Response Surface Simulation – Patchwork Application

Stephan Pannier[1], Michael Beer[2], Martin Liebscher[1], Wolfgang Graf[1]
[1]Institute for Structural Analysis, TU Dresden, Dresden, Germany
[2]Department of Civil Engineering, National University of Singapore, Singapore

Abstract:

The repeated performance of structural analysis, e.g., within a reliability based optimization, for real-world problems require a high computational effort. Due to the complexity of the real-world investigations, which are based on sophisticated Finite Element analyses of large nonlinear systems, a particularly efficient form of an approximation scheme is required. In this paper the improvement of the numerical efficiency utilizing neural network based approximation schemes are discussed. Therefore the theoretical basics of neural networks are introduced. A further improvement of the approximation quality is obtained with network based approximation schemes. Beside committee machines and network composites, a section-wise application of neural networks is presented. The developments are demonstrated by means of numerical examples to emphasize their features and by a practical, industrial-sized example to underline their applicability.

Keywords:

neural network, response surface, numerical efficiency, patchwork application

1 Introduction

A reliability based optimization of industrial-sized problems is a current engineering challenge. This presupposes the coupling of optimization methods and computational schemes for processing reliability analysis with algorithm for deterministic fundamental solutions. A plain application in its crude form is not feasible; an increase of the numerical efficiency of the simulation is required. This can be achieved by improving the numerical efficiency of the respective approaches. However, this is hindered by the tremendous computational cost for the deterministic fundamental solution. Despite considerable advances, a further increase of numerical efficiency is still a central demanding issue for generally enabling the reliability based optimization of large, industry-sized structures. For solution a response surface approximation seems to be most appropriate.

A response surface (RS) approximation with the aid of neural networks [2] is implemented in a reliability based optimization procedure to replace the numerically expensive FE analysis by a fast surrogate model. This possesses a high degree of generality and flexibility and is capable of eliminating noise from the raw data. Section 2 is devoted to elucidating the neural network concept for function approximation. A multi-layer perceptron network with feed-forward architecture is selected as the basic network form. For industrial-sized problems a simple form of a neural network may not be sufficient to capture the properties of the underlying computational model with an appropriate quality. For the solution two strategies are appropriate to remedy this insufficiency.

First, several neural networks are combined to construct approximation machines following different philosophies. Committee machines are formulated with a parallel structure of individual networks to eliminate noise by averaging the network outputs. Network composites are designed to sequentially reduce the approximation error by training only one network on the initial data and successively applying further networks on the error from the previous approximation step.

Second, a patchwork approximation is investigated to improve the approximation quality of local function features. A neural network is applied only localized to limited parts of the response surface. In dependence of the availability of experimental points a distinction between an integrated and a purely applied patchwork approximation may be done. Whereas for a purely applied patchwork approximation the supporting points of the respective patch-networks are predefined, in an integrated patchwork approximation the experimental points are adjusted to the requirements of the patch-networks. As a result the patchwork approximation is performed with appropriate supporting points, arranged in an advantageous manner. Thus the number of input-output pairs is reduced to a small amount.

In Section 4 the capabilities of analyses with a neural network based response surface approximation are demonstrated by means of a numerical example. Finally, advantages of patchwork approximations are demonstrated for an industrial-sized example.

2 Neural network based response surface approximation

2.1 Basic idea

The computational cost of a reliability based optimization of an industrial-sized problem is almost completely caused by the nonlinear FE analysis. Thus, the most effective measure to increase the numerical efficiency is to replace the costly deterministic computational model ($M : \underline{x} \rightarrow \underline{z}$) by a fast approximation solution based on a reasonable amount of initial deterministic computational results $\left[\underline{x}_T; \underline{z}_T\right]$. The reliability based optimization can then be performed with that surrogate model, which enables the utilization of an appropriate sample size for the simulation. The surrogate model is designed to describe a functional dependency between the structural parameters $\underline{x}$ and the structural responses $\underline{z}$ in the form of a response surface approximation, see Eq. (1). Due to a processing with the aid of the activation function, the structural quantities $\underline{x}, \underline{z}$ are transferred to input and output signals of the neural network $\underline{x}_{sig} = f_x(\underline{x}), \underline{z}_{sig} = f_z(\underline{z})$.

