17th International Conference on the Application of Computer
Science and Mathematics in Architecture and Civil Engineering
K. Gürlebeck and C. Könke (eds.)
Weimar, Germany, 12–14 July 2006

ADAPTIVE RESPONSE SURFACE APPROACH USING ARTIFICIAL NEURAL NETWORKS AND MOVING LEAST SQUARES

T. Most and C. Bucher

Institute of Structural Mechanics
Bauhaus-University Weimar, Germany
E-mails:thomas.most@uni-weimar.de
christian.bucher@uni-weimar.de

Keywords: Reliability analysis, response surface method, Moving Least Squares, neural networks, adaptivity

Abstract. *In engineering science the modeling and numerical analysis of complex systems and relations plays an important role. In order to realize such an investigation, for example a stochastic analysis, in a reasonable computational time, approximation procedure have been developed. A very famous approach is the response surface method, where the relation between input and output quantities is represented for example by global polynomials or local interpolation schemes as Moving Least Squares (MLS). In recent years artificial neural networks (ANN) have been applied as well for such purposes.*

Recently an adaptive response surface approach for reliability analyses was proposed, which is very efficient concerning the number of expensive limit state function evaluations. Due to the applied simplex interpolation the procedure is limited to small dimensions. In this paper this approach is extended for larger dimensions using combined ANN and MLS response surfaces for evaluating the adaptation criterion.

1 INTRODUCTION

In engineering science the modeling and numerical analysis of complex systems and relations plays an important role. In order to realize such an investigation, for example a stochastic analysis, in a reasonable computational time, approximation procedure have been developed. A very popular approach is the response surface method, where the relation between input and output quantities is represented for example by global polynomials or local interpolation schemes as Moving Least Squares introduced by [1]. In recent years artificial neural networks (ANN) have been applied as well for such purposes.

Artificial neural networks have been designed to model the processes in the human brain numerically. These mathematical models are used today mainly for classification problems such as pattern recognition. In the recent years a large number of different neural network types have been developed, e.g. the multi-layer perceptron, the radial basis function network, networks with self-organizing maps and recurrent networks.

Neural networks (ANN) have been applied in several studies for stochastic analyses, e.g. in [2], [3], [4], [5], [6], [7] and [8]. In these studies the structural uncertainties in material, geometry and loading have been modeled by a set of random variables. A reliability analysis has been performed either approximating the structural response quantities with neural networks and generating ANN based samples or by reproducing the limit state function by an ANN approximation and decide for the sampling sets upon failure without additional limit state function evaluations. The main advantage of ANN approximation compared to classical RSM using quadratic polynomials is the applicability to higher dimensional problems, since classical RSM is limited to problems of lower dimension due to the more than linearly increasing number of coefficients.

Recently an adaptive response surface approach for reliability analyses was proposed in [9]. This method is very efficient concerning the number of expensive limit state function evaluations, but due to the applied simplex interpolation the procedure is limited to small dimensions. In this paper the original approach in [9] is extended for larger dimensions using combined ANN and MLS response surfaces for evaluating the adaptation criterion. Since only one set of limit state points is used the extended approach needs a much smaller number of limit state points to obtain similar results as with standard procedures, such as response surfaces on directional samples.

2 RELIABILITY ANALYSIS AND RESPONSE SURFACE METHOD

By assuming a random vector

$$\mathbf{X} = [X_1, X_2, \ldots, X_n] \tag{1}$$

of n mutually independent, standard normal random variables X_i and a limit state function $g(\mathbf{x})$ the probability of failure $P(F)$ reads

$$P(F) = \int_{g(\mathbf{x})\leq 0} \varphi_n(\mathbf{x}) d\mathbf{x} \tag{2}$$

where $\varphi_n(.)$ denotes the n-dimensional normal probability density. The limit state function divides the random variable space into a safe domain $S = \{\mathbf{x} : g(\mathbf{x}) > 0\}$ and a failure domain $F = \{\mathbf{x} : g(\mathbf{x}) \leq 0\}$.

The computational challenge in determining the integral of Eq. (2) lies in evaluating the limit state function $g(\mathbf{x})$, which for non-linear systems usually requires an incremental/iterative numerical approach. The basic idea in utilizing the response surface method is to replace the true limit state function $g(\mathbf{x})$ by an approximation $\eta(\mathbf{x})$, the so called response surface, whose function values can be computed more easily.

In this context, it is essential to realize that the limit state function $g(\mathbf{x})$ serves the sole purpose of defining the bounds of integration in Eq. (2). As such, it is quite important that the function $\eta(\mathbf{x})$ approximates this boundary sufficiently well, in particular in the region which contributes most to the failure probability $P(F)$. In [10] it has been pointed out that for reliability analysis it is most important to obtain support points for the response surface very close to or exactly at the limit state $g(\mathbf{x}) = 0$. For this purpose the random vector $\mathbf{X}$ in Eq. (2) is replaced by a random direction unit vector

$$\mathbf{A} = [A_1, A_2, \ldots, A_n] \tag{3}$$

and a random radius R which leads to

$$\mathbf{X} = \mathbf{A}R. \tag{4}$$

Then the integral from Eq. (2) reads

$$P(F) = \int_{r(\mathbf{a})=1} [1 - \chi_n^2(r^{*2}(\mathbf{a}))] f_{\mathbf{A}}(\mathbf{a}) ds(\mathbf{a}) \tag{5}$$

with

$$g(\mathbf{a}r^*(\mathbf{a})) = 0 \tag{6}$$

whereby $f_{\mathbf{A}}(.)$ is the probability density of the random directional unit vector $\mathbf{A}$, $\chi_n^2(.)$ is the χ^2 distribution function with n degrees of freedom and $ds(\mathbf{a})$ denotes integration over the unit hyper-sphere $r(\mathbf{a}) = 1$. For evaluating Eq. (5) only the distances $r^*(\mathbf{a})$ of the limit state surface from the origin in the direction of $\mathbf{a}$ have to be known.

