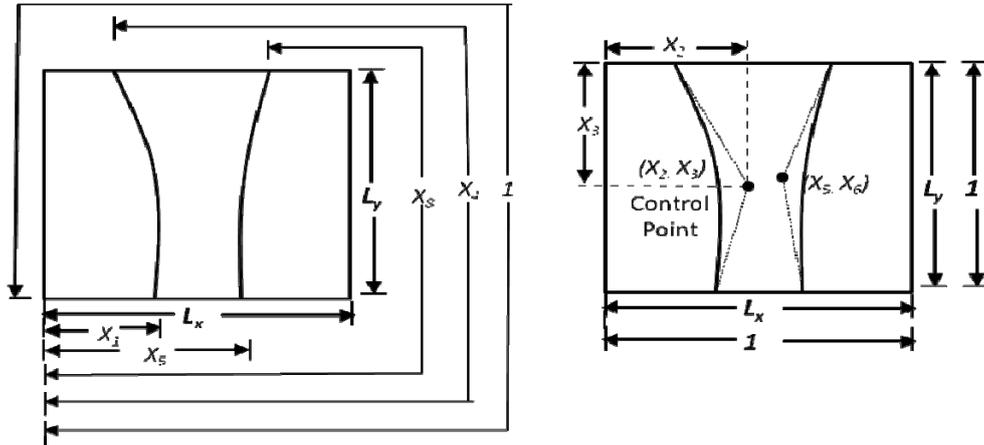


Fig. 1 Top view of a plate with two curvilinear stiffeners

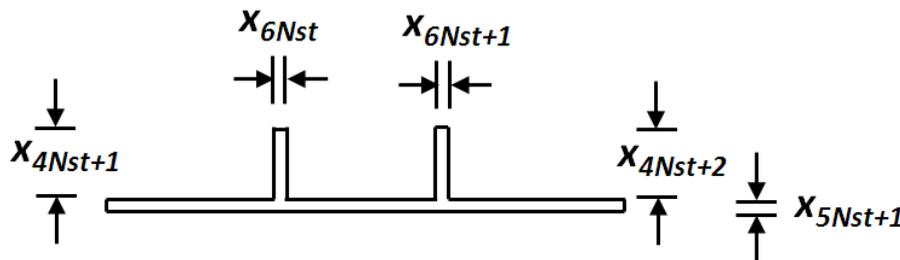


Fig. 2 Front view of a plate with two curvilinear stiffeners

stiffeners. Fig. 1 shows the view of the plate from the top. For this plate we have eight shape variables x_i ($i=1,..,8$) and five size variables x_i ($i=9,..,13$).

Variable $x_{4(i-1)+1}$ and x_{4i} ($i=1,2$) denote the beginning and end points of the i th stiffener, whereas variables $x_{4(i-1)+2}$ and $x_{4(i-1)+3}$ ($i=1,2$) denote the two control points used to define the shape of the i th stiffener. These variables have values normalized with respect to the perimeter of the plate as shown in Fig. 1.

Among the size variables, the first N_{st} variables x_i ($i=4N_{st}+1,.., 5N_{st}$) denote the height of the stiffeners, x_i ($i=5N_{st}+1$) denote the thickness of the plate and the last N_{st} variables x_i ($i=5N_{st}+2,.., 6N_{st}+1$) denote the thickness of the stiffeners, as described in Fig. 2.

Parameters A_j and B_j define the bounds of the design variables. Values of these parameters are $A_j=0$ ($j=1,..,4N_{st}$), $A_j=0.01$ ($j=4N_{st}+1,..,5N_{st}$), $A_j=0.001$ ($j=5N_{st}+1,..,6N_{st}+1$), $B_j=1$ ($j=1,..,4N_{st}$), $B_j=0.06$ ($j=4N_{st}+1,..,5N_{st}$), $B_j=0.005$ ($j=5N_{st}+1,..,6N_{st}+1$).

4. Artificial neural networks

An ANN is a soft computing system that follows the principle of the operation of biological neurons. It is mostly used for pattern recognition, classification etc. because of its ability to capture and represent complex input/output relationships (Sunny and Kapania 2013, Kim and Kapania

2003). The building blocks of ANN are interconnected processing elements known as the neurons. Like biological neural networks, ANNs acquire knowledge through learning and store knowledge within inter-neuron connection strengths known as synaptic weights. The values of connections (weights) between different neurons are adjusted to enable the neuron to perform a specific task. Figure 3 shows the structure of an ANN with one layer of neurons. The input layer takes the input a_j ($j=1,..4$). Each element of the input layer is connected to the neurons in layer-1. Each connection between an input element a_j and the i th neuron is associated with a weight w_{ij} . Apart from that, the i th neuron may have a bias b_i . Output from the i th neuron is the sum total of weighted inputs and bias operated by an activation function f_i . An activation function can be a linear function, the step function, the sigmoid function etc. The output from each neuron can be written as

$$O = f_i \left(\sum_{j=1}^m w_{ij} p_j + b_i \right) \quad (2)$$

A neural network can have a single (or multiple) layers. Depending on the pattern of connections between different layers neural networks can be of two types-feed forward networks and recurrent networks. In the feed forward networks, the data flows from input to the output layers, but there is no feedback connection i.e., there is no flow of data from output to the input layer. Recurrent networks have feedback connections. In some cases, they have dynamic properties i.e., the activations undergo relaxation and evolve to a stable state.

