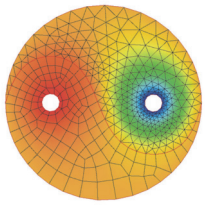
Publishing House
AKAPIT

COMPUTER METHODS IN MATERIALS SCIENCE

Informatyka w Technologii Materiałów

Vol. 14, 2014, No. 3



ANN-BASED METAMODELLING WITH CLUSTERING OF OUTPUT VALUES AS AN APPROACH TO ROBUST INVERSE ANALYSIS

GRZEGORZ GORECKI*, ŁUKASZ RAUCH, MACIEJ PIETRZYK, JAN KUSIAK

AGH University of Science and Technology,
Department of Applied Computer Science and Modelling, Al. Mickiewicza 30, Krakow, Poland
**Corresponding author: ggorecki@agh.edu.pl*

Abstract

Inverse analysis, which uses artificial neural networks as direct problem models, is gaining popularity. To improve robustness of the metamodel the idea of clustering of the networks based on their output was considered in the paper. This idea splits data by the output values, which in the inverse analysis are known as a part of the objective function. The individual networks are trained at small ranges of output values and they are supported by general networks trained on a wide range of data using special maps. The maps indicate which small network should be used to obtain more precise results. Possibility of using the general wide trained network for checking after the inverse analysis, whether the network for small range was correctly selected, is the main advantage of this method. The basic principles of this approach are described in the paper. Case study for identification of material flow stress model confirmed very good capabilities of this technique.

Key words: inverse analysis, artificial neural networks, metamodel, clustering by output values, material flow stress

1. INTRODUCTION

The inverse approach is a very useful tool for identification of various models, including material models used in simulations of materials processing. The inverse algorithm proposed by Szeliga et al. (2006) and validated for various materials by Szeliga and Pietrzyk (2012) gave very good results in various applications. Since in this algorithm finite element (FE) method was used to simulate experiments, long computing times were an important limitation for its wide applications, in particular in the industrial practice. Extensive research on possible decrease of the computing times led to a proposition of substituting the FE model by the metamodel. Quite good results were obtained when Artificial Neural Network (ANN) was applied as the metamodel of the compression test for cylindrical samples (Sztangret et al., 2012). However, reasonably simple Hensel and Spittel (1979) equation was used as a material model in that paper. Beyond this, compression of cylindrical samples (Uniaxial Compression – UC) is the simplest plastometric tests and FE simulation of this test does not require long computing times, therefore, a lot of data could be generated to train the network. Attempts to apply this technique to more complex material models or other tests (Ring Compression – RC; Plane Strain Compression – PSC) resulted in larger errors of training the networks.

Inverse analysis, which uses ANNs as direct problem models, is gaining popularity. This approach begins with obtaining the data and training the network, which is the metamodel of the experiment. Data obtained from the FE program was used by Sztangret et al. (2012) to train the metamodel. In

 ISSN 1641-8581

the inverse analysis a few control points were needed to determine the model coefficients properly. Therefore, in the discussed case the metamodel returns in control points output values (forces), which are related to selected specific displacements of the tool during the test. To improve robustness of the metamodel in the first approach, instead of using one network the problem was split into few networks. After this was done, the accuracy of the networks has improved noticeably. On the other hand, application of the networks to the inverse solution showed that metamodel based optimization may still create problems and the accuracy of the networks requires further improvement. It is due to the fact that the shape of multidimensional objective function is too complicated and small errors of the metamodel may cause stacking the solution in a local minimum. It inspired the Authors to review various methods of improvement of quality of neural networks and to search for a possibility of avoiding problems occurring in application of metamodelling to the inverse analysis.

Research in neural science is currently focused on looking for new kinds of neural network architectures or combinations of existing ones. Many proposed solutions are based on natural biological inspirations. Data preparation before training is a commonly known approach. Data sets can be divided by different algorithms from *k*-means family like in Lazarevic and Obradovic (2001) or neural networks. Kohonen networks are useful for this kind of clustering what was presented by Bezdek et al. (1992). This allowed focusing on networks working with smaller space in solving problems. Many investigations proved that the small domain problems are easier in modelling than the whole space, what was presented by Kuncheva (2005). In that case a specific network realizes only its own part of features. To obtain this effect groups called clusters should be made by using either one of clustering algorithms or a specific neural network. Next good approach is training multiple networks with different architectures at once, sometimes with aid of genetic algorithms, and then selecting the most effective network structures or data sets for specific case, as presented by Kermani et al. (1995). This approach requires predefined groups what could be problematic for some problems.

Regularization is another method to improve quality of neural networks as presented by MacKay (1992). Example approach for this method was implemented in Matlab software and was based on additional information supplied from weight values. Instead of typically used mean square error, a combination of MSE and mean square weights (MSW) was used. In that application regularization was automated, so training ANNs did not require manual specification of combination of MSE and MSW. Beyond this, common solution to avoid overfitting is early stopping. In this approach extra data set is used to monitor possible overfitting and to stop training process in the most appropriate moment.

Another approach to improve quality of neural networks involves multiple network training using the same training set. The output is the average of all networks' output. This solution frequently gives better results than the multiple network training with the best network selection.

Ahmad et al. (2010) proposed solution based on neural networks assembly and genetic algorithms. This assembly is used in both data processing and training, while the genetic algorithm is used to improve training process. The main idea of this method is introduction of separate neural networks, which return value with an acceptable error. The outputs of these networks are combined using weighted average method. Another approach is network retraining, in which existing trained ANNs are again trained using additional training set, which may be a subset of original training set what has presented Parvin et al. (2008).

The objective of this work was formulated having in mind presented review. The fact that in the inverse analysis the approximate location of the output of the metamodel is known was the main motivation. At the final stage of the optimization procedure this output has to be located in the vicinity of the measured output of the experiment. This observation inspired the Authors to apply clustering of the ANN with respect to its output.

2. INVERSE ANALYSIS WITH METAMODELLING

Inverse approach has proved to be a very useful tool for identification of various material models, see for example extensive research in the field of identification of rheological models of materials subjected to plastic deformation (Gelin & Ghouati, 1994; Gavrus et al., 1996; Khoddam et al., 1996; Forestier et al., 2002; Szeliga et al, 2006). The inverse algorithm proposed by Szeliga et al. (2006) and validated for various materials by Szeliga and Pietrzyk (2012) was used in the present work as a reference identification method. This algorithm

was described in details by Szeliga et al. (2006). Briefly, a mathematical model of an arbitrary process or physical phenomenon can be described by a set of equations:

$$\mathbf{d} = F(\mathbf{a}, \mathbf{p}), \qquad F : R^{k+l} \rightarrow R^{r} \tag{1}$$

where: $\mathbf{d} = \{d_1, ..., d_r\}$ - vector of output variables of the model (forces, shape of the sample after the tests), $\mathbf{a} = \{a_1, ..., a_k\}$ - vector of coefficients of the model, $\mathbf{p} = \{p_1, ..., p_l\}$ - vector of the known process parameters (temperatures, strain rates).

When vectors **p** and **a** are known, the solution of the problem (1) is called a direct solution. Inverse solution of the problem (1) called identification is defined as determination of the components of the vector **a** for known vector **d** (from the test) and vector **p**:

$$\mathbf{a} = F^{-1}(\mathbf{d}, \mathbf{p}), \qquad F^{-1} : R^{r+l} \rightarrow R^{k} \tag{2}$$

When the problem is liner, the inverse function can be usually found and the problem can be solved analytically. In the investigated problems of materials processing this relation is nonlinear and the problem is transformed into the optimization task. Thus, the objective of the inverse analysis is determination of the optimum components of vector **a** by searching for the minimum of the objective function, with respect to the components of this vector. The objective function is defined as a square root error between measured and calculated components of the vector **d**:

$$\Phi(\mathbf{a}, \mathbf{p}) = \sum_{i=1}^{n} \beta_i [\mathbf{d}_i^c(\mathbf{a}, \mathbf{p}_i) - \mathbf{d}_i^m]^2 \tag{3}$$

where: $\mathbf{d}_i^m$ - vector containing forces measured in the tests, it can also contain shape of the sample after the test (Szeliga et al., 2006), $\mathbf{d}_i^c$ - vector containing these parameters calculated by the model, β_i – weights of the points, (i = 1 .. n), n – number of measurements.