The points required for the response surface approximation of the limit state $g(\mathbf{x}) = 0$ can be determined using Monte Carlo simulation based on directional sampling [11]. In Fig. 1 the limit state with discrete points is shown in principle. In this study the inverse radius $[r^*(\mathbf{a})]^{-1}$ is approximated with a response surface depending on the direction according to [5]. This leads to approximation values exact or close to zero for unbounded or unimportant regions, respectively. Fig. 2 shows the approximated function of the inverse radius for the unbounded two-dimensional limit state given in Fig. 1.

3 MOVING LEAST SQUARES APPROXIMATION (MLS)

An arbitrary function u is interpolated at a point $\mathbf{x}$ by a polynomial as

$$u^h(\mathbf{x}) = \begin{bmatrix} 1 & x & y & x^2 & xy & y^2 & \ldots \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \mathbf{p}^T(\mathbf{x})\mathbf{a} \tag{7}$$

where $\mathbf{p}(\mathbf{x})$ is the base vector and $\mathbf{a}$ contains the coefficients of the polynomial. These coefficients are constant in the interpolation domain and can be determined directly if the number

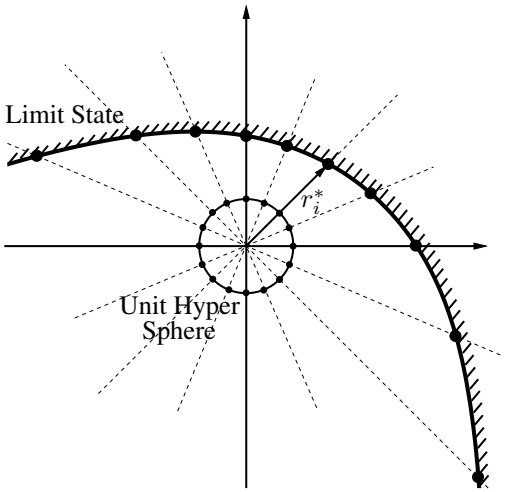

Figure 1: Limit state with discrete points

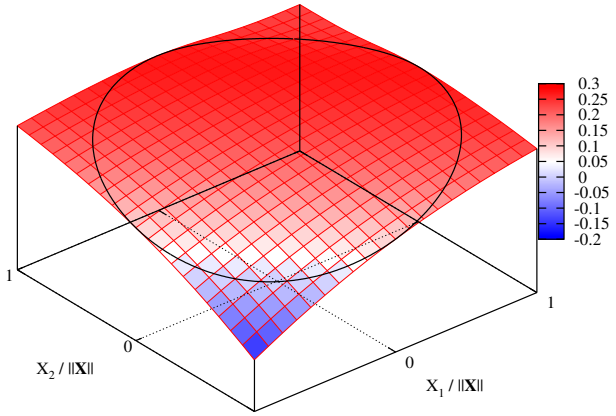

Figure 2: Representation of the inverse radius $[r^*(\mathbf{a})]^{-1}$

of supporting points m used for the interpolation is equivalent to the number of coefficients n. Then the coefficients are simply given as

$$\mathbf{a} = \mathbf{P}^{T^{-1}} \tilde{\mathbf{u}} \tag{8}$$

where $\tilde{\mathbf{u}}$ contains the function values at the supporting points

$$\tilde{\mathbf{u}} = \begin{bmatrix} \tilde{u}_1 & \dots & \tilde{u}_m \end{bmatrix}^T \tag{9}$$

and $\mathbf{P}$ consists of the values of the polynomial basis calculated at the supporting points

$$\mathbf{P} = \begin{bmatrix} P_1(\mathbf{x}_1) & P_1(\mathbf{x}_2) & \dots & P_1(\mathbf{x}_m) \\ P_2(\mathbf{x}_1) & P_2(\mathbf{x}_2) & \dots & P_2(\mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ P_n(\mathbf{x}_1) & P_n(\mathbf{x}_2) & \dots & P_n(\mathbf{x}_m) \end{bmatrix}. \tag{10}$$