The weights and biases of an ANN are adjusted so that a given set of input produces a desired set of output. This is known as the training of an ANN. Training procedures can be divided into two categories-supervised training and unsupervised training. In supervised or associative training, an ANN is provided with a set of input-output pairs. This set of input-output pairs is called the training set. In unsupervised or self organization training, a neural network is trained to respond to a cluster of patterns within the input. Basically, all the training algorithms involving adjustments of weights are variants of the Hebbian learning rule suggested by Hebb in his classic book, *Organization of Behavior* (Hebb 2002). If the i th neuron in the current layer receives an input x_j from the j th neuron in the previous layer and outputs x_i , then according to the Hebbian rule, the weight of the connection w_{ij} has to be modified by adding Δw_{ij} to it, where Δw_{ij} is given by the following equation

$$\Delta w_{ij} = \gamma x_i x_j$$

γ is called the learning rate. In another rule, difference between the actual and desired activation are used as the objective function to be minimized for adjusting the weights

$$\Delta w_{ij} = \gamma (y_j - x_j) x_j$$

Here, y_i is the desired activation. This is called Widrow-Hoff rule or the Delta rule. In a network consisting of single layer of neurons with linear transfer functions, the objective function is a linear function of the weights and biases. So, the derivatives of the objective function with respect to the weights and biases can be easily calculated. In a network with multiple layers of neurons with non linear transfer functions, calculations of the derivatives of the objective function with the weights and biases become complex. Back propagation algorithm of learning makes the calculation of these derivatives easier. Detailed discussions on the different learning rules can be found in Hagan *et al.* (1996).

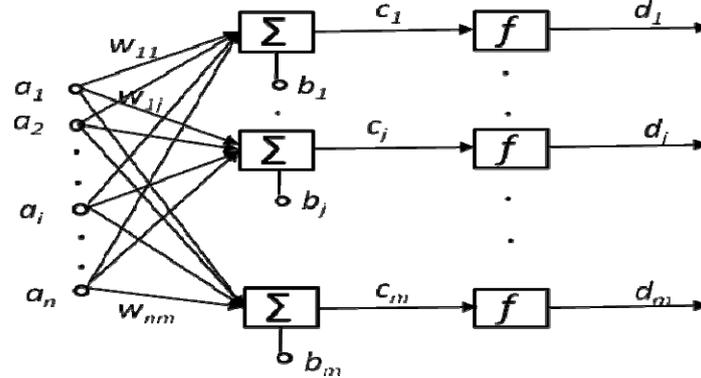


Fig. 3 Architecture of an artificial neural network with one hidden layer

5. The kriging model

Kriging is a way of modeling the relationship between the input variables and the output function as a realization of a stochastic process. Detailed description of this method can be found in Jones (2001). The value of the output for an input vector \bar{x} is modelled as a random process $Y(\bar{x})$ which has a mean μ and variance σ^2 . The correlation between the random variables at the two points \bar{x}_i and \bar{x}_j is given by the relation

$$Corr[Y_i, Y_j] = \exp\left(-\sum_{l=1}^d \theta_l |x_{il} - x_{jl}|^{p_l}\right) \tag{3}$$

Here x_{il} is the l th component of the i th input vector and d is the number of components in each vector.

This is just an example of one specific type of correlation function. One can choose from several other types of correlation functions. All these correlation functions should have two properties common - i) when $\bar{x}_i = \bar{x}_j$, the value of the correlation is 1, and ii) as $|\bar{x}_i - \bar{x}_j|$ tends to infinity, the value of the correlation tends to zero. If we choose the correlation function given by Eq. (3), the parameters θ_l and $p_l(l=1, \dots, d)$ become the unknown parameters associated with the model. Rest of the modeling process involves the determination of values of the parameters θ_l , and $p_l(l=1, \dots, d)$ from a training set. The uncertainty about the function's value at the n training points can be denoted by a vector $Y = \{Y_1, \dots, Y_n\}^T$.

For this vector, the mean and covariance are μ and $cov(Y_i, Y_j) = \sigma R_{ij}$. Where R_{ij} is the correlation between Y_i and Y_j . If the output vector at the n th training point is $\{y_1, \dots, y_n\}^T$, then the likelihood of the observed data at the training points is given by

$$P = \frac{1}{(2\pi)^{n/2} \sigma^n |R|^{1/2}} \exp\left[-\frac{(y - I\mu)' R^{-1} (y - I\mu)}{2\sigma^2}\right] \tag{4}$$

This likelihood is a function of σ and μ , which are function of θ_l , and $p_l(l=1, \dots, d)$. By maximizing P , optimum values of θ_l , and $p_l(l=1, \dots, d)$ are obtained. We denote the optimum values of θ_l , and $p_l(l=1, \dots, d)$ by $\hat{\theta}_l$ and \hat{p}_l . Using this model, the value of output at any point \bar{x}^* is

obtained as

$$y^* = \hat{\mu} + r' R^{-1}(y - I\hat{\mu}) \tag{5}$$

Here, $\hat{\mu}_i$ is the value of μ obtained using the values of $\hat{\theta}_i$ and \hat{d}_i and r' is a vector where the i th element r_i is given by $Corr[Y_i, Y^*]$.