$$\underline{z}_{sig} = f_{RS}(\underline{x}_{sig}) \,. \tag{1}$$

For response surface approximation a variety of options exist; see [5, 7]. The suitability of the particular developments primarily depends on the properties of the computational model. Due to the very gen-

eral properties of the FE analysis, which can hardly be limited to convenient cases, a high degree of generality and flexibility of the approximation is demanded. In this context, a neural network provides a powerful basis for response surface approximation. This can extract information from initial deterministic computational results and can subsequently reproduce the structural response based on the extracted information only. According to the universal function approximation theorem, neural networks are capable of uniformly approximating any kind of nonlinear functions over a compact domain of definition to any degree of accuracy. That is, there is virtually no restriction for a response surface approximation with the aid of neural networks.

2.2 Network constitution

The construct of an artificial neural network is based on the design of the human brain. It is constituted by information-processing units – so called neurons, which are connected by synapses. A specific network architecture is built by combining these two main components according to a particular structure; see Fig. 1. Generally, it is reasonable to search for a neural network structure that is simple and clearly arranged while it yields reliable results. A feedforward architecture is selected as the basis, which has already been demonstrated as useful in solving various engineering problems of different kind.

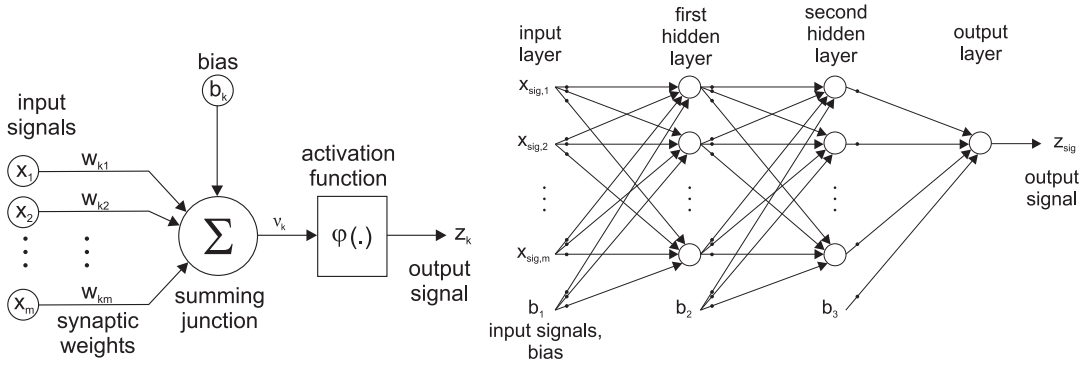
In the network, each synapse connects two neurons with each other. It enables the signal flow from one neuron to the next one. Each neuron contains a summing junction v, which lumps together the incoming signals x_{sig}, each one weighted by a specific synaptic weight w. The summation of input signals for neuron k, see Eq. (2), involves an additional external parameter called bias b,

$$v_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \,. \tag{2}$$

The weights w and the bias b allow the neuron to be adjusted to the particular problem. Their specific values are determined during the network training. This is performed on the basis of the error back-propagation algorithm; see [2]. The result v from the summing in Eq. (2) is the input argument for the subsequently called activation function $\varphi(.)$, which produces the output z_k of the neuron. Different types of activation functions are available such as threshold functions, piecewise linear functions and sigmoidal functions. In view of the training of the network, it is advantageous to implement differentiable activation functions. The most popular activation function is the logistic sigmoid function

$$\varphi(v_k) = \frac{1}{1+exp(-v_k)} \,, \tag{3}$$

which is selected for the further consideration.