Within the "Moving Least Squares" (MLS) interpolation method [1] the number of supporting points m exceeds the number of coefficients n, which leads to an overdetermined system of equations. This kind of optimization problem can be solved by using a least squares approach

$$\mathbf{P}\tilde{\mathbf{u}} = \mathbf{P}\mathbf{P}^T\mathbf{a}(\mathbf{x}) \tag{11}$$

with changing ("moving") coefficients $\mathbf{a}(\mathbf{x})$. In order to introduce a local character a distance depending weighting function $w = w(s)$ is defined, where s is the normalized distance between the interpolation point and the considered supporting point

$$s_i = \frac{\|\mathbf{x} - \mathbf{x}_i\|}{D} \tag{12}$$

and D is the influence radius, which is defined as a numerical parameter. Using the introduced weighting function, Eq. (11) is expanded to

$$\mathbf{B}(\mathbf{x})\tilde{\mathbf{u}} = \mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}), \tag{13}$$

where $\mathbf{A}(\mathbf{x})$ and $\mathbf{B}(\mathbf{x})$ are given as

$$\begin{aligned} \mathbf{A}(\mathbf{x}) &= \mathbf{P}\mathbf{W}(\mathbf{x})\mathbf{P}^T, \\ \mathbf{B}(\mathbf{x}) &= \mathbf{P}\mathbf{W}(\mathbf{x}), \end{aligned} \tag{14}$$

and the diagonal matrix $\mathbf{W}(\mathbf{x})$ can be determined as

$$\mathbf{W}(\mathbf{x}) = \begin{bmatrix} w(\mathbf{x}-\mathbf{x}_1) & 0 & \dots & 0 \\ 0 & w(\mathbf{x}-\mathbf{x}_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w(\mathbf{x}-\mathbf{x}_m) \end{bmatrix}. \tag{15}$$

The interpolated value of the function u at $\mathbf{x}$ can be obtained by introducing the MLS shape functions

$$u^h(\mathbf{x}) = \mathbf{\Phi}^{MLS}(\mathbf{x})\tilde{\mathbf{u}}, \quad \mathbf{\Phi}^{MLS}(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{A}(\mathbf{x})^{-1}\mathbf{B}(\mathbf{x}). \tag{16}$$

The classical MLS interpolation does not pass through the support point values due to the applied least squares approach. This implies that the interpolation condition is not fulfilled,

$$\Phi_i^{MLS}(\mathbf{x}_j) \neq \delta_{ij}. \tag{17}$$

A large number of different weighting functions for the Moving Least Squares interpolation can be found in the literature. In [12] a good overview is given. Due to the applied least square approach explained above, the obtained interpolation functions by using classical weighting types have a strong dependence on the size of the influence radius D, which leads to significant problems for irregular sets of support points. Because of these problems, in [13] the authors presented a new weighting function which enables the fulfillment of the MLS interpolation condition with very high accuracy without any additional numerical effort

$$\Phi_i^{MLS}(\mathbf{x}_j) \approx \delta_{ij}. \tag{18}$$

This can only be reached if Eq. (19) holds,

$$w_i(\mathbf{x}_j) \approx \delta_{ij}. \tag{19}$$

This so-called regularized weighting function reads for one support point i at an interpolation point $\mathbf{x}$

$$w_R(s_i) = \frac{\tilde{w}_R(s_i)}{\sum_{j=1}^{m} \tilde{w}_R(s_j)} \tag{20}$$

with

$$\tilde{w}_R(s) = \left(s^2 + \epsilon\right)^{-2}; \quad \epsilon \ll 1. \tag{21}$$

The regularization parameter ϵ has to be chosen small enough to fulfill Eq. (19) with high accuracy, but large enough to obtain a regular, differentiable function at $s = 0$ within the machine precision. In [13] it is recommended to use the value

$$\epsilon = 10^{-5}. \tag{22}$$

4 APPROXIMATION USING ARTIFICIAL NEURAL NETWORKS

The most widely used network type for approximation problems is the multi-layer perceptron which is also called feed-forward back-propagation network. This network type is used in this study and is shown in Fig. 3. The network consists of an input layer, several hidden layers and

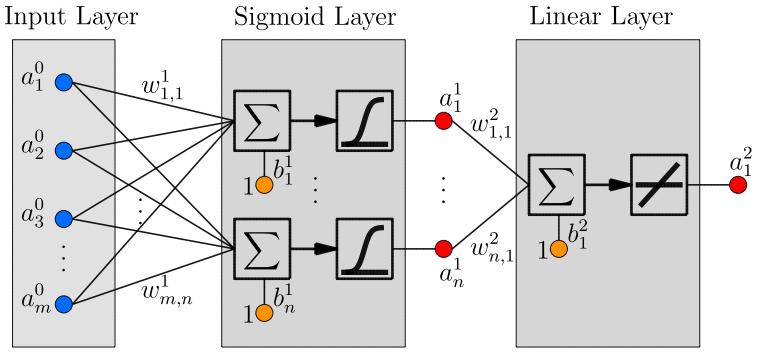

Figure 3: Neural network with feed-forward architecture and one hidden layer

an output layer and all nodes which are called neurons of one layer are connected with each neuron of the previous layer. At the neuron level a bias is added to the weighted sum of the inputs and the neuron transfer function is applied, which can be of linear and nonlinear type. The output of a single neuron reads

$$a_i^j = f_i^j(x) = f\left(\sum_{k=1}^{m} w_{k,i}^j a_k^{j-1} + b_i^j\right) \tag{23}$$

where m is the number of input impulses, i is the number of the current neuron in the layer j. $w_{k,i}^j$ is the synaptic weight factor for the connection of the neuron i, j with the neuron $k, j-1$. For the approximation of functions generally a combination of layers with sigmoid transfer functions $f(x) = 1/(1+e^{-x})$ and a linear output layer with $f(x) = x$ are used.