6. The artificial neural network residual kriging

Modelling using the ANN residual kriging method involves two steps-training an ANN and fitting a kriging model to the data. Let us suppose that we have a training set consisting of n input vectors \bar{x}_1 to \bar{x}_n and n outputs y_1 and to y_n .

In the first step, we train an ANN using this training set. The architecture of the ANN depends on the user's choice. Using the ANN, we predict the values of the output at the training points using the trained ANN. Let us denote the ANN predicted output at the i th training point as y_{ni} . Next, we find out the residue $y_{ri}=y_i-y_{ni}$ at all the data points ($i=1,\dots,n$). We define a new training data set that has \bar{x}_1 to \bar{x}_n as the inputs and y_{r1} to y_{rn} as the outputs. By using this new training data set, we fit a kriging model. This completes the development of an ANN residual kriging model. Total output at a point \bar{x}^* from this model will be the sum of the output obtained from the ANN and the kriging model.

7. The two step optimization process

The surrogate model based optimization was performed in an iterative way. Each iteration involves two steps. In the first step, the size variables are fixed and the buckling parameter is minimized by considering the shape variables as the design variables. In the second step, the shape variables are fixed at the optimum values obtained in the first step and the mass is minimized

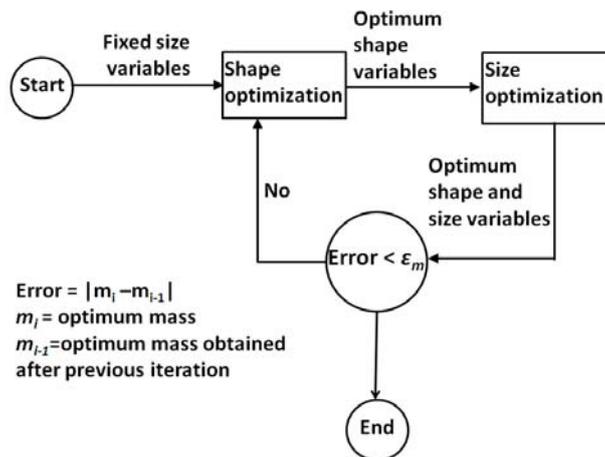


Fig. 4 The two step optimization algorithm

keeping the three constraints - buckling parameter, von Mises stress parameter and crippling parameter within the permissible values by considering the size variable as the design variables. Afterwards, we fix the size variable at the optimum values obtained by performing the size optimization at the previous iteration and perform the shape optimization. Again we fix the shape variables at the optimum values and perform the size optimization. The iteration continues until the difference in the values of the optimum mass obtained at two successive iterations is less than a tolerance ε_m defined by the user. Fig. 4 shows the schematic view of this iterative optimization procedure. In the first step, we consider the whole design space for both shape and size optimization. In each successive iteration, we consider a smaller design space around the optima obtained in the previous iteration for both shape and size optimization.

8. The optimization process using the surrogate model based approach

8.1 Sampling algorithm

Both the shape and size optimization steps involve random sampling. Before describing each of the shape and size optimization steps it is necessary to describe the sampling algorithm used in those steps. According to our sampling algorithm, the j th element of the i th sample is

$$x_{sij} = (R_{nd} + (R_{shj} - 1)) / n_s$$

Here, n_s is the total number of samples generated. Here R_{nd} is a random quantity between 0 and 1. R_{sh} is an array that contains a random combination of the integers ranging from 1 to n_s . The j th element of this array is R_{shj} .

8.2 Modified rosenbrock's function

Before applying our optimization scheme to a complex problem involving shape and size optimization, we evaluated its performance on a standard bench-mark problem, optimization of the modified Rosenbrock's function. The optimization problems involves the minimization of a modified version of the classical Rosenbrock's function (Rosenbrock 1960, Perex *et al.* 2012) with N dimensional constraint. The optimization problem can be formulated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{i=1}^n \left(100(x_i - x_{i+1})^2 + (1 - x_i)^2 \right)^2 \\ & \sum_{i=1}^n \left(0.1 - (x_i - 1)^3 - (x_{i+1} - 1) \right) \leq 0 \\ & -5.12_j \leq x_j \leq 5.12_j, \quad j = 1, \dots, N. \end{aligned} \quad (6)$$

Here, N is the number of design variables considered. This optimization was performed in one step i.e., we did not have to break the optimization in different steps unlike the optimization of the stiffened plate described in the next two sub sections. For the surrogate model based optimization, the objective function was modified as shown in Eq. (7) to account for the constraint

$$f_{\text{mod}} = f_0 + w_1 f_1 H(f_1) \quad (7)$$

Here, f_0 is the objective function and f_1 is the nonlinear constraint shown in Eq. (6). The function $H(f_1)$ has a value zero when $f_1 < 0$ and has a value 1 when $f_1 > 0$.