Thus, inverse analysis is composed of three steps: experiment, FE simulation of the experiment and optimization. Flow chart of this algorithm is shown in figure 1. It is well known that connection of the FE model with optimization techniques involves very long computing times. To avoid these extreme times two step inverse analysis was proposed by Szeliga et al. (2006). In the first step each test is investigated separately and stress-strain curve in a tabular form is obtained. Simplified corrections for variations of the temperature and strain rate during the test are introduced. These tabular data are approximated using selected flow stress function and coefficients obtained in the approximation are used as a starting point for the second step of the inverse analysis. This starting point is usually close to the minimum and the solution can be obtained in a reasonable time, in particular when non-gradient optimization methods are used.

The two-step inverse algorithm was applied to identification of material model on the basis of various experimental tests. It was shown that obtained model parameters were independent of the test which was used or of sample dimensions, see publications (Szeliga et al., 2002; Gawąd et al., 2005). This two-step algorithm has been successively used to solve problems of identification of material rheological model in many practical applications, see for example papers for steels for automotive industry (Hadasik et al., 2006) and steels used for manufacturing heavy crank shafts (Sztangret et al., 2011a). The weak points of the two-step solution are well known. All drawbacks of the non-gradient optimization methods could not be avoided. There was no proof whether the global minimum was found. There was no proof of the uniqueness of the solution either. Some improvement of the reliability of this approach was obtained when the sensitivity analysis was applied in parallel with the inverse solution and regularization of this solution was introduced (Szeliga, 2013). Nevertheless, due to long computing times application of the modern optimization methods based on observations of the nature (Genetic Algorithms, Evolutionary Algorithms, Immune Systems, Aunt Algorithm, etc.) was still not possible.

All these observations inspired Authors to search for a substitution of the FE direct problem model in the inverse solution by much faster model. Application of the metamodel based on the Artificial Neural Network gave promising results (Sztangret et al., 2012). This substitution is marked schematically in figure 1. The general idea of metamodelling relates to a postulation that metamodel approximates the model of considered process. Metamodel must correctly correspond to the model and the metamodel output value has to be evaluated with a radically lower computing time than using the original model. Thus, metamodelling is a process of construction of an approximation of the analysed model, on the basis of different techniques. In other words, the metamodel is a model of the model (Kusiak et al., 2014). The accuracy of the metamodel usually depends on the used metamodelling technique and on the number of the input data points. Usually, the higher number of points gives the better metamodel

accuracy. The metamodelling technique based on artificial neural network (ANN) was used in this paper. Examples of successful application of the ANN in optimization can be found in (Sztangret et al., 2012; Kusiak et al., 2014).

2. SOLVING PROBLEM

Currently many problems can be solved by using metamodelling. The goal of research for this paper was to identify properly coefficients in the material model using inverse solution with the metamodel for

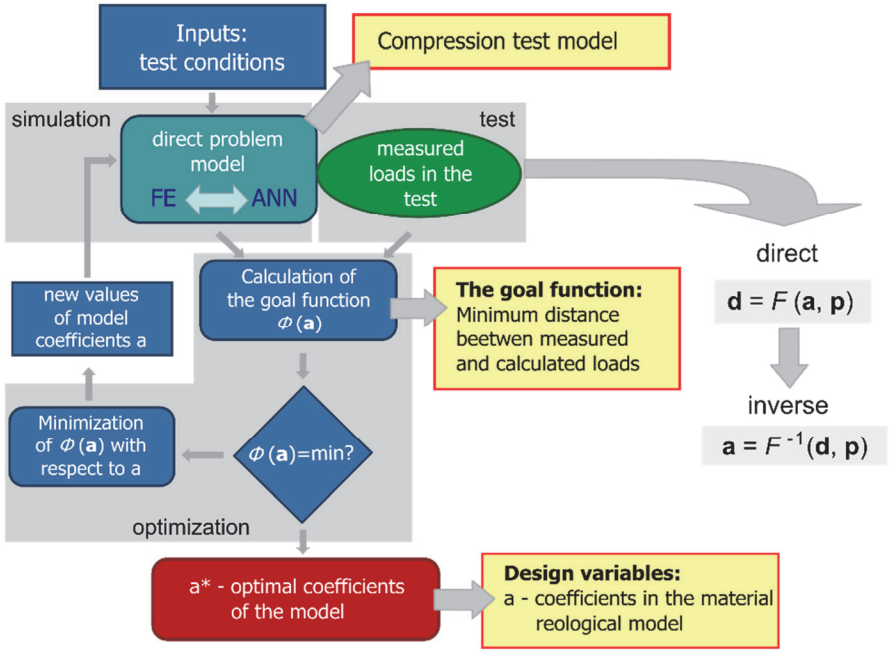


Fig. 1. *Flow chart of the inverse algorithm with the FE model and the ANN metamodel used alternatively as the direct problem models.*

ANN is an information processing system composed of a given number of artificial neurons (Tadeusiewicz et al., 2010 and 2014). Analysis of research described in (Sztangret et al., 2012; Kusiak et al., 2014) has shown, that ANN metamodel is efficient in the inverse analysis only for simpler material models, with a number of coefficients not exceeding five. An increase of the number of coefficients resulted in an increase of the error of the ANN, which further involved much larger increase of the error of the inverse analysis. Moreover, in the case of complex experiments, simulation of which is time consuming, generation of a satisfactory training data set is laborious. Therefore, the main objective of the present work was investigation of a possibility of improvement of the ANN accuracy by clustering of the ANN with respect to the value of the output (compression load). This objective was motivated and justified by the fact that, as it has already been mentioned in Introduction, at the final stage of the optimization in the inverse analysis the approximate location of the output of the ANN has to be close to the value obtained from measurements. The idea of clustering of the ANN for plastometric tests is presented in the following sections of the paper.

the compression tests. Many different equations can be used to describe behaviour of material during forming. Equation proposed by Gavrus et al. (1996) was considered:

$$\sigma_p = \sqrt{3}\left[W p_1 \varepsilon^{p_3} \exp\left(\frac{p_2}{RT}\right) + (1-W) p_5 \exp\left(\frac{p_6}{RT}\right)\right]\left(\sqrt{3}\dot{\varepsilon}\right)^{p_4} \qquad (4)$$

$$W = \exp(-p_7 \varepsilon)$$

where: σ_p – flow stress, ε – strain, $\dot{\varepsilon}$ – strain rate, T – temperature.

Equation (4) contains 7 coefficients (p_1-p_7), which have to be identified on the basis of experiment, which in this case was uniaxial compression test (UC). This problem has indirect nature so the inverse analysis was required to find the best fit. FE simulations of the UC test were performed to generate the data needed for training the metamodel of this test. To reproduce shape of the flow curve by the metamodel, few control points included in this metamodel were needed. During simulations ten forces were calculated for different displacements. Example of selection of displacements and forces obtained for these displacements are shown in figure

2. Values of forces for each selected displacement ($x_1 - x_{10}$) may move along the vertical lines, depending on the resistance to deformation of the material. All forces gathered together and compared with the measured forces allowed identification of the material flow stress model. Since dimensions of the sample are standardized, the metamodel input contained: friction coefficient, temperature, strain rate and 7 coefficients in the material flow stress model – equation (4).

