19th International Conference
ENGINEERING MECHANICS 2013
Svratka, Czech Republic, May 13 – 16, 2013

MULTIOBJECTIVE ADAPTIVE UPDATING OF SURROGATE MODELS

E. Myšáková, A. Pospíšilová, M. Lepš [1]

Summary: *A meta-model (or a surrogate) is a modern name for what was traditionally called a response surface. Our contribution presents an adaptive updating procedure to improve the quality of the surrogate. It is based on the minimax metric as an objective coming from the space-filling domain of the Design of Experiments. The second objective is aimed at an approximation quality in the region of interest. The final goal of the surrogate usage is to fit structural reliability problems.*

Keywords: ***Multiobjective optimization, Design of Experiments, miniMax, Maximin, Adaptive, Surrogate, Radial Basis Function Models, Limit State Function***

1. Introduction

Reliability-based design optimization (RBDO) is a research area that tries to optimize structures under assumption of uncertainties. Usually, an objective function (e.g. a structure weight, a maximal displacement etc.) is to be minimized with respect to constraints in which a probabilistic approach is included (Kang, 2005). It is hard or nearly impossible to create an analytical probabilistic approach on real structures thus some alternative method should be used. Our planned goal is to utilize a surrogate-based Monte Carlo approach (Dubourg, 2011) enhanced by an adaptive Design of (Computer) Experiments (DoE) (Sudret, 2007). This contribution presents the adaptive part shown on a simple example.

Mechanical problems are often modeled by a finite element method (FEM). Monte Carlo methods sample plenty of points in a design space to get the response of the FEM model. Here, the designs of experiments are used to choose which solutions (so-called points) will be solved to obtain maximum information out of the model with a minimal number of sampling points. Concurrently, sensitivities of the inputs can be obtained. Here, we are optimizing space-filling properties of the DoE by maximizing the minimal interpoint distance, i.e. the Maximin approach, by utilizing Simulated Annealing algorithm (Myšáková and Lepš, 2012).

As the model is enumerated many times, it is appropriate to use some meta-model (surrogate) that is easier to solve and give the approximation for the response on the original model. Radial Basis Function Networks (RBFN) (Kučerová et al., 2005) or Kriging (Sacks et al., 1989) are

[1] Bc. Eva Myšáková, Ing. Adéla Pospıšilová, doc. Ing. Matěj Lepš, Ph.D., Faculty of Civil Engineering, Czech Technical University in Prague, Thákurova 7, 166 29 Prague 6, tel. +420 224 355 326, e-mail leps@cml.fsv.cvut.cz

examples of very popular meta-models that can be applied in our work. For generating meta-models, an appropriate number of sampling points is needed (Jurecka, 2007). Moreover, to improve the quality of the surrogate, an adaptive updating procedure is proposed. It is based on the miniMax metric as an objective coming from the space-filling domain of the Design of Experiments. Overall, there are two criteria that have to be optimized. The first criterion (i.e. miniMax) is to maximize the nearest distance of the added point from already sampled points. The second criterion is to be as close as possible to the approximate Limit State Function, i.e. we are not concentrated on the whole domain, but only on the border between the failure and safe region. These two criteria lead to multi-objective optimization. In our work, a modified Nondominated Sorting Genetic Algorithm II (NSGA-II) is used.

A Nondominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al., 2000) is a method based on evolutionary principles. To create a new generation, only a mutation operator is used to support an exploration part of the algorithm (Lee et al., 2004). A selection from several consecutive Pareto fronts is followed by the computation of the crowding distances. Selected individuals with the greatest crowding distance are then used as a new generation. After a predetermined number of generations, only Pareto front is added as an adaptive update.

The paper is organized as follows. Next section describes the surrogate that will be used. Particularly, the Radial Basis Function Network will be presented as an interpolation of any black-box model. To solve multiobjective problems, it seems to be advantageous to use population-based genetic algorithms. As an example, the NSGA-II algorithm is described. Next, the multiobjective updating procedure will be introduced to consecutively improve the quality of the RBFN. Finally, to demonstrate the efficiency of the mentioned approach, the results of one well-known benchmark is presented.