At first, we generate a set of n_1 sample points using $x = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n]$ the random sampling algorithm described in the previous subsection. After that, we evaluate f_{mod} at all the sample points. We define the set $y = \{y_1, y_2, \dots, y_{n1}\}$, where y_i is the value of f_{mod} at the i th sample point. Next, we develop an ANN residual kriging model considering the pair of x_z and y_z as the training set. Now, we find the sample point that has the minimum value of y . Let us denote this value as y_{min} . Next, we generate $N_1 (N_1 \gg n_1)$ set of new sample points. According to Jones (2001), the probability of f_{mod} having a value $(y_{min} - I)$ at any of these new sample points \bar{x} is given by

$$P(I) = \frac{1}{\sqrt{2\pi s(\bar{x})}} \exp\left(-\frac{(y_{min} - I - y(\bar{x}))^2}{2s^2(\bar{x})}\right) \quad (8)$$

Here, I is a variable which can have a real positive value. We call it the probability of \bar{x} having an improvement by a value I over y_{hmin} . By integrating $P(I)$ as given in Eq. (8) in the range $I \in [0, \infty]$, we get the expected value of improvement E at \bar{x} . In this way, we determine E at all of these new sample points and arrange them in descending order. We sort out n_2 points having the higher values E and update the sets x and y . We fit an ANN residual kriging model using this updated sets x and y and find out y_{min} . Again, we generate a new set of N_1 sample points and sort out n_2 points having the higher values E and update the sets x and y . We continue this iteration unless the difference of y_{min} at two consecutive iterations is below a tolerance value ε . The value of y_{min} at the last iteration is the minimum values of y . The corresponding value of f_0 is the minimum value of the objective function.

8.3 Shape optimization

As mentioned before, for shape optimization at each iteration, we keep the size variables fixed. We denote the range of the i th shape variable as $d_{hi} \equiv [d_{hi1}, d_{hi2}]$ ($i=1, \dots, N_h$), where N_h is the number of shape variables. At the first iterative step $d_{hi1}=0$, and $d_{hi2}=1$. At the j th iterative step ($j>1$), $d_{hi1} = \hat{x}_{hi} - \varepsilon_{hi}$ and $d_{hi2} = \hat{x}_{hi} + \varepsilon_{hi}$. Here \hat{x}_{hi} is the optimum value of the i th shape variable obtained in the previous iteration, and $\varepsilon_{hi} \ll 1$ is a user defined value used to define the range of the design variables in each iterative step. We use ε_{hi} ($i=1, \dots, N_h$)=0.1.

Based on some observations, we impose a few constraints on the shape variables. It was observed that when the end points of two stiffeners are very close, the Nastran based FEA fails. To avoid such failures, we considered only the design points where the distance between any two stiffener end points along the edges is more than 0.02 m.

According to the discussion in Section 3, design variables $x_{h4(i-1)+1}$, $x_{h4(i-1)+2}$, $x_{h4(i-1)+3}$, x_{h4i} define the profile of the i th stiffener. Variables $x_{h4(i-1)+1}$ and x_{h4i} denote the two ends of the stiffeners. So, if we interchange the values of the two end points we get the same stiffener. So, we reduced the design space by putting the following constraint

$$x_{h4(i-1)+1} < x_{h4(i-1)+1}; \text{ where } i \leq N_{st}$$

It can also be observed that if the profiles of any two stiffeners are interchanged, the structure of the panel remains the same. Based on this observation, we added the following constraint

$$x_{h4(i-1)+1} < x_{h4i}; \text{ where } i \leq N_{st}-1$$

We also added a few constraints based on the symmetry of the panel. Here, we consider the symmetry in terms of the geometry, material properties, boundary condition and loads. If the panel is symmetric with respect to the X axis, but not symmetric with respect to the Y axis, we can add the following constraint

$$x_1 \leq \frac{L_x + \frac{L_y}{2}}{2(L_x + L_y)}$$

If the panel is symmetric with respect to the Y axis, but not symmetric with respect to the X axis, we can add the following constraint

$$x_1 \leq \frac{L_x}{4(L_x + L_y)}$$

or

$$\frac{L_x}{2(L_x + L_y)} \leq x_1 \leq \frac{1}{2}$$

If the panel has symmetry with respect to both the X and Y axis, we can add the following constraint

$$x_1 \leq \frac{L_x}{4(L_x + L_y)}$$

or

$$\frac{L_x}{2(L_x + L_y)} \leq x_1 \leq \frac{L_x + L_y}{2(L_x + L_y)}$$

At first, we generate a set of n_{h1} sample points $x_h = [\bar{x}_{h1}, \bar{x}_{h2}, \dots, \bar{x}_{hn_{h1}}]$ using a random sampling algorithm. The sample algorithm is described in the Sub-Section A of Section VIII.