(a) Construct of a neuron (b) Feedforward neural network

Figure 1: Constitution of a neural network

The neurons are arranged in a layered structure to form a multi-layer feedforward neural network, see Fig. 1. According to the feedforward philosophy, this permits a signal flow exclusively in forward direction

through the network. The layers are referred to as input layer, $L = 1$, hidden layers, $L = 2, ..., m-1$, and output layer, $L = m$. For a neural network with $L = 4$ the output signal $\underline{z}_{sig}$ is determined with

$$\begin{aligned} \underline{z}_{sig} &= \varphi\left(\sum_{j^{(3)}=1}^{J^{(3)}}\left[w_{j^{(4)},j^{(3)}}\cdot\varphi\left(v_{j^{(3)}}\right)\right]+b_{j^{(4)}}\right) \\ v_{j^{(3)}} &= \sum_{j^{(2)}=1}^{J^{(2)}}\left[w_{j^{(3)},j^{(2)}}\cdot\varphi\left(\sum_{j^{(1)}=1}^{J^{(1)}}\left[w_{j^{(2)},j^{(1)}}\cdot x_j\right]+b_{j^{(2)}}\right)\right]+b_{j^{(3)}}\ . \end{aligned} \tag{4}$$

In Eq. (4) the term $J^{(L)}$ denotes the number of neurons of the layer L, $w_{j^{(L)}j^{(L-1)}}$ is the synaptic weight of neuron $j^{(L-1)}$ between the layer $L-1$ and the neuron $j^{(L)}$ of the layer L for $L = 1, \ldots, m$. The number of input and output signals determines the number of the neurons of the input and the output layer, respectively. The number of hidden layers and the number of neurons in the hidden layers have to be specified in dependence on the particular problem. This represents an optimization problem, which may be solved with the objective of an optimum training behavior and optimum training result.

2.3 Training of neural networks

The adjustment of the synaptic weights w is commonly called the training of the neural network. These values are adjusted during the training phase of the neural network depending on the particular problem. The knowledge represented by a neural network after the training is stored in its synaptic weights. For the training procedure the back propagation algorithm is applied, which requires an existing data set. A training data set consists of training samples with a number of pairs of structural parameters (input vector $\underline{x}_T$) and corresponding structural responses (output vector $\underline{z}_T$). These samples have to be determined in advance by a repeated evaluation of the deterministic fundamental solution (mapping model M). In a first step an input vector $\underline{x}_{sig}$ is presented to the neural network and the associated neural network response $\underline{z}_{sig}$ is computed. The approximation error

$$e_k = z_{T,k} - z_{sig} \quad \forall \quad z_{T,k} \in \underline{z} \tag{5}$$

is determined. The errors e_k of all neurons m_k of the output layer are summarized with

$$\varepsilon = \frac{1}{2}\sum_{k=1}^{m_k} e_k^2 . \tag{6}$$

The synaptic weights are modified to minimize the objective function Eq. (6) with a gradient based approach whereas the adjustment is proportional to the gradient $\partial\varepsilon/\partial w$. A comprehensive overview of the back propagation algorithm is provided in [2].

3 Network-based approximation schemes

In the case of complicated dependencies between input and output data – as appears in the response surface approximation for real-world problems – a simple form of a neural network may not be sufficient to capture the properties of the underlying computational model with an appropriate quality. A further improvement of the approximation quality is required. Beyond an increase of the complexity of an individual neural network by adding more hidden layers and/or neurons, several neural networks may be combined to construct approximation machines according to different philosophies. Another viable concept targets on an approximation scheme oriented to local function features of the response surface.

3.1 Committee machines and network composites

Committee machines are formulated with a parallel structure of individual neural networks $f_{RS}^{(i)}$. Each network is separately applied to the same approximation problem with the same input-output pairs $\left[\underline{x}_T; \underline{z}_T\right]$. As the specification of a particular network and the network training is affected by random influences, a neural network solution for response surface approximation is generally not unique. That is, the $n-$fold

application of a neural network leads to n different response surface approximations. This effect appears, in particular, if the underlying data exhibit noise, for example, due to numerical sensitivities. It is thus reasonable to finally eliminate those noise effects by averaging the outputs $\underline{z}^{(i)} = f_{RS}^{(i)}(\underline{x})$ of the individual networks $f_{RS}^{(i)}$ of the committee machine (Eq. (7)).