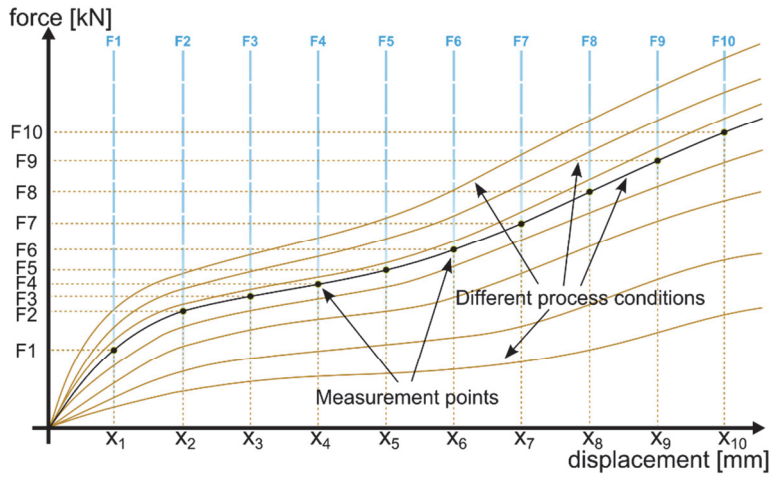


Fig. 2. Relation between forces and displacements for different conditions.

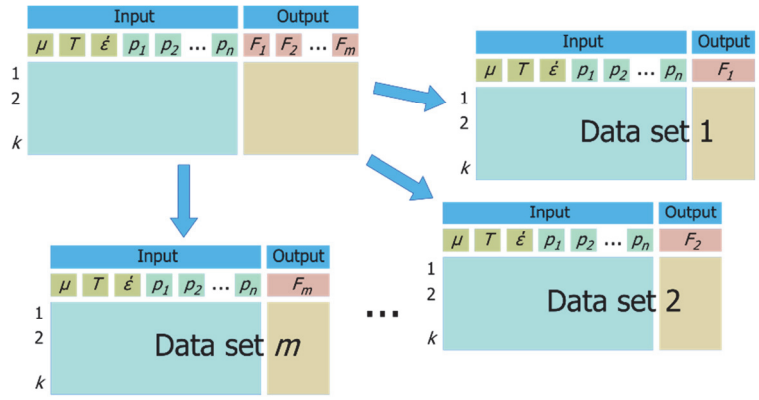


Fig. 3. Grid of the data separation process based on the output values of the model.

3. DESCRIPTION OF THE METHOD

The most important aspect of the designed method is dividing all the data into smaller ranges classified by the output values. The following steps are needed to make this process possible:

a) Creation of separated data sets

If the data are available from the process, which is described by relation $\underline{F}(\mathbf{x}) = \mathbf{y}$ ($R^n \rightarrow R^m$, where n represents numer of input variables and m represents number of output variables of solving problem), these need to be separated into smaller parts as follows: $F_1(\mathbf{x}) = \mathbf{y}_1$, $F_2(\mathbf{x}) = \mathbf{y}_2$, …, $F_m(\mathbf{x}) = \mathbf{y}_m$ ($R^n \rightarrow R_1$, $R^n \rightarrow R_2$, …, $R^n \rightarrow R_m$). Separation of rows by output values is shown in figure 3.

b) Quantization of the data sets

In this step each data set is split into small ranges of the output values using specific criteria:

$$i = \left\lfloor \frac{F - F_{min}}{d} \right\rfloor \tag{5}$$

where: i – index of the specific quant (equal to quantized value), F – value of force, for which the quant is searched, F_{min} – minimal value of force for the data set, d – quantization step size (constant for specific data set).

After this step the data should not be further separable in all following steps of the process, so they can be called as quants of the data. It is important to remember the order of consecutive quants, which will be needed for future association Force→Network. The quants are proposed only to make the specific association possible without increasing computational costs. In a very small scope case of values for quants, some quants can be empty. Introduction of the specific quantization criterion is required to make the quantization process possible.

For presented approach all quants have the same size what is presented in figure 5. Blue vertical lines separate quants. Red arrows describe direction of growth of forces. Smaller values are associated to the left side quants while larger values are associated to the right side along the arrow. This process must be performed separately for each output value (F_1, F_2, …, F_m). For different data sets it is possible to use other quantization criteria, which should be defined by the user of the system.

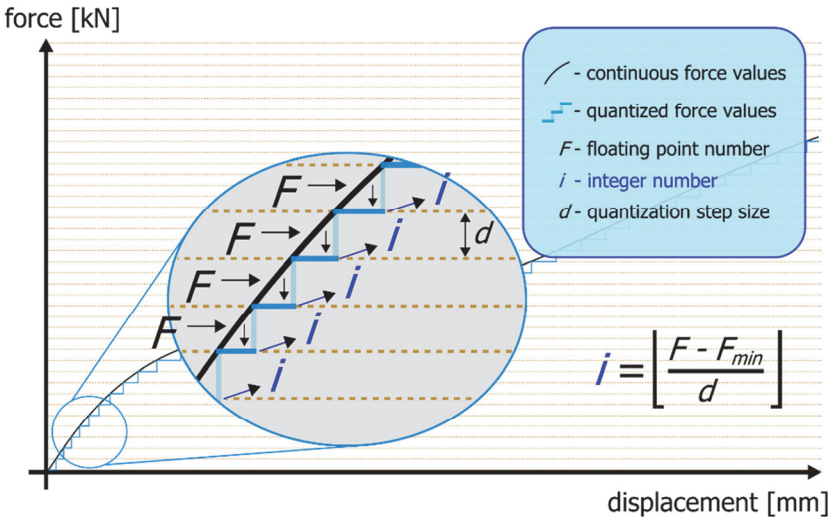


Fig. 4. *Visualization for force value quantization process according to equation (5). Dashed horizontal lines represent possible quantization levels for force values. The values are casted onto related integer value which indicates indexes used with map to find expected network. It will be described more precisely in steps c) and d).*

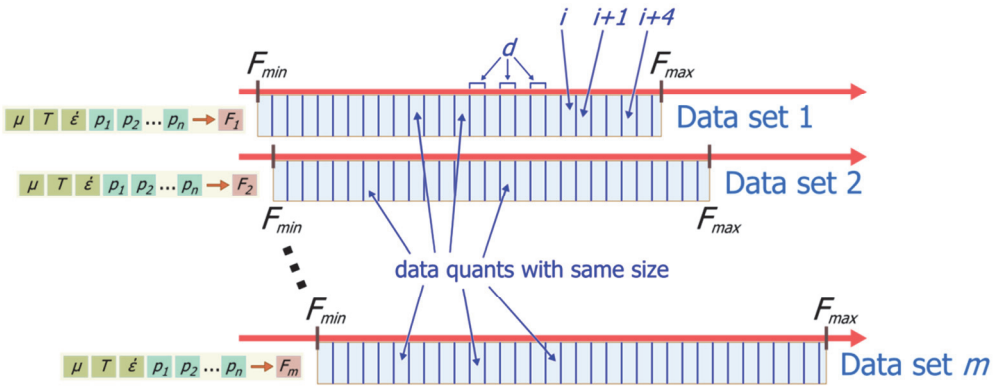


Fig. 5. *Data quantization on the basis of the output values.*

c) Grouping of data into the quants

The quants have to contain values with similar features according to the general clustering idea. The simplest approach is selecting small sizes of quants, what means that all quants will contain about one output value related to a specific row and many quants can have similar features. Next step is grouping process in which the values are assigned into specific groups. These groups are used in the next step for neural networks training. Quantitative criteria were used to create groups. It means that starting from the smallest values the quants were put sequentially into groups to obtain about 1200 rows in each group. The selected criteria were based on the following relations observed during numerical tests:

- Neural networks trained at small data ranges learned more effectively in the case analysed in this paper, therefore, it is advised to get values from neighbourhood.
- Majority of data rows in the training data file contains small values of outputs (forces), therefore, for larger forces larger sizes of quants were required to obtain large enough number of data,
- For small groups efficiency of training decreased.
- For too big groups error of the network increased what caused returning to a classical approach.