Afterwards, we evaluate the samples using EBF3PanelOpt. We define the set $y_h = [y_{h1}, y_{h2}, \dots, y_{hn_{h1}}]$, where y_{hi} is the buckling parameter obtained using EBF3PanelOpt at the i th sample point. Now, we fit an ANN residual kriging model considering the pair of x_h and y_h as the training set. Now, we find the sample point that has the minimum value of buckling factor. Let us denote this value as y_{hmin} . We find out of the expectation (E_h) of improvement over y_{hmin} at all the sample points using the same procedure described in the previous subsection. We sort out first n_{h2} sample points and add to the set x_h . By running EBF3PanelOpt, we obtain the response of the structure at these new sample points and add the values of the buckling parameters to the set y_h . Then we fit an ANN residual kriging model using this updated sets x_h and y_h and find out y_{hmin} . Again, we generate a new set of N_{h1} sample points and sort out n_{h2} points having the higher values E_h and update the sets x_h and y_h . We continue this iteration unless the difference of y_{hmin} at two consecutive iteration steps is below the tolerance value ε_h . The value of y_{hmin} at the last iteration is the minimum values of the buckling parameter and the values of the corresponding shape variables give the optimum shape.

8.4 Size optimization

In each iteration, we fix the values of the shape variables at their optimum values obtained in the shape optimization step and find the optimum values of the size variable that minimize the mass keeping the buckling parameter, crippling parameter and von Mises stress parameter in the permissible range.

We denote the range of the i th shape variable by $d_{zi} \equiv [d_{zi1}, d_{zi2}]$ ($i=1, \dots, N_{sz}$), where N_{sz} is the number of size variables. At the first iterative step, $d_{zi1} = A_i + 2N_{st}$, and $d_{zi2} = B_i + 2N_{st}$. At the j th iterative step ($j > 1$), $d_{zi1} = \hat{x}_{zi} - \varepsilon_{szi}$, $d_{zi2} = \hat{x}_{zi} + \varepsilon_{szi}$. Here \hat{x}_{zi} is the optimum values of the i th size variable obtained in the previous iteration, and $\varepsilon_{szi} \ll 1$ is a user-defined value used to define the range of the design variables in each iterative step. We use $\varepsilon_{szi} (i=1, \dots, N_{sz}) = 0.03$.

The size optimization step involves a constrained optimization where the objective function is mass and the constraints are the buckling parameter, the crippling parameter and the von Mises stress parameter. At first, we convert this constrained optimization problem into an unconstrained optimization problem. We define a new objective function for size optimization as

$$f_{\text{mod}} = f_0 + w_1 f_1 H(f_1) + w_2 f_2 H(f_2) + w_3 f_3 H(f_3) \quad (9)$$

Here, $H(f)$ is function which is zero when $f < 1$ and has a value 1 when $f \geq 1$, $w_i (i=1, \dots, 3)$ are weights. The values of $w_i (i=1, \dots, 3)$ is determined by the user. Here we use $w_i = 1.5$ ($i=1, \dots, 3$). A close look at Eq. (9) shows that by solving an unconstrained minimization problem, by considering y_z as the design variables one can minimize mass keeping the buckling parameter, the crippling parameter and the von Mises stress parameter in the range $[0, 1]$.

We follow the same optimization procedure described in this section to minimize y_z . At first, we generate a set of n_{z1} sample points $x_z = [\bar{x}_{z1}, \bar{x}_{z2}, \dots, \bar{x}_{zn_1}]$ using the random sampling algorithm described in Sub-Section A of Section VII. After that, we evaluate the responses using EBF3PanelOpt and calculate f_{mod} at all the sample points. We define the set $y_z = \{y_{z1}, y_{z2}, \dots, y_{zn_1}\}$, where y_{zi} is the value of f_{mod} at the i th sample point. Next, we fit an ANN residual kriging model considering the pair of x_z and y_z as the training set. Now, we find the sample point that has the minimum value of y_z . Let us denote this value as $y_{z\text{min}}$. Next, we generate N_{z1} set of sample points and following the procedure we find out of the expectation (E_z) of improvement over $y_{z\text{min}}$ at all the sample points. We sort out n_{z2} points having the higher values E_z and update the sets x_z and y_z . We fit an ANN residual kriging model using this updated sets x_z and y_z and find out $y_{z\text{min}}$. Again, we generate a new set of N_{z1} sample points and sort out n_{z2} points having the higher values E_z and update the sets x_z and y_z . We continue this iteration unless the difference of $y_{z\text{min}}$ at two consecutive iteration is below a tolerance value ε_z . The value of $y_{z\text{min}}$ at the last iteration is the minimum values of y_z at that iteration. The corresponding value of mass is the minimum value of mass at that iteration.

9. Results and discussion

9.1 Modified Rosenbrock's function

We optimized the modified Rosenbrock's function assuming $N=2$ using both traditional GA based approach and our proposed surrogate model based approach. The surrogate model based optimization scheme had to evaluate the objective function 3983 times and found the value of the

objective function f_0 to be 0.021. Using the GA, the objective function had to be evaluated 38098 times and the value of the objective function was found to be 0.022.

9.2 Rectangular panel with two stiffeners

We considered the design optimization problem of a rectangular panel with two curvilinear stiffeners. The baseline panel configuration was provided by our industrial partner Lockheed Martin. The panel is a typical aircraft section panel. Generally these panel loads are non-uniform, but for the design purpose, these loads are averaged. The dimension of the baseline panel and the loads at the four edges are shown in Fig. 5. The panel was considered to be simply supported at the four edges.