$$\underline{z} = \frac{1}{n}\sum_{i=1}^{n}\underline{z}^{(i)} \tag{7}$$

Network composites [1], as another approximation scheme, are focused on the simultaneous representation of local and global function features. A first neural network $f_{RS}^{(1)}$ is trained to capture the global trend of the response surface reflected in the initial training data set $\left[\underline{x}_T; \underline{z}_T\right]$. Then, the remaining approximation error is computed $\underline{e} = \underline{z}_T - \underline{z}^{(1)}$. The error surface primarily reflects local function features, which have not been captured by the first network $f_{RS}^{(1)}$. A second network $f_{RS}^{(2)}$ is then trained on the error surface with the data set $\left[\underline{x}_T; \underline{z}_T - \underline{z}^{(1)}\right]$ and can so concentrate on the local function features only. The sum of the network outputs then yields an improved approximation (Eq. (8)). This scheme may be applied further to sequentially reduce the approximation error. Therefore the training data set is updated for each network $f_{RS}^{(i)}$ to $\left[\underline{x}_T; \underline{z}_T - \sum_{q=1}^{i-1}\underline{z}^{(q)}\right]$. The result $\underline{z}$ is obtained as the sum of output $\underline{z}^{(i)} = f_{RS}^{(i)}(\underline{x})$ of the individual networks $f_{RS}^{(i)}$, see Eq. (8).

$$\underline{z} = \sum_{i=1}^{n}\underline{z}^{(i)} \tag{8}$$

3.2 Patchwork approximation

The quality of a response surface approximation may be improved by approximating local function features. With a section-wise application of response surface approximations a patchwork-like result is obtained. Thereby, the design of a patch is not bounded to any requirements. It may be best constituted in dependence of the respective problem and the available set of input-output pairs. The approximation of the response surface within a patch may be performed with arbitrary approximation schemes. Generally, for real-world problems the behavior of local parts of the response surface are less complex than the function features of the complete response surface. Thus, the requirements onto an approximation scheme of local function features is less rigorous. To preserve a high degree of generality and flexibility an application of neural networks is reasonable. If the supporting points of a patch-network are determined in combination with the patchwork approximations, an beneficial union is established. This is referred to as an integrated patchwork approximation. However, for an predefined set of input-output pairs a pure patchwork approximation is also enabled.

In an *integrated patchwork approximation* the design of experiments is adjusted to the requirements of the patch-network. In dependence of the required approximation quality the number of input-output pairs is predefined. Furthermore, the constitution of the patch-size is adjustable to the respective problem. Generally, the user-defined specifications may increase the approximation quality on the one hand, but on the other hand it always augment the number of deterministic fundamental solutions. The size of a patch may be determined in dependence of the initial size of the input space $\underline{\mathbb{X}}^{(i)} \subset \mathbb{R}^n$, see Eq. (9). Thereby, $\underline{a}$ scales the size of the patch and n denotes the number of individual patch-networks.

$$\underline{p}^{(i)} = \underline{a} \cdot \Delta\underline{x} \mid i = 1, \ldots, n \tag{9}$$

To optimize the local approximation quality, the patches are specified online so that the point of interest for the approximation in each case coincides with the center of gravity of the patch. If a patch does not contain a minimum number of input-output points, the point of interest is evaluated with the aid of the computational model M for structural analysis. This leads to a moderate supplementation of the available set of input-output pairs in sparsely populated domains. Due to an integrated determination of the supporting points for a patch-network the number of input-output pairs is reduced to a minimal amount.

A *pure patchwork approximation* utilizes an available set of input-output pairs. In the neighborhood of the point of interest a predefined number of input-output pairs, which are closest to the point of interest, are determined with the aid of the euclidean distance. Thus, the patch-size depends on the distribution of the input-output pairs. The point of interest is obligatory not the center of gravity. A point of interest, situated in a sparsely populated domain, is analyzed without an adequate set of input-output pairs.

In view of the numerical efficiency of patchwork approximation the limitation of the network approximation to small patches and subsets of the initial data enables the application of small neural networks. Consequently, the training is associated with a low numerical cost. Otherwise, within a patchwork approximation the neural networks have to be trained several times. The repetition of the training is linked to an increase of the computational cost. This becomes important if a high number of points have to be evaluated, e.g., in reliability analysis utilizing Monte Carlo simulation. The numerical efficiency can be improved by re-using patch networks for further points of interest. Thereby it may be demanded, that the new point of interest $\underline{x}$ have to be situated, in relation to the center of gravity of a still trained patch-network $p^{(i)}$, within a predefined distance.