2. Radial Basis Function Network

Artificial neural networks (NNs)[2] were developed to simulate the processes in a human brain but later on it was discovered that they can be effectively used for many problems like pattern recognition, different approximations and predictions, control of systems, etc, see e.g. works Bishop (1995), Haykin (1998) or Waszczyszyn and Ziemianski (2005). In this work, they will be used "only" as general approximation tools.

A neural network is created with several neurons (here called perceptrons) which are mutually interconnected. In this work we will deal with so called **feed-forward**, **layered** neural networks, i.e. neurons form sorted layers, each layer is connected with the previous and the next layer and the signal is processed directly from the inputs neurons to the output ones.

2.1. Approximation of a black-box function by RBFN

This type of a neural network is designed to simulate a black-box function $f(\mathbf{x})$ by its interpolation $F(\mathbf{x})$ given by the sum of basis functions multiplied by appropriate weights, see Figure 1. In other words,

$$f(\mathbf{x}) \approx F(\mathbf{x}) = \sum_{i=1}^{N} b_i(\mathbf{x}) w_i \,, \tag{1}$$

where $\mathbf{x}$ is a vector of unknowns, $b_i(\mathbf{x})$ is a basis function of the i-th neuron, w_i is a weight of the i-th neuron and N is the total number of neurons creating the net.

[2] Hereafter we will use only the term *neural network* instead of *artificial neural network* for the sake of simplicity.

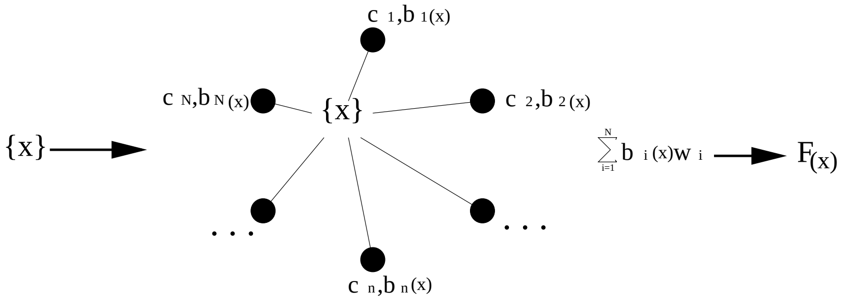

Figure 1: An approximation using RBFN

The basis function has the most often used "Gaussian" shape given by

$$b_i(\mathbf{x}) = e^{-\|\mathbf{x}-\mathbf{c}_i\|^2/r} , \tag{2}$$

where $\mathbf{c}_i$ is a vector of coordinates of the center for the i-th basis function and r is a norm. Normalization ensures that basis functions will produce similar values for different scales in multidimensional spaces. The selection of the norm r is not crucial and therefore the most common form is used:

$$r = \frac{d_{max}}{\sqrt[dim]{dimN}} , \tag{3}$$

where d_{max} is a maximal distance within the domain, dim is the number of dimensions and N is the number of neurons.

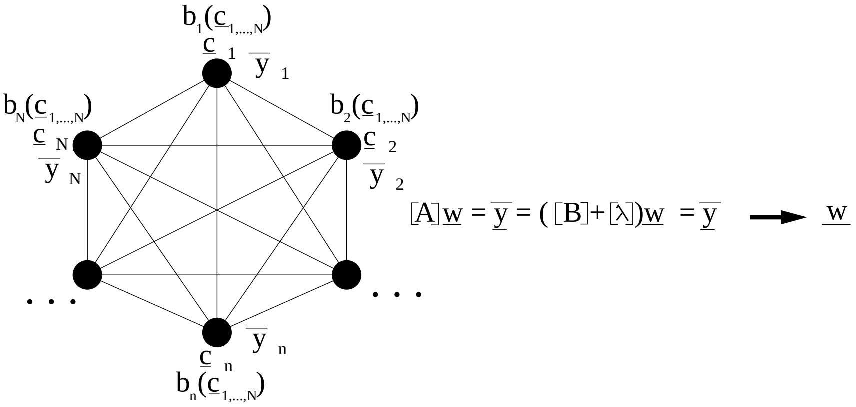

Figure 2: Training of a neural net

2.2. Training of a neural net

The weights of individual neurons can be obtained by the process of "training", see also Figure 2. Consider a set of training data