At first, we solved the optimization problem described in section 3 for this panel using our proposed two step optimization procedure that uses an ANN residual kriging based surrogate model. The values of the parameters ϵ_h , ϵ_z , and ϵ_m were taken to be 0.08, 0.04, and 0.02. To complete this optimization procedure the EBF3PanelOpt had to be run 2846 times. The mass obtained was 1.9962 kg. Now, the same optimization problem is solved using genetic algorithm. The genetic algorithm based optimization process also uses the two step approach described in section VII and uses the same constraints. However, the shape and size optimizations at iteration are performed using the genetic algorithm. For the surrogate model based optimization, we converted the constrained optimization problem into an unconstrained one for size optimization. While using the genetic algorithm, we solved it as a constrained optimization problem that considers mass as the objective function and buckling parameter, crippling parameter, and von Mises stress parameter as the constraints. We used the same values of ϵ_h , ϵ_z and ϵ_m for genetic algorithm based optimization. For this genetic algorithm based optimization, EBF3PanelOpt needed to be run 5381 times. The optimal mass obtained was 2.0137 kg.

Next, we changed the values of ϵ_h , ϵ_z , and ϵ_m to 0.004, 0.004, and 0.004 and optimized the panel using the two approaches. Using the surrogate model based approach, we obtained a design having mass equal to 1.9237 kg by running the EBF3PanelOpt 3895 times. For the GA based approach, EBF3PanelOpt was run 7149 times. The optimum mass obtained was 1.9507 kg. Table 1 shows the designs obtained using the surrogate model based approach and the GA based approach. From these results, it can be observed that the surrogate model approach is giving a better design with less computational effort.

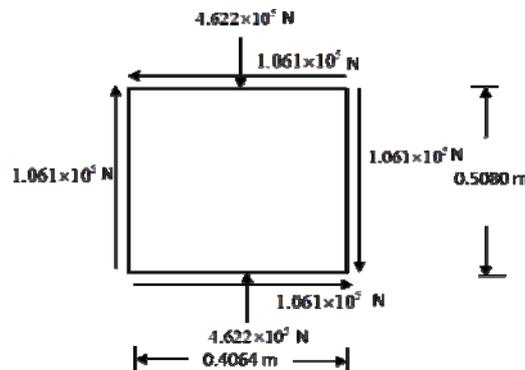
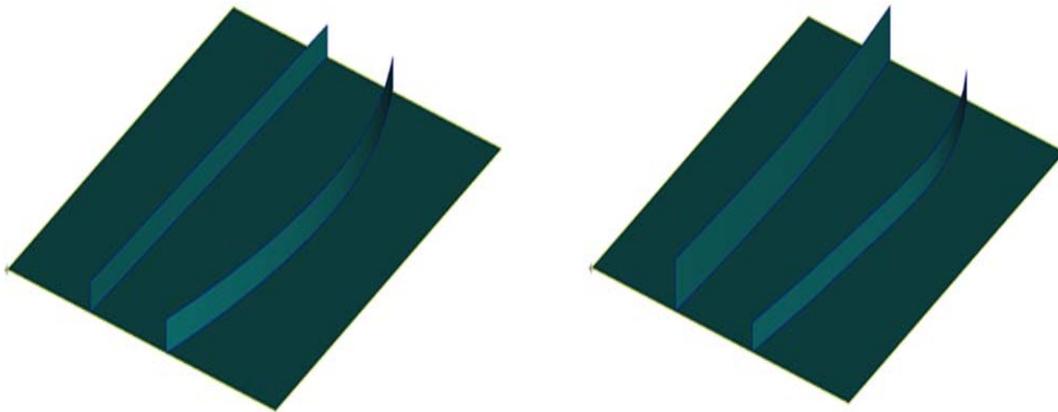


Fig. 5 Baseline panel considered in example problem 1



Mass = 1.9237 Kg

Buckling Parameter = 1.000955

Von Mises Stress Parameter = 0.373

Crippling Stress Parameter = 0.56

(a) Design obtained by our surrogate model based optimization

Mass = 1.9507 Kg

Buckling Parameter = 1.00759

Von Mises Stress Parameter = 0.391

Crippling Stress Parameter = 0.538

(b) Design obtained by GA based optimization

Fig. 6 Designs obtained using the surrogate model and GA based approaches for example problem 1

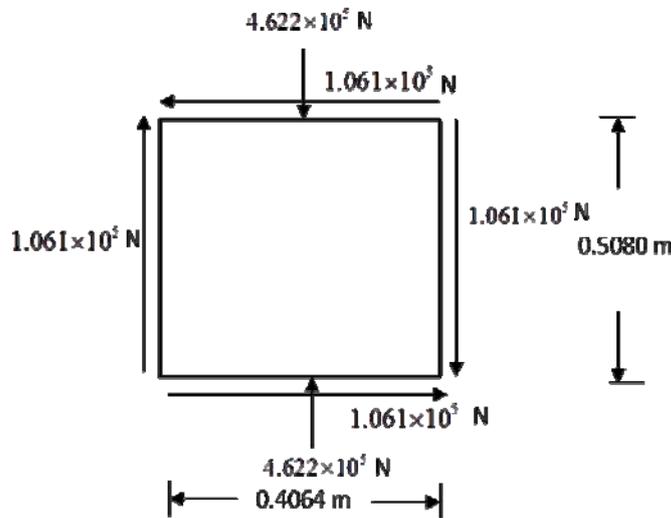
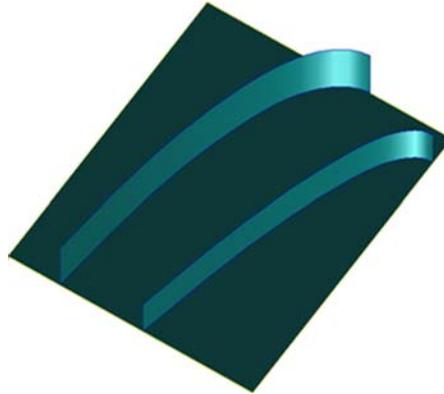