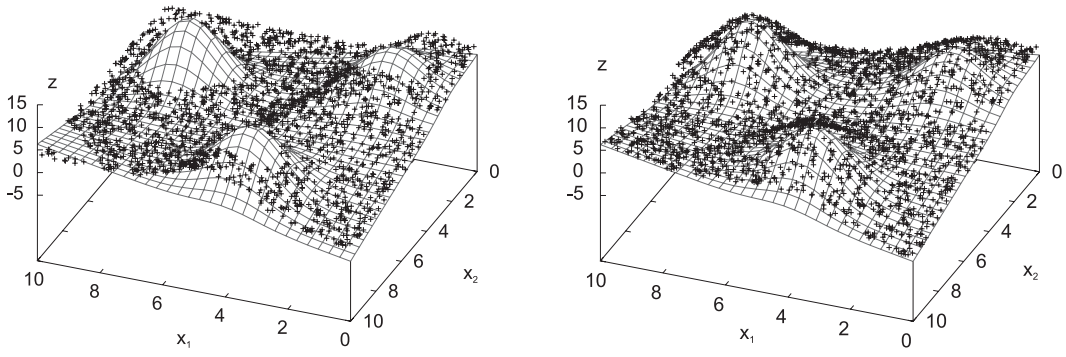
4 Examples

4.1 Benchmark study for patchwork approximation

The patchwork approximation is demonstrated by means of a numerical example. The aim of the investigation is to compare a single network with a patch-network. The constitution of the neural networks with $L = 3$ and $J^{(2)} = 2$ (Section 2.2) is kept simple for the purpose of visualization. The function $z(\underline{x})$ of interest (Fig. 2) can be expressed mathematically with

$$e(a, b, c, \underline{x}) = c \cdot \exp\left[-\frac{1}{2}(x_1 - a)^2 - \frac{1}{2}(x_2 - b)^2\right] \tag{10}$$

$$z(\underline{x}) = e(2, 2, 9, \underline{x}) + e(8, 3, 15, \underline{x}) + e(4, 8, 12, \underline{x}) + \frac{(x_1 - 5) \cdot (x_2 - 5)}{4} + rand(). \tag{11}$$



(a) Single neural network (b) Patchwork approximation

Figure 2: Single neural network vs. patchwork approximations of $z(\underline{x})$

For a set of 100 input-output pairs the respective response surfaces are approximated (Fig. 2). It is obvious, that a simple single neural network is hindered to reflect the characteristic parts of the performance function. The neural network provide only the potential to give an account of the global trend of the response surface. That can be elucidated under consideration of the network constitution. In contrast, a patchwork approximation with the same network constitution approximates the performance function with an appropriate quality. Even the local characteristics identified with the patchwork approach, which is of importance especially in reliability analysis. Assigning the results to real-world problems, it may be concluded, that the patchwork approach is capable of improving the approximation quality utilizing the same network constitution.

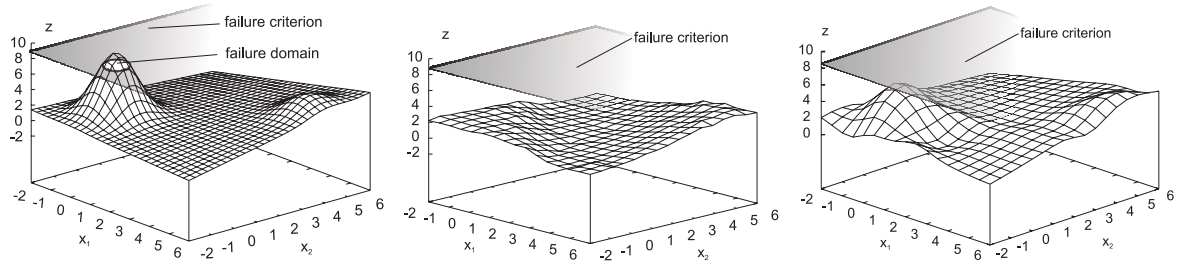
4.2 Benchmark study for an integrated patchwork approximation