Selected data groups will be used in the future to retrain the networks. The general idea of the algorithm of the model is shown in figure 6. Example of the process of grouping of quants is presented in figure 7.

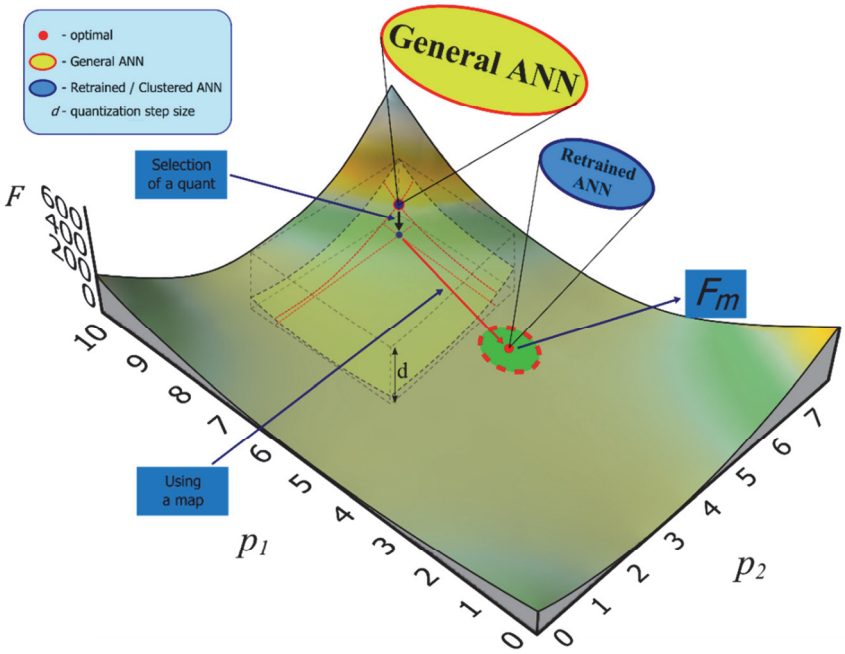


Fig. 6. *The general idea of the algorithm for identification based on clustered networks, F_m is the measured force in the objective function (1). Details about using map are presented in the next steps.*

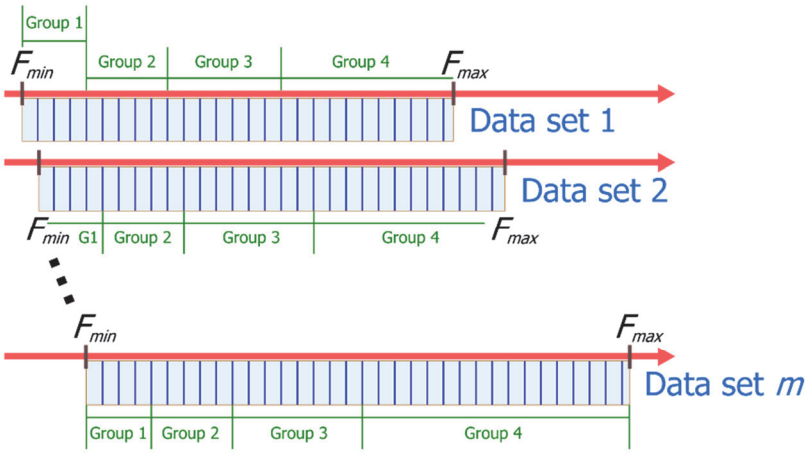


Fig. 7. *Grouping of data quants using quantitative criteria.*

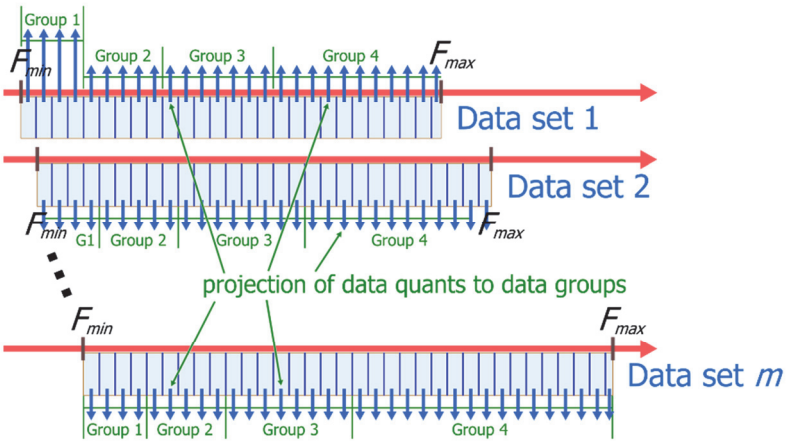


Fig. 8. *Creation of maps containing key indexes of quants and values of indexes of groups.*

d) Creation of maps

This step should be made in parallel with the grouping of the data into the quants (step c). Special maps are required to enable assessment, which network should be related with a specific force. This map allows determination which group includes the specific force and then which network was retrained to work with that force returning better results. Map is realized as simple one dimensional array. In the simplest way the values of array contain indexes of groups. The indexes of values have to represent indexes of data quants. This relation can be writ-

ten as $index\_of\_group = \mathbf{MAP}\left(\left\lfloor\frac{(F-F_{min})}{d}\right\rfloor\right)$ remembering that each data set can have its own map. Figure 8 shows the idea of mapping.

e) Network training

In the first step of training it is required to train each of the networks using data sets ($F_1(\mathbf{x}) = \mathbf{y}_1$, $F_2(\mathbf{x}) = \mathbf{y}_2$, …, $F_m(\mathbf{x}) = \mathbf{y}_m$ ($R^n \rightarrow R_1$, $R^n \rightarrow R_2$, …, $R^n \rightarrow R_m$)). Good practice is to retrain existing networks. Before retraining networks are copied in amount of the number of groups. In consequence, the copied networks could be retrained on the data from the appropriate group. This approach allowed decreasing of the time for the whole process and improvement of the training effectiveness.

Before training a few new quants from the closest neighbourhood were additionally added into the training data to guarantee better results around the boundaries between the ranges. Figure 9 shows the idea of training of networks for different data sources.

4. IDEA OF THE MODEL

Schema of a simple clustered metamodel is shown in figure 10. All networks inside the metamodel use the same set of input parameters. At the beginning of each step of the identification metamodel uses networks trained for all data sets. Results from the first level are quantized and forwarded to the **MAP**s, which return indexes for more precise networks. These networks further return results for the whole metamodel. It means that to obtain the results during optimization it is necessary to compute ANN twice. The time needed for these operations is very short in comparison to the FE calculations. Since it does not require any calculations, the **MAP** uses simple relations inside. The complexity of this approach is $O(1)$. For carried out tests the **MAP** is a vector. During optimization active precise networks are changed to focus at the end on the one, which is correct. Possibility of verification of results by comparing working range of the target network with expected forces is an important advantage of this method.