$$(\overline{\mathbf{x}_i}, \overline{y_i}) \ , \ i = 1, \ldots, p \ , \tag{4}$$

where $\overline{y_i}$ is a black-box function value in the $\overline{\mathbf{x}_i}$ point and p is a total number of records in the training set. For a usual RBFN the training set is identical with the basis functions centers, therefore we can write the training set also as

$$(\overline{\mathbf{c}_i}, \overline{y_i}) \ , \ i = 1, \ldots, N \ . \tag{5}$$

The weights w_i can be obtained from the equality between function values of a black-box function and its NN approximation in the function basis centers. Particularly,

$$f(\mathbf{c}_i) = F(\mathbf{c}_i) \tag{6}$$

and therefore

$$\min E = \min \sum_{i=1}^{N} [(\overline{y_i} - F(\mathbf{c}_i))^2 + \lambda_i w_i^2] \ , \tag{7}$$

where λ_i is used to regularize the system of equations 15 and it is set to $\lambda_i = 10^{-7}$ in our computations. Inserting Equation 1 into 7 we get

$$\min E = \min \sum_{i=1}^{N} [(\overline{y_i} - \sum_{j=1}^{N} b_j(\mathbf{c}_i) w_j)^2 + \lambda_i w_i^2] \ . \tag{8}$$

To satisfy Equation 7 resp. 8, the following identity has to be fulfilled

$$\frac{\partial E}{\partial w_i} = 2 \sum_{i=1}^{N} [(\overline{y_i} - \sum_{j=1}^{N} b_j(\mathbf{c}_i) w_j)(b_i(\mathbf{c}_i)) + \lambda_i w_i] = 0 \ . \tag{9}$$

Define a matrix $[\mathsf{A}]_N$

$$[\mathsf{A}]_N = [\mathsf{B}]_N + [\Lambda]_N \ , \tag{10}$$

where $[\mathsf{B}]_N$ is a basis function matrix

$$[\mathsf{B}]_N = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_N]^T \ , \tag{11}$$
$$\mathbf{b}_i = [b_1(\mathbf{c}_i), b_2(\mathbf{c}_i), \ldots, b_N(\mathbf{c}_i)] \tag{12}$$

and a $[\Lambda]_N$ is a diagonal matrix with all λ_i non-zero members. Next, we can write a vector of outputs

$$\mathbf{w} = [w_1, w_2, \ldots, w_N] \tag{13}$$

and a vector of black-box function values

$$\overline{\mathbf{y}} = [\overline{y_1}, \overline{y_2}, \ldots, \overline{y_N}] \ . \tag{14}$$

Then, Equation 9 becomes

$$[\mathsf{A}]_N \, \mathbf{w} = \overline{\mathbf{y}} \ . \tag{15}$$

By solving this system of *linear* equations we can obtain a vector of weights $\mathbf{w}$ for the given training set. Hence, from Equation 1, we can easily compute the approximation $F(\mathbf{x}) \approx f(\mathbf{x})$ for any vector $\mathbf{x}$.