The discussed approximation schemes are applied to a reliability analysis with a numerical example to verify their capabilities. For this purpose the computational model is substituted by the predefined function

$$z(\underline{x}) = 8 \cdot \exp\left(-\left(x_1^2 + x_2^2\right)\right) + 2 \cdot \exp\left(-\left((x_1-5)^2 + (x_2-4)^2\right)\right) + 1 + \frac{x_1 \cdot x_2}{10}. \tag{12}$$

Both input variables x_1 and x_2 are modeled as random variables X_1 and X_2, respectively, with a normal distribution, the expected value $\mu = 2.7$, and the standard deviation $\sigma = 1.0$. The failure criterion is specified with the limit state function $g(\underline{x}) = 8.7 - z(\underline{x})$. Failure occurs for $g(\underline{x}) \leq 0$; see Fig. 3. The reliability analysis is performed with the aid of a direct Monte Carlo simulation (MCS) to determine the failure probability. As a small failure probability is expected, the total number of evaluated points is predefined with $1,000,000$.

To reduce the number of deterministic solutions the Monte Carlo simulation is performed with the aid of neural network based response surface approximation. In this example three different approximation schemes are realized: single network, network composite, and patchwork-like approximation; see Section 3. As in the example above (Section 4.1) the constitution of the network is kept simple. The network-based approximation functions for $z(\underline{x})$ via a single network and a network composite cannot capture the behavior of $z(\underline{x})$ over the failure domain to a sufficient degree of accuracy; see Fig. 3. This would lead to a failure probability estimation of $\hat{P}_f = 0$.



(a) Performance function (b) Single network (c) Composite network

Figure 3: Neural network approximations of $z(\underline{x})$

On this account the patchwork approximation with neural networks is applied to improve the approximation quality over the failure domain. The determination of the input-output pairs for the supporting points of the respective patch-networks are integrated into the patchwork application. At least five points are required for the approximation of $z(\underline{x})$ within a patch. The size p_j of the patches, see Section 3.2, are defined in dependence on the maximum size of the available set of points $\underline{x}_T$ with actually known functional values $z(\underline{x}_T)$. To validate the results, the failure probability estimation is repeated 50 times, and the mean value $\hat{P}_f$ as well as the standard deviation $\sigma_{\hat{P}_f}$ is listed in Table 1.

In order to assess the quality of the result a reference solution is generated with direct Monte Carlo simulation utilizing the actual function $z(\underline{x})$ according to Eq. (12). It becomes obvious that the patchwork-like response surface approximation can achieve a computational effort that may enable a reliability analysis with a nonlinear FE simulation. This needs only $1,000$ actual evaluations of $z(\underline{x})$ to achieve a reasonable quality of the result.

Table 1: Mean value $\hat{P}_f$ and standard deviation $\sigma_{\hat{P}_f}$ of the estimated failure probabilities

Sampling method/ total sample size	failure probability $\hat{P}_f$	Standard deviation $\sigma_{\hat{P}_f}$
Direct MCS		
$N = 10^6$	$1.380 \cdot 10^{-5}$	$3.224 \cdot 10^{6}$
Direct MCS with patchwork RS approximation		
$N = 6,800,\ a = 0.03$	$1.167 \cdot 10^{-5}$	$3.228 \cdot 10^{-6}$
$N = 3,000,\ a = 0.05$	$1.067 \cdot 10^{-5}$	$3.740 \cdot 10^{-6}$
$N = 1,000,\ a = 0.10$	$1.253 \cdot 10^{-5}$	$4.657 \cdot 10^{-6}$

4.3 Reliability based optimization of a deep drawing process

In this example a reliability based optimization of a deep drawing process [6], depicted with the deep drawing device in Fig. 4, is performed. The aim of the investigation is to determine a setting of design quantities in view of a reliable manufacturing process.

Figure 4: Deep drawing device

Design quantities $\underline{d}$, which may be freely selected during the design process, are the mean value of the binder force d_1 and the mean values of the draw bead forces $d_2, \ldots, d_7$. The respective design ranges are predefined with intervals; $d_1 = [1400, 2400]$, $d_2 = d_3 = d_7 = [20, 200]$, $d_4 = [50, 120]$, $d_5 = [60, 120]$, $d_6 = [70, 130]$.