Fig. 7 Baseline panel considered in example problem 2

After that, we changed the loading condition on the panel as shown in Fig. 7 and optimized the panel using our surrogate model based approach and the traditional genetic algorithm based approach. With the surrogate model we had to run the EBF3PanelOpt 3573 times and got an optimum mass of 0.89685 kg. The traditional genetic algorithm based approach had to run the EBF3PanelOpt 6382 times. The optimum mass obtained was 0.9529 kg.



Mass = 0.89685 Kg

Buckling Parameter = 1.00484

Von Mises Stress Parameter = 0.30117

Crippling Stress Parameter = 0.641306

Fig. 8 Design obtained using the surrogate model approach for example problem 2

10. Conclusions

A surrogate model based optimization technique using an ANN residual kriging model has been proposed for the design optimization of stiffened plate with curvilinear stiffeners. The optimization is performed in an iterative way. Each iteration involves shape optimization followed by size optimization using the proposed ANN residual kriging model. Two example problems involving the mass minimization of the stiffened panel with curvilinear stiffeners have been solved using the proposed surrogate model based optimization procedure and the results have been compared with the use genetic algorithm but without the use of any surrogate model. Comparison of the results show that this proposed optimization procedure give better design with less computational effort as compared to the traditional genetic algorithm approach without using any surrogate model. We see the surrogate model approach to be even more efficient than the GA based approach for optimizing the Modified Rosenbrock's function.

Acknowledgments

The work presented here is funded under NASA Subsonic Fixed Wing Hybrid Body Technologies NRA Contract (NASA NNL08AA02C) with Karen M. Brown Taminger as the Associate Principal Investigator and Contracting Office Technical Representative and Richard Keith Bird as the Contract Monitor. We thank them for their suggestions and continuous encouragement. The authors would like to thank our partners in the NRA project, Mr. David Havens, Mr. Robert J. Olliffe, and Dr. Steve Engelstad, all from Lockheed Martin Aeronautics Company of Marietta, GA, for their contribution to the present research. We also want to thank other members of our group, especially Mr. Ved Vyas Duggirala for his help with parallelizing the EBF3PanelOpt.

References

- Ahmed, M.Y.M. and Qin, N. (2009), "Surrogate-based aerodynamic design optimization: Use of Surrogates in Aerodynamic Design Optimization", *13th International Conference on Aerospace Sciences & Aviation Technology*, ASAT- 13 - AE - 14, 1-26.
- Balabanov, V.O., Giunta, A.A., Golovidov, O., Grossman, B., Mason, W.H., Watson, L.T. and Haftka, R.T. (1999), "Reasonable design space approach to response surface approximation", *J. Aircraft*, **36**(1), 308-315.
- Booker, A.J., Dennis, J.E., Frank, P.D., Serafini, D. and Torczon, V. (1998), "Optimization using surrogate objectives on a helicopter test example", *Comput. Meth. Optim. Des. Control*, Boston: Birkhauser, 49-58.
- Box, G.E.P. and Wilson, K.B. (1951), "On the experimental attainment of optimum conditions", *J. Roy. Statist. Soc., Series B*, **13**(1), 1-35.
- Daberkow, D.D. and Mavris, D.N. (1998), "New approaches to conceptual and preliminary aircraft design: a comparative assessment of a neural network formulation and a response surface methodology", *1998 World Aviation Conference*, Anaheim, CA.
- Demyanov, V., Kanevsky, M., Chernov, S., Saveliyeva, E. and Timonin, V. (1998), "Neural network residual kriging application for climatic data", *J. Geographic Inform. Decision Anal.*, **2**(2), 215-232.
- Efron, B. (1983), "Estimating the error rate of a prediction rule: improvement on cross-validation", *J. Am. Statist. Assoc.*, **78**(382), 316-331.
- Forrester, A.I.J. and Keane, A.J. (2009), "Recent advances in surrogate based optimization", *Prog. Aero. Sci.*, **45**(1), 50-79.
- Giunta, A.A. (1997), "Aircraft multidisciplinary design optimization using design of experiments theory and response surface modeling methods", Ph.D. Dissertation, Faculty of Virginia Polytechnic Inst. And State Univ., Blacksburg, VA.
- Hagan, M.T., Demuth, H.B. and Beale, M. (1996), *Neural network design*, PWS Publishing Company.
- Hebb, D.O. (2002), *Organization of behavior: a neuropsychological theory*, L. Erlbaum Associates.
- Hedayat, A., Sloane, N. and Stufken, J. (1999), *Orthogonal arrays: theory and applications*, Springer, Series in Statistics, Berlin.
- Jones, D.R. (2001), "A taxonomy of global optimization methods based on response surface", *J. Global Optimiz.*, **21**(4), 345-383.
- Jones, D. and Schonlau, M. (1998), "Expensive global optimization of expensive black-box functions", *J. Global Optimiz.*, **13**, 455-492.
- Keane, A.J. (2004), "Design search and optimization using radial basis functions with regression capabilities", *ParmeeIC Berlin*, 39-49.
- Kim, Y.Y. and Kapania, R.K. (2003), "Neural networks for inverse problems in damage identification and optical imaging", *AIAA J.*, **41**(4), 732-740.
- Kiranyaz, S., Ince, T., Yildirim, A. and Gabbouj, M. (2009), "Evolutionary artificial neural networks by multi dimensional particle swarm optimization", *Neural Networks*, **22**(10), 1448-1462.
- Leary, S., Bhaskar, A. and Keane, A. (2003), "Optimal orthogonal array based latin hypercubes", *J. Appl. Statist.*, **30**(5), 585-598.
- Matheron, G. (1963), "Principles of geostatistics", *Economic Geol.*, **58**(8), 1246-1266.
- McKay, M., Conover, W. and Beckman, R. (1979), "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code", *Technometrics*, **42**(1), 239-245.
- Michler, A. and Heinrich, R. (2012), "Surrogate-enhanced simulation of aircraft in trimmed state", *Comput. Meth. Appl. Mech. Eng.*, **217-220**, 96-110.
- Mulani, S.B., Slemple, W.C.H. and Kapania, R.K. (2010), "Curvilinear stiffened panel optimization framework for multiple load cases", *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, Fort Worth, Texas.
- Mulani, S.B., Slemple, W.C.H. and Kapania, R.K. (2012), "EBF3PanelOpt: An overview and recent developments of an optimization framework for stiffened panel using curvilinear stiffeners", *AeroMat*