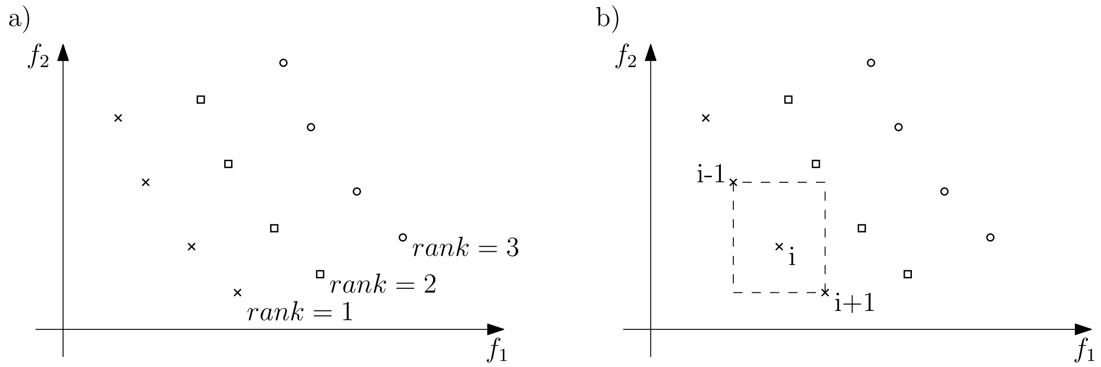

Figure 3: Calculation for NSGA-II of a) Nondominated rank, b) Crowding distance

3. Nondominated Sorting Genetic Algorithm II

A Nondominated Sorting Genetic Algorithm II (NSGA-II) was firstly published in Deb et al. (2000). It is an improved approach of NSGA where the main disadvantage was the high computational complexity of non-dominated sorting. In this version of the algorithm, there is no need to sort the population in every generation; also the original version has a lack of elitism and a need for specifying the sharing parameter for obtaining a wide variety of solutions. This algorithm deals with all of these disadvantages for obtaining a better solution much faster (Deb et al., 2002).

The first population is randomly created with a given distribution. In this work, the uniform distribution within a given range (lower and upper bounds of the parameters) is obtained by optimizing space-filling properties of the DoE by maximizing the minimal interpoint distance, i.e. the Maximin approach, by utilizing Simulated Annealing algorithm (Myšáková and Lepš, 2012). All individuals are sorted into each front (see Fig. 3a). The number of solutions n_p which dominate the solution p and a set of solutions S_p that the solution p dominates are calculated for each individual p. Afterwards, the $rank = 1$ is assigned for every individual with $n_p = 0$. These solutions are called nondominated and create the first front. This procedure is repeated until each individual has assigned its own rank. This means that all individuals are placed into the fronts. Solutions with $rank = 1$ are dominated by solutions with $rank = 0$, are nondominated to each other and dominates all solutions with $rank$ higher than 1. The same rule is applied to all individuals with higher ranks. To maintain diversity within the optimal fronts the crowding distance attribute is used for all nondominated solutions. This parameter designates how close the other solutions in the neighbourhood (see 3b) are. The bigger the crowding distance is, the better diversity in the solutions is.

The tournament selection for choosing individuals is used afterwards. The quality of the individual is identified with the very low rank and with the very high crowding distance. The more superior the individual is, the better chance to create an offspring with its properties is. Consequently, the crossover and mutation operators are used for creating the new offspring population.

The parent as well as the offspring population are sorted again and the $rank$ and the crowding distance are assigned to each individual. Only N individuals comes into the tournament selection. This procedure is stopped after the given number of generations is evaluated.

4. Multiobjective updating of RBFN

Here, we propose an adaptive updating scheme for consecutive improvement of the RBFN. It is based on an idea, that the new samples should be added in positions which are not covered with already enumerated points. Such a property is characterized by a miniMax metric coming from the space-filling domain of the Design of Experiments. Given a set of n points in a d-dimensional hypercube, the miniMax is the radius of the biggest sphere with its center inside the hypercube that does not contain any point of the set (van Dam, 2005; Pronzato and Müller, 2012). The second idea coming from Dubourg (2011) for reliability analysis purposes is to sample only the Limit State Function and not the whole design space. In other words, the sampling should concentrate only on a division line between two spaces, i.e. between the failure domain and safe region, respectively. This enables to minimize computational demands since only a smaller portion of the design domain is searched through.