The remaining quantities are prescribed and non-alterable. Among all input quantities, sixteen are indicated to cause the uncertainty of the results and thus to influence the reliability predominantly. The quantification of the uncertain input parameters, elucidated in detail in [4], is accomplished with the generalized uncertainty model fuzzy randomness [3], see Table 2. Thereby, it is assumed that the randomness in respective input parameters is modeled with a normal distribution.

Table 2: Fuzzy random input quantities $\underline{\tilde{X}}$

fuzzy random quantity		normal distribution	
		mean value	standard deviation
yield strength f_y	$\tilde{X}_1$	0.14	$< 0.0067; 0.008; 0.01 >$
strength coefficient K	$\tilde{X}_2$	0.55	$< 0.0367; 0.044; 0.055 >$
hardening exponent n	$\tilde{X}_3$	$< 0.23; 0.275; 0.3 >$	0
friction coefficient μ	$\tilde{X}_4$	$< 0.05; 0.075; 0.1 >$	0
perturbation longitudinal p_1	$\tilde{X}_5$	$< -0.005; 0.0; 0.005 >$	0
perturbation lateral p_2	$\tilde{X}_6$	$< -0.005; 0.0; 0.005 >$	0
material parameter r_0	$\tilde{X}_7$	2.25	$< 0.0833; 0.1; 0.125 >$
material parameter r_{45}	$\tilde{X}_8$	1.7	$< 0.1; 0.12; 0.15 >$
material parameter r_{90}	$\tilde{X}_9$	2.85	$< 0.167; 0.14; 0.175 >$
draw bead force 1	$\tilde{X}_{10}$	d_2	$< 4; 5; 6 >$
draw bead force 2	$\tilde{X}_{11}$	d_3	$< 4; 5; 6 >$
draw bead force 3	$\tilde{X}_{12}$	d_4	$< 4; 5; 6 >$
draw bead force 4	$\tilde{X}_{13}$	d_5	$< 4; 5; 6 >$
draw bead force 5	$\tilde{X}_{14}$	d_6	$< 4; 5; 6 >$
draw bead force 6	$\tilde{X}_{15}$	d_7	$< 4; 5; 6 >$
binder force	$\tilde{X}_{16}$	d_1	$< 40; 50; 60 >$

As a result of the fuzzy stochastic analysis the fuzzy failure probability $\tilde{P}_f$ of the maximal shell thickness reduction z is evaluated. The design constraint for the maximal shell thickness reduction is predefined with $40\,\%$. As the aim of the investigation is to guarantee a reliable manufacturing process, the objective is to minimize the maximal possible failure probability $P_{f,\alpha=0,l}$ of the fuzzy failure probability $\tilde{P}_f$.

Facing the time consuming character of a single FE simulation an employment of RS approximation is inevitable. In an preliminary investigation the applicability of a single neural network approximation and

a patchwork approximation is tested. Therefore a sparsely set of $N = 234$ is determined with the aid of the computational model M. Utilizing such points a response surface approximation is established with both strategies. To validate the respective approximation quality, 100 input-output pairs are additionally evaluated. The obtained results are compared with the aspired ones by means of a mean square error. A single neural network deviates in the mean $9.2\,\%$ from the apsired result, while a patchwork approximation diverge 3.8%. Analyzing the results, it appears that the patch-networks approximate local function features more appropriate, see Table 3 (results are standardized). Considering the low number of input-output pairs, both approximation schemes provide useful results with an acceptable approximation quality. By reason of an better mean square error, the reliability based optimization is performed with the aid of a patchwork approximation.

Table 3: Comparison of selected results

results (standardized) computed with		
computational model M	single neural network	patchwork approximation
22.031	13.950	21.012
25.554	14.765	23.430
26.713	09.013	26.934
14.398	09.134	13.697
17.289	16.246	16.044
26.798	25.634	18.941
20.074	17.648	18.360
17.114	19.029	17.079
19.456	18.746	19.531
19.668	12.808	14.559

The result of the optimal design $\underline{d}_{opt}$ is depicted in Fig. 5 typified by the fuzzy cumulative distribution function and the fuzzy failure probability $\tilde{P}_f$. Further investigation have to validate the quality of the obtained optimal design utilizing an enlarged set of input-output pairs.