The training of a neural network has an important influence on the approximation quality. This training for a feed-forward multi-layered network is called back propagation where the network operation is executed reversely for the training sets and the calculated input values are compared to the given values. Depending on a given learning rate the calculated error is corrected and the same procedure is repeated until the training values are reproduced optimally by the network. If no convergence is achieved, generally the training is terminated after a given number of training loops called epochs. Different learning algorithm have been developed for this purpose which are described in [14]. In this work the Scaled Conjugate Gradient Algorithm proposed by [15] is used since it has a fast and stable convergence.

The design of the network architecture has an another important influence on the approximation. Depending on the number of available training samples the number of neurons in the hidden layers has to be chosen in that way, that the so-called over-fitting is avoided. This phenomenon occurs, if the number of hidden nodes is to large for the number of training samples. Then the network can converge easier and fits well for the training data but it can not generalize well for other data. In [14] it is mentioned that the number of training samples m should be larger than the number of adjustable parameters

$$(n+2)M + 1 < m \tag{24}$$

where n is the number of input values and M is the number of hidden neurons for a network with single hidden layer.

5 ADAPTIVE RESPONSE SURFACE APPROACH

In this section an adaptive response surface approach is presented which is based on the method proposed by [9]. In [9] two independent response surfaces each having a separate set of supports are used, whereby each response surface interpolates the conditional probabilities of failure using simplices. As error estimate used for the adaptation the maximum difference between both response surfaces concerning the conditional probabilities of failure was applied

$$e_{1,2} = max(e_i)_{1,2} = max\left[(y_i)_{1,2}|(p_i^*)_2 - (p_i^*)_1|\right]; \quad i = 1 \dots N_S, \tag{25}$$

where $(y_i)_{1,2}$ denotes a weighting of the current simplex, defined as the volume of the hypersphere segment spanned by the simplex and the origin and N_S is the number of simplices. The conditional probabilities of failure are computed in the direction of the centroid of the simplices as follows

$$p_i^* = [1 - \chi_n^2(r^{*2}(\mathbf{a}_i))]. \tag{26}$$

In the original approach the first response surface is adapted, where the maximum error e_1 was found and afterwards the second response surface is modified at e_2. This procedure is repeated until a certain accuracy is obtained.

Due to the limitation of the simplex interpolation for small dimensional problems, in this paper the adaptive approach proposed by [9] is extended for the application of neural networks and Moving Least Squares. For this extension several ways are discussed here. The first one is to use two response surfaces with two independent sets of supports points according to [9]. Then the maximum difference between both response surfaces can be obtained by using directional sampling. This results in a possible configuration as shown in Fig. 4 and may lead to wasted

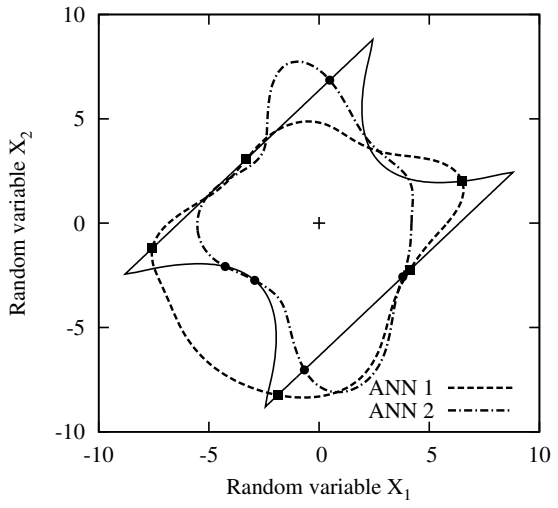

Figure 4: Adaptation using two independent ANN approximations with independent sets of supports

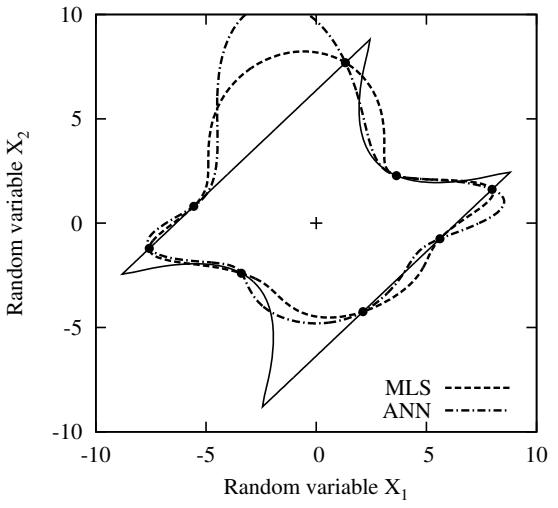

Figure 5: Adaptation using ANN and MLS approximation with only one set of supports

support points, if one response surface is to be adapted but already has a support point close to the location of the maximum difference and the other response surface does not have support

points in this region. Especially if a larger number of support points already exists a lot of points from both sets are almost identical. In this paper this problem is solved by using one set of points with two different approximation methods, namely neural networks and Moving Least Squares. This is shown for a certain state in Fig. 5. Investigations in the framework of this study have shown, that this procedure is more efficient concerning a smaller number of support points.