These two criteria, i.e. the miniMax and the distance to the LSF can be used as two objectives for the selection which samples will be evaluated. Therefore, we have applied the already presented NSGA-II algorithm to find Pareto front of these two criteria. The algorithm is again starting from the optimized DoE with respect to the Maximin metric, see e.g. Figure 4 (right). Then, the NSGA-II algorithm is run and an approximation of Pareto front of the miniMax vs. distance to the LSF is found. We are starting to evaluate the members from Pareto set from the left, i.e. from the lowest distance to the LSF, see Figure 6 (top) for an illustration. Then we recompute the miniMax metrics for all remaining points, not to add points that are too close to each other. Therefore, only a portion of original Pareto front is truly evaluated. The procedure iterate until a prescribed number of iterations is attained or if some error reaches a prescribed level.

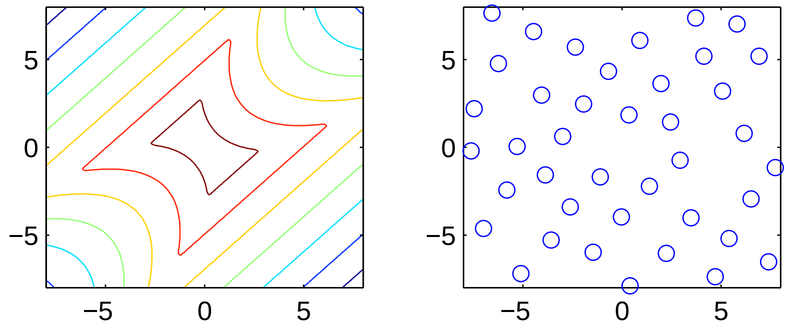

Figure 4: Contours of the example (left) and starting DoE (right). Note that the red contour is for LSF.

5. Example and results

The methodology will be shown on an academic example taken from (Waarts, 2000), see Figure 4 (left). The aim is to as closely as possible describe the zero level of the following function:

$$F(x) = min \begin{pmatrix} 3 + (x_1 - x_2)^2/10 - (x_1 + x_2)/\sqrt{2} \\ 3 + (x_1 - x_2)^2/10 + (x_1 + x_2)/\sqrt{2} \\ x_1 - x_2 + 7/\sqrt{2} \\ x_2 - x_1 + 7/\sqrt{2} \end{pmatrix}, \qquad x \in [-8; +8]^2 \quad . \tag{16}$$

This function can be understood as a safety margin and its zero level, i.e. $F(x) = 0$, as a Limit State Function. The progress is depicted in Figure 6. Five iterations drawn in columns are shown. The top line presents individual Pareto fronts, showing evaluated solutions in red points. The blue points then represents points of found Pareto front that are too close to already evaluated ones. In the middle line the evaluation of the contours are presented. Note that already after these five iterations, the contour of the zero line is starting to be similar to its goal depicted in Figure 4 (left). The last line then documents the adaptive sampling along the LSF.

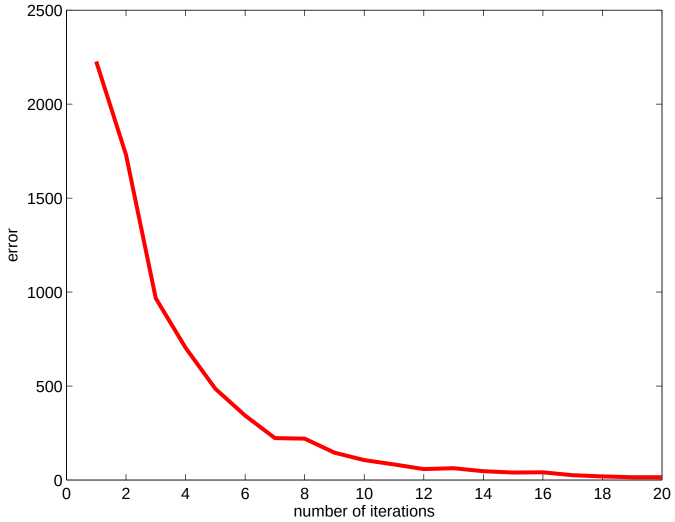

Figure 5: The development of an error during the iterations of the proposed procedure.