- 2012 Conference and Exposition, Charlotte, NC.
- Mulani, S.B., Slemp, W.C.H. and Kapania, R.K. (2013), "EBF3PanelOpt: An optimization framework for curvilinear blade-stiffened panels", *Thin-Wall. Struct.*, **63**, 13-26.
- Myers, R.H. and Montgomery, D.C. (1995), *Response surface methodology - process and product optimization using designed experiment*, New York: Wiley.
- Niu, M.C.Y. (2005), *Airframe stress analysis and sizing*, Technical Book Company.
- Palmer, K. and Tsui, K. (2001), "A minimum bias latin hypercube design", *IIE Trans*, **33**(9), 793-808.
- Queipo, N.V., Haftka, R.T., Shyy, W., Goel, T., Vidyathan, R. and Tucker, P.K. (2005), "Surrogate based analysis and optimization", *Prog. Aero. Sci.*, **41**(1), 1-28.
- Rosenbrock, H.H. (1960), "An automatic method for finding the greatest or least value of a function", *Comput. J.*, **3**(3), 173-184.
- Sacks, J., Schiller, S. and Welch, W. (1989), "Designs for computer experiments", *Technometrics*, **31**(1), 41-47.
- Sacks, J., Welch W., Mitchell T. and Wynn H. (1993), "Design and analysis of computer experiments", *Statistic. Sci.*, **4**, 409-423.
- Shen, Z.Q., Shi, J.B., Wang, K., Kong, F.S. and Bailey, J.S. (2004), "Neural network ensemble residual kriging application for spatial variability of soil properties", *Podesphere*, **14**(3), 289-296.
- Sunny, M.R. and Kapania, R.K. (2013), "Damage detection in a prestressed membrane using a wavelet-based neurofuzzy system", *AIAA J.*, **51**(11), 2558-2569.
- Toal, D.J. and Keane, A.J. (2011), "Efficient multipoint aerodynamic design optimization via cokriging", *J. Aircraft*, **48**(5), 1685-1695.
- Vavalle, A. and Qin, N. (2007), "Iterative response surface based optimization scheme for transonic airfoil design", *J. Aircraft*, **44**(2), 365-376.
- Venter, G., Haftka, R.T. and Starnes, J.H. (1998), "Construction of response surface approximations for design optimization", *AIAA J.*, **36**(12), 2242-2249.
- Wang, H., Li, E., Li, G.Y. and Zhong, Z.H. (2008), "Development of metamodeling based optimization system for high nonlinear engineering problems", *Adv. Eng. Softw.*, **39**(8), 629-645.
- Welch, S.M., Roe, J.L. and Dong, Z. (2003), "A genetic neural network model of flowering time control in *arabidopsis thaliana*", *Agronomy J.*, **95**(1), 71-81.
- Ye, K. (1998), "Orthogonal column latin hypercubes and their application in computer experiments", *J. Am. Statist. Assoc.*, **93**(444), 1430-1439.
- Zaabab, A.H., Zhang, Q-L. and Nakhla, M. (1995), "A neural modeling approach to circuit optimization and statistical design", *IEEE Transactions on Microwave Theory and Techniques*, **43**, 1349-1358.