If problems are investigated with only one effective region contributing to the failure probability, the adaptation criterion using the maximum difference of the conditional probabilities of failure

$$e_p = max|(p_i^*)_2 - (p_i^*)_1|; \quad i = 1 \dots N_{DS}, \tag{27}$$

where i is the evaluated directional sample out of a set of N_{DS} realizations, always leads to a convergence of the method to the exact solution. For the analysis of systems having more than one effective region, this criterion may lead to a convergence to the wrong solution. This caused by the fact that one or more of these effective regions may not be detected by the algorithm, since their conditional failure probability obtained from the initial response surfaces is too small due to missing supports points in this region. This problem will be clarified in the second example. In this study the problem is solved using a given number of steps adapting the response surfaces at the maximum difference between the approximated radii

$$e_r = max|(r_i^*)_2 - (r_i^*)_1|; \quad i = 1 \dots N_{DS}. \tag{28}$$

6 NUMERICAL EXAMPLES

6.1 Simple frame structure

This is a simple analytical example representing the failure of a one-bay one-storey frame. The failure is assumed to be described by first-order rigid-plastic hinge theory. Due to the presence of horizontal and vertical loads (cf. Fig. 6) there are three relevant collapse mechanisms. The loads are assumed to be two Gaussian random variables. Using directional sampling with

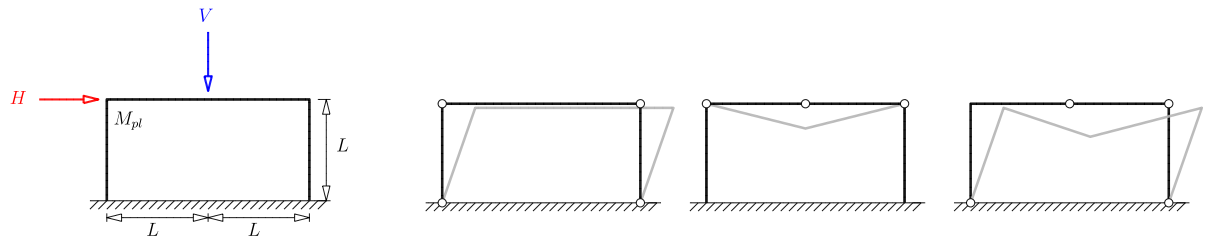

Figure 6: Simple frame with collapse mechanisms

1000 samples, the failure probability is determined as $\hat{P}_{ref}(F) = 1.208 \cdot 10^{-4}$. This value is used as a reference for comparison to the results based on the response surface approaches. In Fig. 7 the initial and the adapted response surfaces are shown for a single run of the algorithm. As adaptation criterion the maximum difference of the conditional probabilities of failure in Eq. 27 is used. The number of hidden neurons M is chosen automatically from the number of support points m rewriting Eq. 24 as follows

$$M = \frac{m-1}{n+2} = \frac{m-1}{4}, \tag{29}$$

whereby for rational values M only the next smaller integer value is taken and a minium value of one neuron is defined. The failure probability is determined using 1000 directional samples on the response surfaces and the maximum difference is obtained from these samples. The figure indicates, that the adaptation in the effective region works quite well. In Fig. 8 the convergence of the failure probability estimate is shown. For the presented single run the estimated value converges very fast compared to the ANN and MLS response surfaces on directional samples. The average error of 50 runs shows a similar behavior, but the ANN results show several peaks in the curves. These peaks are the result of an artificial oscillation of the network approximation in the unimportant region, which can be seen in Fig. 7. For only a few cases this oscillation goes close to the mean value and estimates an additional effective region. This is mainly caused by the simple dimensioning of the network using the maximum number of possible neurons and only one training phase for each network configuration. Nevertheless, the results of ANN and MLS approximation are very similar by using the adaptive and the directional samples as support points and the adaptive approach is very efficient compared to the response surfaces on directional samples and to plain directional sampling.

6.2 Complex limit state function

In this example a complex nonlinear limit state function given in [16] is investigated. This two-dimensional function is defined by a set of two linear and two quadratic functions:

$$\begin{aligned} g_1(\mathbf{x}) &= 0.1(x_1 - x_2)^2 - (x_1 + x_2)/\sqrt{2} + \beta_1 \\ g_2(\mathbf{x}) &= 0.1(x_1 - x_2)^2 + (x_1 + x_2)/\sqrt{2} + \beta_2 \\ g_3(\mathbf{x}) &= x_1 - x_2 + \sqrt{2}\beta_3 \\ g_4(\mathbf{x}) &= -x_1 + x_2 + \sqrt{2}\beta_4 \end{aligned} \tag{30}$$

whereby X_1 and X_2 are independent Gaussian random variables with zero mean and unit standard error. The reliability indices are given in [16] as $\beta_1 = \beta_2 = 4.0$ and $\beta_3 = \beta_4 = 4.5$. This limit state function has four effective regions, which contribute to the failure probability. The reference value for the probability of failure is obtained as $\hat{P}_{ref}(F) = 4.541 \cdot 10^{-5}$ by using again 1000 directional samples.