To evaluate the difference between the original and the approximate model, i.e. between Equation 16 and the RBFN, an error function has to be defined. Since we are interested in the quality of the description of the LSF, i.e. in the division of the design domain into a positive and a negative part, our error function counts a number of times the signs of these two models differ. Particularly, the error is evaluated on a regular 1000×1000 grid over a design domain and the differences are summed up. The evolution of this error for the studied benchmark is shown in Figure 5.

6. Conclusions

The presented contribution shows a new sampling-based surrogate that is adaptively updated during several iterations. The idea of the updating is based on the multiobjective nature of the space-filling properties of DoEs. Here, two objectives – the miniMax and the distance to some interesting region – are searched through with the evolutionary multiobjective algorithm. Moreover, only a portion of found Pareto front is added to ensure good space-filling properties. Since the iterative process is used, the tracked errors are consecutively minimized.

The future application of the proposed procedure is to solve the Reliability Based Design Optimization (RBDO) problem consisting of minimization of the weight of the structure as the first objective and minimization of the probability of failure characterized by a reliability index as the second objective. The latter can be evaluated by Monte Carlo-based Sampling algorithms utilizing the presented surrogate model. And again, this problem is multiobjective and the presented NSGA-based algorithm can be used.

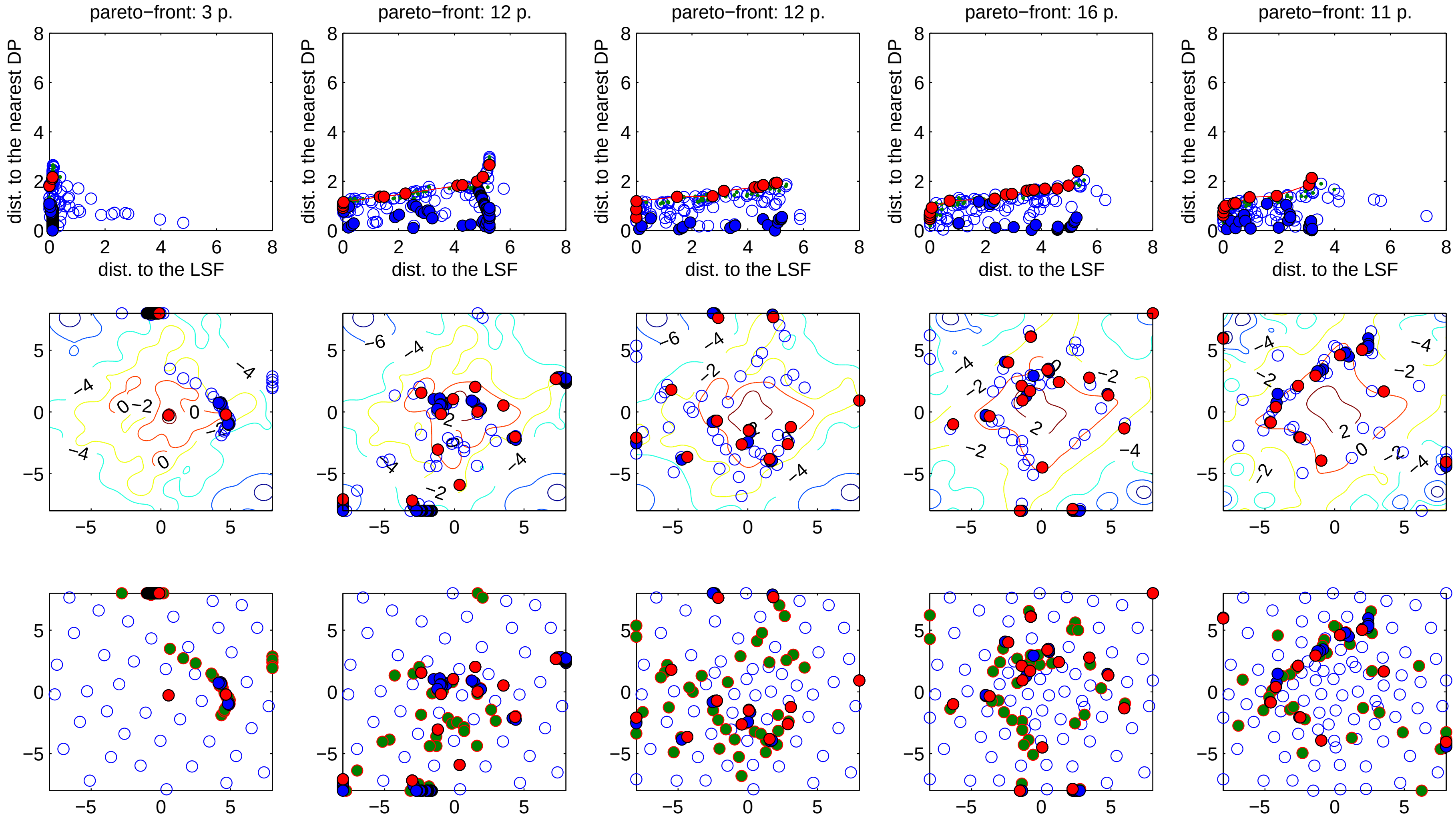

Figure 6: Pareto front (top), contours of the problem with DoEs (middle) and DoEs' points (bottom). Key: Red – added and computed solutions, Blue – points that were too close to other Pareto front points, Green – the remaining points of population and Blue empty points – the original DoE.

7. Acknowledgment

The authors gratefully acknowledge the financial support from the Grant Agency of the Czech Republic through the project GAČR P105/12/1146 and from the Grant Agency of the Czech Technical University in Prague, the grant SGS13/034/OHK1/1T/11.

8. References