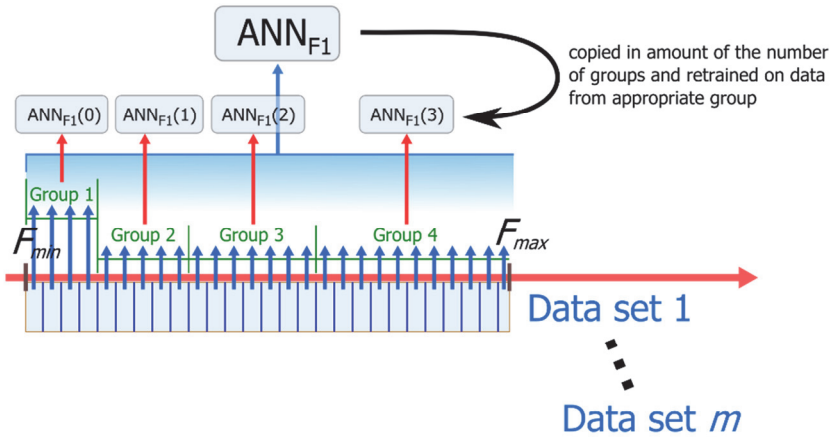


Fig. 9. *Relation between data groups and networks.*

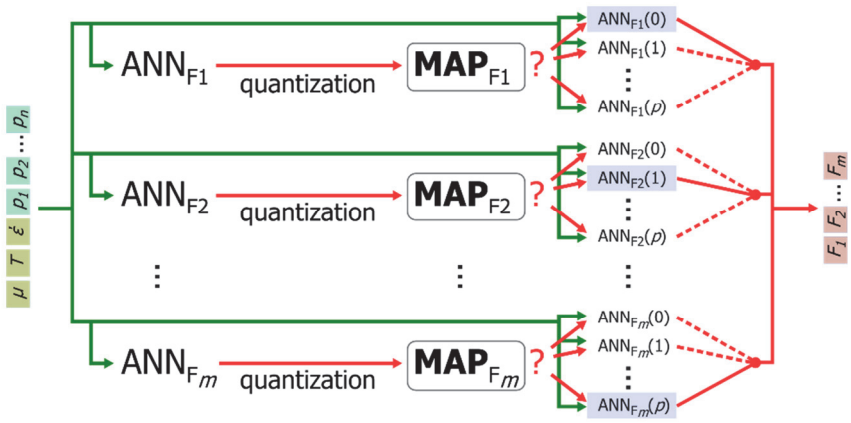


Fig. 10. *Schematic illustration of the clustered metamodel.*

5. BENCHMARK FUNCTION

Proposed method was tested at two-parameter Rastrigin function what was presented at equation (6). This function returns only one value, so first step in the described earlier method was avoided. To obtain data for parameter, values have been generated stochastically in range [-5, 5] for each parameter. To train Backpropagation Algorithm with inertia was used. Different structures of neural network was tested. As result the following network architecture was used: 2-10-10-1, where 10-10 are hidden layers with sigmoid functions and input(2-)/oputput(-1) layer with linear function. Trained chart was presented on figure below.

$$f(x_1, x_2) = 20 + x_1^2 - 10cos(2\pi x_1) + x_2^2 - 10cos(2\pi x_2) \quad (6)$$