Due to the complex shape of the limit state function a minimum number of five neurons is chosen for the ANN approximation in order to obtain a response surface passing through the support point values. This means that for less than 21 support points the network is over-fitted, which leads to strong oscillations between the points.

In Fig. 9a an adapted state using the probability criterion is shown. It can be observed, that only two effective regions have been detected by the algorithm. An adaptation using the radius criterion from Eq. 28 which is shown in Fig. 9b can represent the complex shape much better, but due to the regular distributed number of points this procedure would converge very slowly to the exact failure probability. For this reason a combined criterion is used, whereby the first ten steps are adapted with the radius criterion to detect the effective regions and the following steps use the probability criterion to get a fast convergence of the failure probability. In Fig. 9c an adapted state for this case is shown.

In Fig. 10 the convergence of a single run and of the average error of 50 simulations are shown. As in the previous example some peaks in the ANN curve can be observed, which are more significant for less than 21 support points due to the over-fitting, but the convergence behavior of the adapted simulation is again much better than using response surfaces on directional samples.

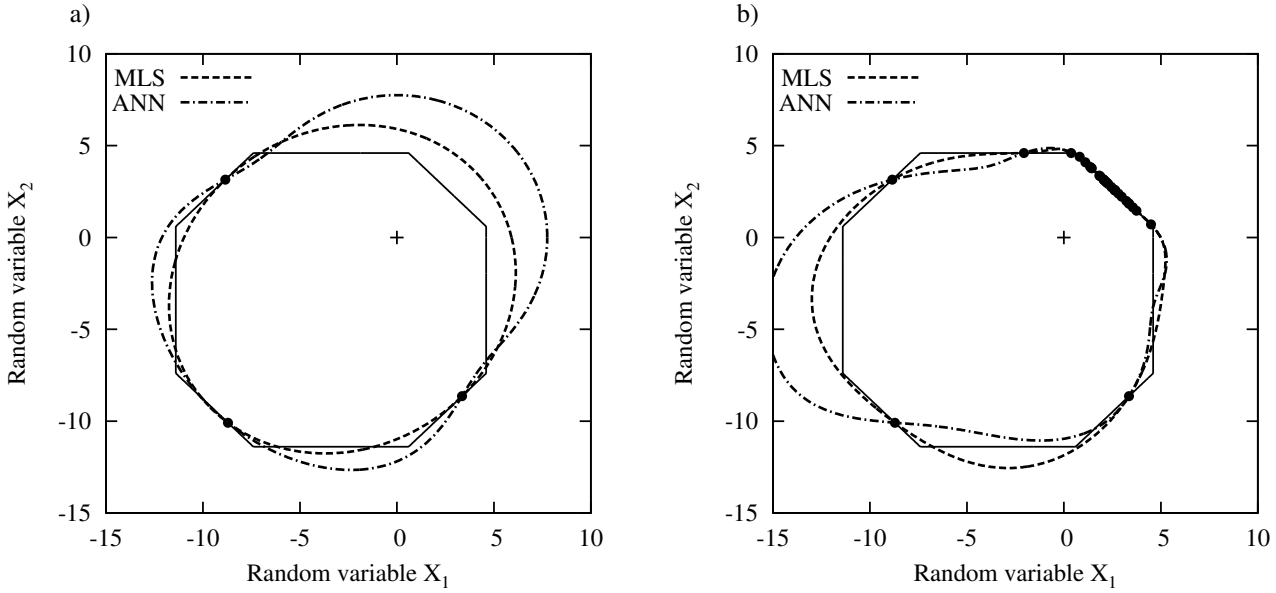

Figure 7: a) Initial state with 3 support points and b) adapted state with 33 supports points for the approximation of the limit state function of a simple frame structure