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. pages 849–858. Springer.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans. Evol. Comp*, 6(2):182–197.

Dubourg, V. (2011). *Adaptive surrogate models for reliability analysis and reliability-based design optimization*. PhD thesis, Blaise Pascal University - Clermont II.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 2nd edition.

Jurecka, F. (2007). *Robust Design Optimization Based on Metamodeling Techniques*. PhD thesis, Technische Univesität München.

Kang, Z. (2005). *Robust Design Optimization of Structures under Uncertainties*. PhD thesis, Institut für Statik und Dynamik der Luft- ind Raumfahrkonstruktionen Univesität Stuttgart.

Kučerová, A., Lepš, M., and Skoček, J. (2005). Large Black-box Functions Optimization Using Radial Basis Function Networks. In *Proceedings of the Tenth International Conference on Civil, Structural and Environmental Engineering Computing*, Stirling. Civil-Comp Press Ltd.

Lee, J.-S., Cho, T.-S., Lee, J., Jang, M.-K., Jang, T.-K., Nam, D., and Park, C. H. (2004). A stochastic search approach for the multidimensional largest empty sphere problem. *Citeseer*.

Myšáková, E. and Lepš, M. (2012). Comparison of Maximin LHS Design Strategies. In *Proceedings of the 3rd Conference Nano and Macro Mechanics NMM 2012*, pages 163–170. CTU in Prague.

Pronzato, L. and Müller, W. G. (2012). Design of computer experiments: space filling and beyond. *Statistics and Computing*, 22(3):681–701.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4(4):409–435.

Sudret, B. (2007). *Uncertainty propagation and sensitivity analysis in mechanical models – Contributions to structural reliability and stochastic spectral methods*. Habilitation à diriger des recherches, Universitè Blaise Pascal, Clermont-Ferrand, France.

van Dam, E. R. (2005). Two-dimensional minimax latin hypercube designs. Discussion Paper 2005-105, Tilburg University, Center for Economic Research.

Waarts, P.-H. (2000). *Structural reliability using finite element methods: an appraisal of DARS: Directional Adaptive Response Surface Sampling*. PhD thesis, Technical University of Delft, The Netherlands.

Waszczyszyn, Z. and Ziemianski, L. (2005). *Parameter identification of materials and structures*, chapter Neural networks in the identification analysis of structural mechanics problems, pages 265–340. SpringerWienNewYork.