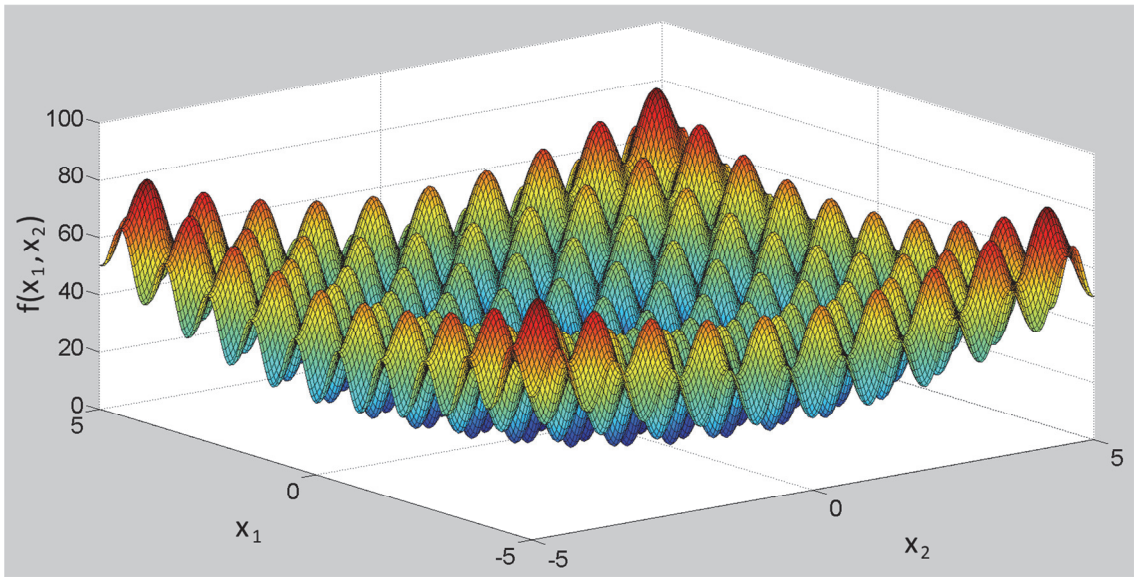


Fig. 11. *Benchamrk trained function.*

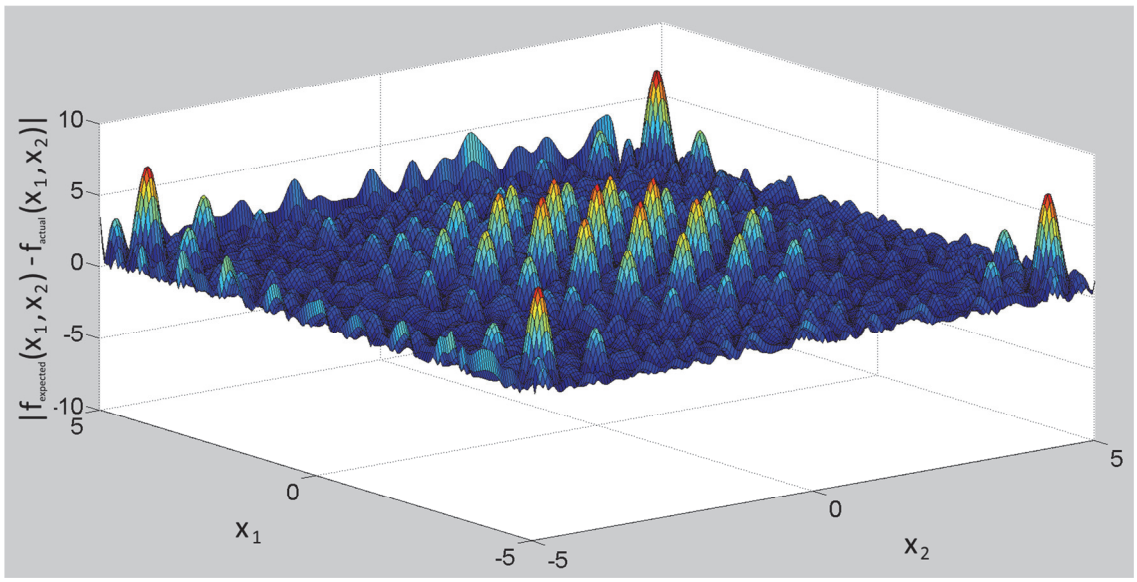


Fig. 12. *Absolute error for validation computed with trained network.*

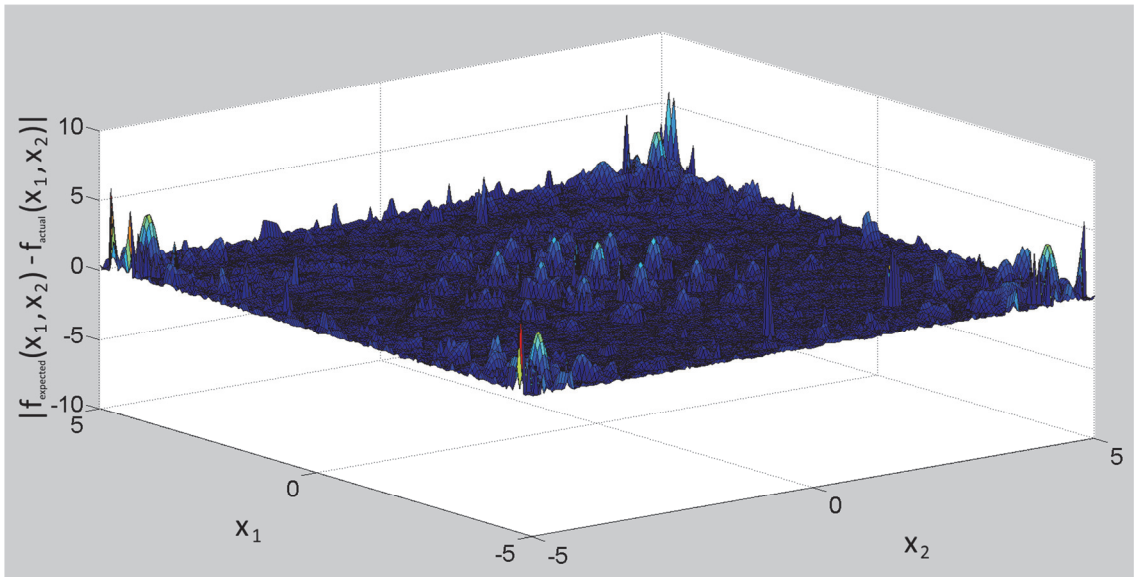


Fig. 13. *Absolute error for validation computed with trained clustered network.*

During training 11 thousand samples was used. Additionally for theoretical verification new set of values was computed using Rastrigin function (40 thousand samples). The absolute error graph for verification set for neural network was presented on figure bellow.

The goal of this example is focused to show how clustering can improve results. So to improve this metamodel neural network was clustered by output values according to presented in this paper approach. Training data was splitted into 11 data groups. Values at bounds of data groups were assigned to each neighbour groups to improve switching clustered networks. For this way prepared metamodel - containing clustered networks inside - error graph was prepared too, what was showed on below figure.

For presented approaches (figure 11 and 12) Root Mean Square Error (RMSE) was used too for clear comparision. Returned errors are equal to 1.1309 for first case and 0.5486 for clustered case – what looks sense looking for error graphs.

6. RESULTS

The first approach before clustering focused on classic inverse with metamodel approach. Ten artificial neural networks were trained. Those results were not satisfactory, what prompted to introduce clustering of the networks. Three networks were stored during training. The first network was a currently trained network. The second network was that with a minimal validation error (the network was validated after each of epochs) and the third network was that with the minimal error for training data (it was similar to the second one but it was updated after each of the epochs). Table 1 shows the summary of the obtained results.

The following error formula was used:

$$\Phi_{\mathrm{RMSE}} = \sqrt{\frac{1}{nf}\left(\sum_i^n \sum_j^f \left(1 - \frac{F_{ij}}{F\prime_{ij}}\right)^2\right)} \qquad (6)$$

where: n – number of tests, f – number of forces in one test, F_{ij} – current force value (from the metamodel), F'_{ij} – expected force value (from FE).

Table 1. *Comparison of results of classic approach and clustering, where Φ is defined by equation (7) and represents the error of the metamodel in comparison to the FE results.*

Classic metamodeling approach											
Test number	Φ_{RMSE}										
	total	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
1	0.0963	0.1645	0.0668	0.071	0.0792	0.0873	0.0853	0.0824	0.0951	0.0771	0.1153
2	0.0963	0.165	0.0667	0.0705	0.0788	0.0872	0.0853	0.0826	0.0953	0.0765	0.1155
3	0.0964	0.1653	0.0668	0.0697	0.0761	0.0878	0.0857	0.0812	0.0972	0.0771	0.1158
4	0.0964	0.1649	0.0637	0.0701	0.0785	0.0876	0.0863	0.0819	0.0965	0.0774	0.1157
5	0.0964	0.1635	0.0673	0.0713	0.0787	0.0877	0.0863	0.0817	0.0952	0.0779	0.1155
6	0.0963	0.165	0.0668	0.0702	0.0786	0.0872	0.0853	0.0825	0.0956	0.0764	0.1155
7	0.0964	0.1646	0.0658	0.0701	0.077	0.0873	0.0846	0.0835	0.0979	0.076	0.1161
8	0.0963	0.165	0.0668	0.0702	0.0786	0.0872	0.0853	0.0825	0.0956	0.0764	0.1155
9	0.0964	0.1662	0.0692	0.0697	0.0776	0.0869	0.0846	0.0823	0.0958	0.075	0.1152
10	0.0963	0.165	0.0668	0.0702	0.0786	0.0872	0.0853	0.0825	0.0956	0.0764	0.1155
Mean	0.0963	0.1649	0.0667	0.0703	0.0782	0.0873	0.0854	0.0823	0.096	0.0766	0.1156
Clustered metamodeling approach											
Test number	Φ_{RMSE}										
	total	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
1	0.0813	0.0729	0.1029	0.072	0.0724	0.057	0.0982	0.0993	0.0866	0.0975	0.1297
2	0.0809	0.1112	0.0776	0.0853	0.0706	0.0669	0.0949	0.0812	0.0825	0.0893	0.0767
3	0.0886	0.146	0.0888	0.0667	0.0662	0.0716	0.1004	0.0866	0.1089	0.0769	0.1462
4	0.0792	0.1261	0.0595	0.0744	0.0563	0.0587	0.0772	0.0912	0.1104	0.0681	0.1455
5	0.0833	0.1562	0.0529	0.0931	0.0701	0.0721	0.0756	0.0711	0.0893	0.0683	0.0908
6	0.0865	0.1437	0.0484	0.0885	0.0633	0.0787	0.0842	0.0685	0.1128	0.0961	0.1892
7	0.0846	0.1507	0.0705	0.0743	0.0705	0.0816	0.0908	0.0734	0.0871	0.0738	0.1181
8	0.0825	0.0903	0.0673	0.0806	0.081	0.0707	0.0856	0.1034	0.1184	0.0733	0.2246
9	0.0826	0.0899	0.0659	0.0793	0.0808	0.0712	0.0864	0.1041	0.1197	0.0728	0.2234
10	0.0835	0.1542	0.0536	0.0802	0.0692	0.075	0.0827	0.0659	0.0986	0.0733	0.0952
Mean	0.0833	0.1241	0.0687	0.0794	0.07	0.0704	0.0876	0.0845	0.1014	0.0789	0.1439

Comparison of force value errors for each has been presented below.