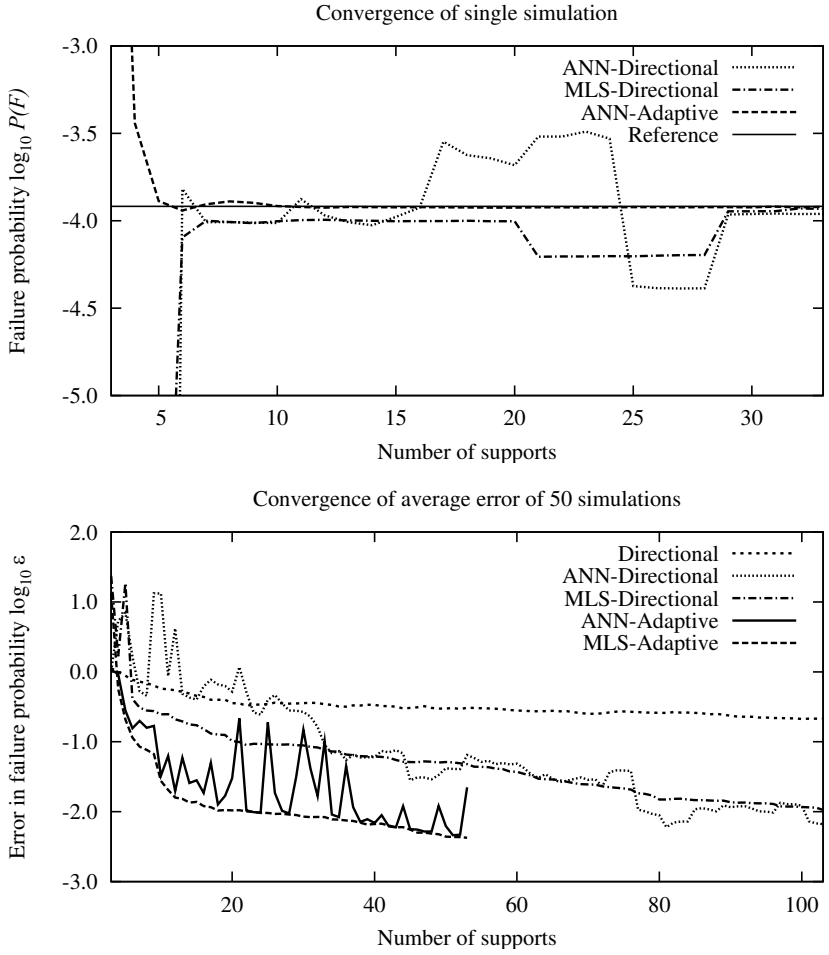

Figure 8: Convergence of the estimated failure probability for the investigated frame using directional sampling, ANN and MLS response surfaces on directional samples and ANN and MLS response surfaces on adaptive samples

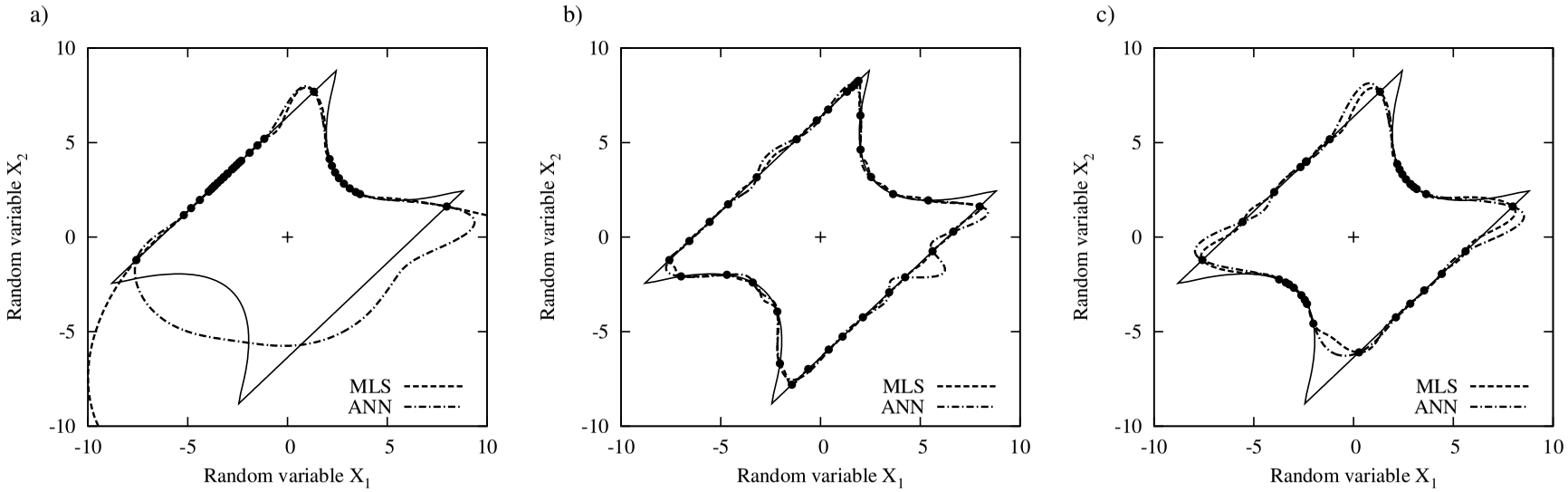

Figure 9: Adapted state with 33 points for the complex limit state function using a) the probability criterion, b) the radius criterion and c) the combination of both criteria