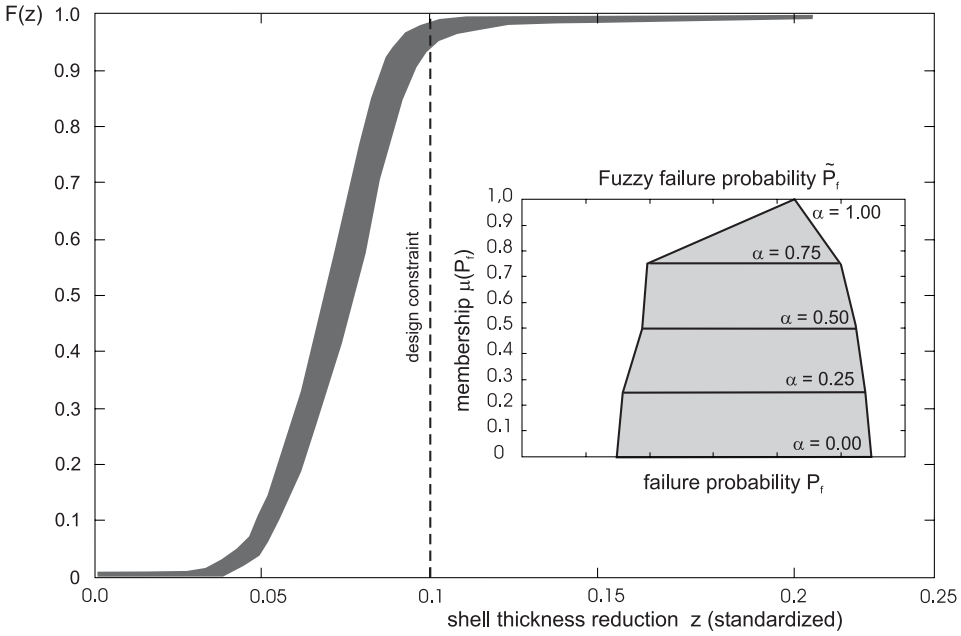


Figure 5: Fuzzy cumulative distribution function and fuzzy failure probability

5 Conclusions

The presented investigations demonstrate that neural network based response surface approximation generally provides a powerful tool to reduce the computational cost for reliability based optimization problems. With the aid of various approximation schemes the numerical efficiency is improved even in conjunction with time consuming and complex nonlinear computational models. The concept of response surface approximation schemes have been verified with respect to their capabilities in improving the numerical efficiency. The theoretical investigations have been underlined with an industry-sized example of a reliability based optimization of a deep drawing process. Utilizing expensive Finite Element analyses, patchwork approximations have been revealed as an appropriate solution. This emphasizes the relevance of network based approximation schemes for engineering practice.

Acknowledgment
The authors gratefully acknowledge the support of the German Research Foundation (DFG) and the courtesy of DaimlerChrysler.

References

[1] W. Beyer, M. Liebscher, M. Beer, and W. Graf. Neural network based response surface methods – a comparative study. In *Proceedings of the 5th German LS-DYNA Forum*, Ulm, 2006.

[2] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, New York, 1999.

[3] B. Möller and M. Beer. *Fuzzy Randomness - Uncertainty in Civil Engineering and Computational Mechanics*. Springer, Berlin, 2004.

[4] H. Müllerschön, D. Lorenz, W. Roux, M. Liebscher, S. Pannier, and K. Roll. Probabilistic analysis of uncertainties in the manufacturing process of metal forming. In *6th European LS-DYNA Users Conference*, 2007.

[5] R. H. Myers and D. C. Montgomery. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. Wiley, New York, 1995.

[6] S. Pannier, M. Liebscher, M. Beer, and W. Graf. Fuzzy stochastic simulation of deep drawing processes. In *EUROMECH Colloquium 482 – Efficient Methods for Robust Design and Optimization*, 2007. submitted.

[7] T. Simpson, J. Poplinski, P. N. Koch, and J. Allen. Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, 17(2):129–150, 2001.