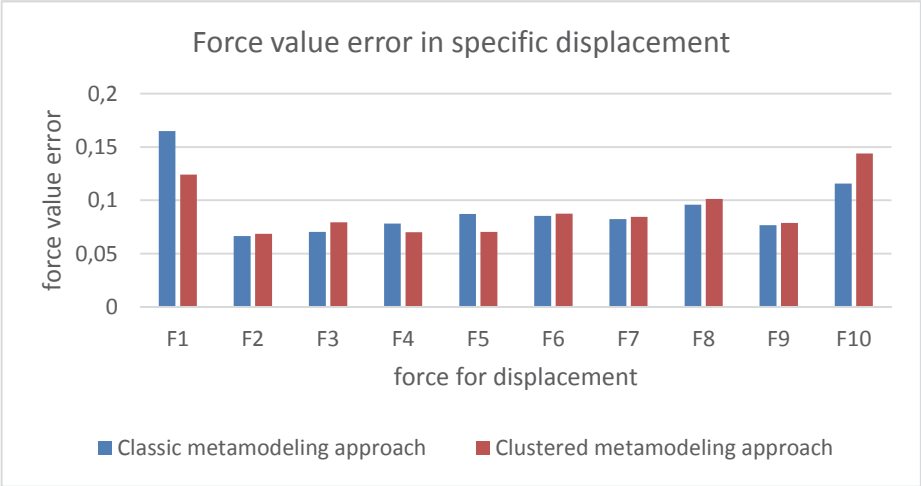


Fig. 14. *Force error values comparison for classic and clustering approach. Error has been computed with Φ_{RMSE} formula (6) with appropriate j-index value. As we can see the most biggest errors are for F1 and F10.*

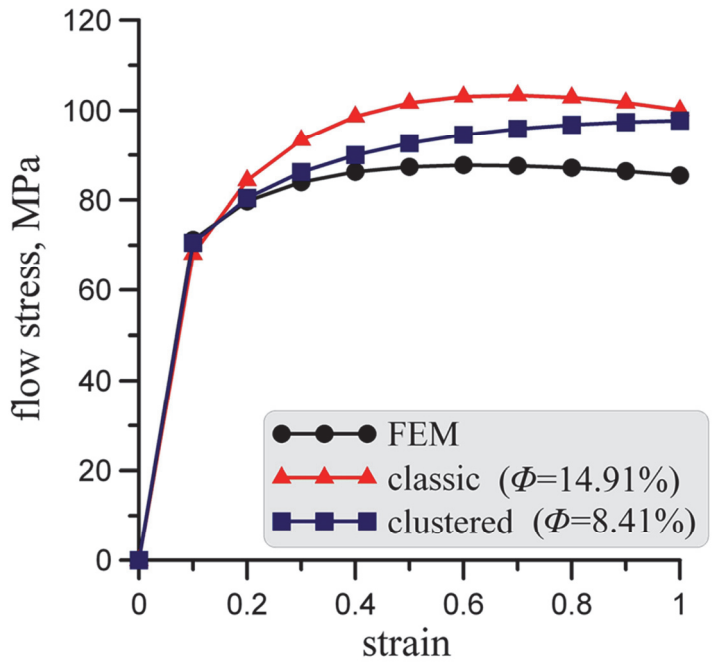


Fig. 15. *The comparison of material models for three different approaches applied in the inverse analysis.*

Table 2. *The results of inverse analysis with different approaches*

Approach	p_1	p_2	p_3	p_4	p_5	p_6	p_7
FEM (Φ=0.075)	2.423	4517.1	0.208	0.122	0.00519	10956	0.534
Classic ANN (Φ=0.114)	2.980	4733.3	0.392	0.103	0.234	3042.4	0.589
Clustered ANN (Φ=0.062)	1.509	5104.5	0.217	0.112	3.339	929.46	0.211

As we can see the most biggest errors are for F1 and F10. In F1 case it can be determined by very small forces values and working ranges of clustered networks. For more bigger forces number of samples growing down so ranges must to be bigger to get enough samples to make training effective. Going into this side we coming back first approach (one big training set).

Additionally clustering was verified with practical problem. Compression tests for the DP steel containing 0.11% C, 1.45% Mn, 0.19% Si, 0.27% Cr, 0.04% Cu, 0.042% Al, 0.013% Ti, 0.004% N were performed. This steel has been widely investigated by the Authors as far as manufacturing of automotive parts is considered. See papers (Pietrzyk et al., 2009; Kuziak & Pietrzyk, 2011, Sztangret et al., 2011b) for details. Effectiveness of clustering method was compared with the conventional inverse approach based on the FE model. Coefficients in equation (4) calculated using the three methods are given in table 2.

The coefficients obtained from inverse analysis and introduced to equation (4) allowed to obtain plots of the flow stress, which are presented in figure 14. The error of approximation obtained by using classic and clustering-based approaches were evaluated as Mean Square Errors (MSE) between these curves and results obtained by using FEM. The values are also presented in figure 15.

Final values of the objective function defined by equation (3) are given in figure 14. These values represent the accuracy of the inverse solution. The plots in figure 11 show comparison of experimental results (filled points) and the results of the inverse

analysis with metamodel (dashed lines) and the results of the conventional inverse analysis with FE (solid lines). It can be concluded that very good agreement between the results based on the two inverse approaches was obtained. Some discrepancies between numerical calculations and experimental data are due to the lack of the capability of equation (4) to describe material response in a wide range of temperatures and strain rates.

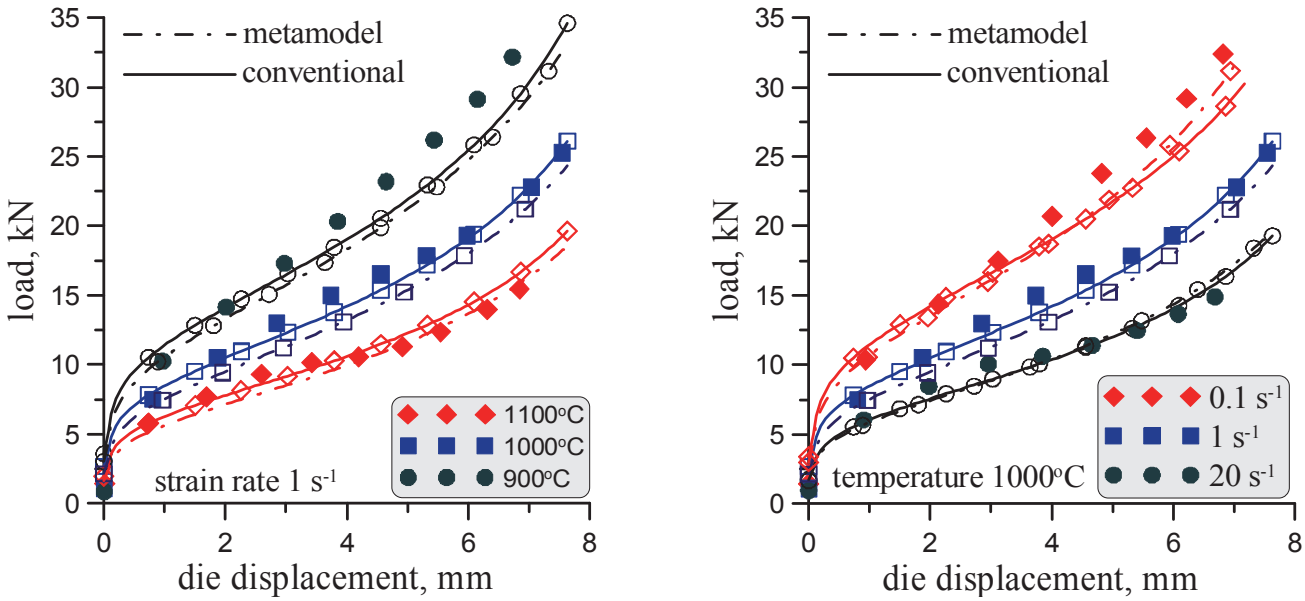


Fig. 16. *Verification of the clustering of the network in the inverse analysis. Filled points represent experiment, lines represent FE simulations of the tests with equation (4) used as the flow stress model in the constitutive law. Coefficients in equation (4) were determined using inverse analysis with the FE direct model (solid lines) and inverse analysis with metamodel (dashed lines).*

7. CONCLUSIONS

The use of metamodelling approach allows significant reduction of computational time of the inverse analysis. Moreover, performed analysis showed that for reasonably simple material models results obtained from the inverse solution with the metamodel are very close to those obtained from the inverse solution with the FE model. The difference between the two solutions is negligible. Clustering of the ANN was proposed for the more complex material models. Analysis of results obtained for clustered networks allowed drawing the following conclusions:

- Computing times of the inverse analysis with metamodel are few orders of magnitude shorter comparing to the classical inverse with the FE model. In consequence, advanced optimization techniques could be used in the former approach. Much better search through the domain of variables was possible and, in some cases, accuracy of the inverse analysis with the metamodel was better than that for the classical approach.
- An advantage can be made from the fact that in the inverse analysis the location of the optimal value of the model output is known – it has to be equal to the experimental value. It allows clustering of the networks with respect to the value of the output. Significant increase of the accuracy of predictions was obtained due to clustering.
- Clustering of the networks could be useful for solving more complex identification problems.
- Beyond this, the following detailed conclusions were drawn on the basis of numerical tests, which were performed:
- Networks trained at smaller ranges are more effective during training and they give smaller errors.
- Computing times of the inverse analysis are still short despite two times more calls of the direct model (the ANN), because the time of one metamodel computation is in the order of 0.1 ms.
- More effective approach is retrain clustered network from existing one in comparison to network with new architecture or weights.
- Clustering is based on similar known approaches – divide problem into smaller problems.