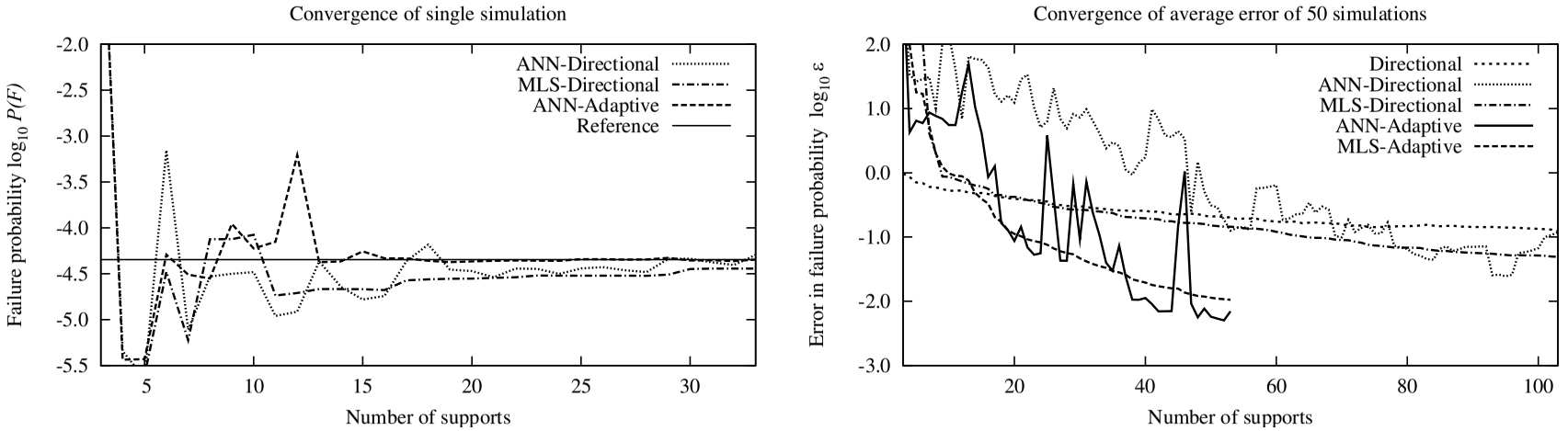

Figure 10: Convergence of the estimated failure probability for the complex limit state function

7 CONCLUSIONS

In this paper an efficient adaptive response surface approach is presented. This method uses one set of limit state points with two different approximation methods, Moving Least Squares and neural networks. As adaptation criterion a combination by using the maximum difference in the conditional probabilities of failure and the maximum difference in the approximated radii is applied successfully. Compared to response surfaces on directional samples or to plain directional sampling the failure probability can be estimated with a much smaller number of limit state points.

In the numerical results it is shown that the automatic dimensioning of the neural network using the maximum number of neurons where no over-fitting occurs works well, but still shows some problems, which can be seen in oscillations in regions with less number of support points. This problem could be solved in future e.g. if only the smallest number of neurons is used, for which the network training could be realized with a given error tolerance. Further improvements are necessary to pass through the support point values for a small number of points without allowing over-fitting. In addition the authors will analyze in future studies the applicability of Radial Basis Function networks for the presented purpose instead of the Multi-Layer Perceptrons.

ACKNOWLEDGEMENT

This research has been supported by the German Research Council (DFG) through Collaborative Research Center 524, which is gratefully acknowledged by the authors.

REFERENCES

[1] P. Lancaster and K. Salkauskas. Surface generated by moving least squares methods. *Mathematics of Computation*, 37:141–158, 1981.

[2] M. Papadrakakis, V. Papadopoulos, and N. D. Lagaros. Structural reliability analysis of elastic-plastic structures using neural networks and Monte Carlo Simulation. *Computer Methods in Applied Mechanics and Engineering*, 136:145–163, 1996.

[3] J. E. Hurtado and D. A. Alvarez. Neural-network-based reliability analysis: a comparative study. *Computer Methods in Applied Mechanics and Engineering*, 191:113–132, 2001.

[4] S. V. S. Cabral and L. S. Katafygiotis. Neural network based response surface method and adaptive importance sampling for reliability analysis of large structural systems. In R.B. Corotis, G.I. Schuëller, and Shinozuka M., editors, *Proc. 8th Intern. Conf. on Structural Safety and Reliability, Newport Beach, USA, June 17-22, 2001, CD-ROM*. Balkema, Rotterdam, 2001.

[5] J. Nie and B. R. Ellingwood. A new directional simulation method for system reliability. Part II: application of neural networks. *Probabilistic Engineering Mechanics*, 19:437–447, 2004.

[6] H. M. Gomes and A. M. Awruch. Comparison of response surface and neural network with other methods for structural reliability analysis. *Structural Safety*, 26:49–67, 2004.

[7] J. Deng, G. Desheng, L. Xibing, and Q. Y Zhong. Structural reliability analysis for implicit performance functions using artificial neural networks. *Structural Safety*, 27:25–48, 2005.

[8] L. Schueremans and D. Van Gemert. Benefit of splines and neural networks in simulation based structural reliability analysis. *Structural Safety*, 27:246–261, 2005.

[9] M. Macke and J. Riedel. Adaptive response surface method for structural reliability calculation. In R.B. Corotis et al., editors, *Proc. 8th Int. Conf. Structural Safety and Reliability (ICOSSAR), Newport Beach, USA, June 17-22, 2001*. Swets & Zeitlinger, Lisse, 2001.

[10] W. Ouypornprasert, C. Bucher, and G. I. Schuëller. On the application of conditional integration in structural reliability analysis. In A. H-S. Ang, M. Shinozuka, and G. I. Schuëller, editors, *Proc. 5th Int. Conf. on Structural Safety and Reliability, San Francisco*, pages 1683–1689, 1989.

[11] P. Bjerager. Probability integration by directional simulation. *Journal of Engineering Mechanics*, 114:1288–1302, 1988.

[12] T. P. Fries and H. G. Matthies. Classification and overview of meshfree methods. Informatikbericht 2003-3, Technical University Braunschweig, Brunswick, Germany, 2003.

[13] T. Most and C. Bucher. A Moving Least Squares weighting function for the Element-free Galerkin Method which almost fulfills essential boundary conditions. *Structural Engineering and Mechanics*, 21(3):315–332, 2005.

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

[15] M. Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6:525–533, 1993.

[16] S. Katsuki and D. M. Frangopol. Hyperspace division method for structural reliability. *Journal of Engineering Mechanics*, 120:2405–2427, 1994.