ACKNOWLEDGEMENTS

Financial assistance of the NCN, project N N508 629 740, is acknowledged.

REFERENCES

Ahmad, F., Matisa, N.A., Hussain, Z., Boudville, R., Osman, M.K., 2010, Genetic Algorithm - Artificial Neural Network (GA-ANN) hybrid intelligence for cancer diagnosis, *Int. Conf.e on Computational Intelligence*, Liverpool, 78-83.

Bezdek, J. C., Tsao, E.C. Pal, N.K., 1992, Fuzzy Kohonen clustering networks, *Proc. IEEE International Conference on Fuzzy System*, San Diego, 1035-1043.

Forestier, R., Massoni, E., Chastel, Y., 2002, Estimation of constitutive parameters using an inverse method coupled to a 3D finite element software, *Journal of Materials Processing Technology*, 125, 594-601. // 1-s2.0-S0924013602004065-mainrrr.pdf

Gawąd, J., Kuziak, R., Madej, Ł., Szeliga, D., Pietrzyk, M., 2005, Identification of rheological parameters on the basis of various types of compression and tension tests, *Steel Research International*, 76, 131-137.

Gavrus, A., Massoni, E., Chenot, J.L., 1996, An Inverse Analysis Using a Finite Element Model for Identification of Rheological Parameters, *Journal of Materials Processing Technology*, 60, 447-454.

Gelin, J.C., Ghouati, O., 1994, The inverse method for determining viscoplastic properties of aluminium alloys, *Journal of Materials Processing Technology*, 34, 435-440.

Hadasik, E., Kuziak, R., Kawalla, R., Adamczyk, M., Pietrzyk, M., 2006, Rheological model for simulation of hot rolling of new generation steel strips for automotive industry, *Steel Research International*, 77, 927-933.

Kermani, B.G., White, M.W., Nagle, H.T., 1995, Feature extraction by genetic algorithms for neural networks in breast cancer classification, *Engineering in Medicine and Biology Society, IEEE 17th Annual Conference*, 1, 831-832.

Khoddam, S., Lam, Y.C., Thomson, P.F., 1996, Numerical results achieved with an inverse computational method for determining the constitutive parameters using the hot torsion test results, *Steel Research*, 67, 39-43.

Kuncheva, L.I., 2005, *Combining pattern classifiers, Methods and Algorithms*, Wiley, New York.

Kusiak, J., Sztangret, Ł, Rauch, Ł., Pietrzyk, M., 2014, Metamodel driven optimization of thermomechanical industrial processes, *Computer Methods in Materials Science*, 14, 20-26.

Kuziak, R., Pietrzyk, M., 2011, Physical and numerical simulation of the manufacturing chain for the DP steel strips, *Steel Research International*, special edition conf. ICTP, 756-761.

Lazarevic, A., Obradovic, Z., 2001, Effective pruning of neural network classifier ensembles, *Proc. Int. Joint Conf. on Neural Networks*, 2, 796-801.

MacKay, D.J.C., 1992, Bayesian interpolation, *Neural Computation*, 4, 415-447.

Parvin, H., Alizadeh, H., Minaei-Bidgoli, B., Analoui, M., 2008, CCHR: Combination of classifiers using Heuristic Retraining, *Int. Conf. on Networked Computing and advanced Information Management (NCM 2008)*, Gyeongju.

Pietrzyk, M., Kusiak, J., Kuziak, R., Zalecki, W., 2009, Optimization of laminar cooling of hot rolled DP steels, *Proc. XXVIII Verformungskundliches Kolloquium*, Planneralm, 285-294.

Szeliga, D., Matuszyk, P., Kuziak, R., Pietrzyk, M., 2002, Identification of rheological parameters on the basis of various types of plastometric tests, *Journal of Materials Processing Technology*, 125-126, 150-154.

Szeliga, D., Gawąd, J., Pietrzyk, M., 2006, Inverse Analysis for Identification of Rheological and Friction Models in Metal Forming, *Computer Methods in Applied Mechanics and Engineering*, 195, 6778-6798.

Szeliga, D., Pietrzyk, M., 2012, Identification of rheological models and boundary conditions in metal forming, *International Journal of Materials and Product Technology*, 39, 388-405.

Szeliga, D., 2013, *Identification problems in metal forming. A comprehensive study*, Publisher AGH, 291, Kraków.

Sztangret, Ł., Milenin, A., Sztangret, M., Walczyk, W., Pietrzyk, M., Kusiak, J., 2011a, Computer aided design of the best TR forging technology for crank shafts, *Computer Methods in Materials Science*, 11, 237-242.

Sztangret, Ł., Szeliga, D., Kusiak, J., Pietrzyk, M., 2011b, Identification of material properties of the DP steel based on plastometric tests and on industrial hot strip rolling, *Computer Methods in Materials Science*, 11, 542-550.

Sztangret, Ł., Szeliga, D., Kusiak, J., Pietrzyk, M., 2012, Application of the inverse analysis with metamodelling for the identification of the metal flow stress, *Canadian Metallurgical Quarterly*, 51, 440-446.

Tadeusiewicz R., Chaki R., Chaki N., 2014, Exploring Neural Networks with C#, CRC Press, Taylor & Francis Group, Boca Raton.

Tadeusiewicz R., 2010, New Trends in Neurocybernetics. *Computer Methods in Materials Science*, 10, 1, 1-7.

PROGRAMOWANIE GENETYCZNE W STEROWANIU OTWIERANIEM DYSZY KADZI

Streszczenie

Analiza odwrotna, w której wykorzystuje się metamodele staje się coraz bardziej popularna. Aby poprawić jakość metamodeli w przedstawianej publikacji zaproponowano klasteryzację, w której do podziału sieci neuronowych na mniejsze wykorzystano wartości wyjściowe badanego problemu. Przedstawiona idea bazuje na rozbiciu danych wykorzystując informację o wartościach wyjściowych odwzorowywanej funkcji, które w przypadku analizy odwrotnej są znane jako cześć wartości wyjściowych funkcji celu. Dla każdego wydzielonego w ten sposób zestawu danych wyuczono osobną sieć neuronową. Aby możliwa była współpraca takich sieci wytrenowano także sieć na całym zakresie danych oraz stworzono specjalne mapy, dzięki którym możliwe było jednoznaczne określenie, która z sieci wytrenowana na wąskim zakresie danych powinna zostać użyta w specyficznych sytuacjach. Dzięki takiemu podejściu możliwe jest zweryfikowanie czy sieć wytrenowana na wąskim zakresie danych, która została ostatecznie wykorzystana do obliczenia wartości wyjściowej została poprawnie wybrana, co stanowi dużą zaletę dla tego podejścia. Przeprowadzone testy dla identyfikacji naprężenia uplastyczniającego potwierdziły bardzo dobrą jakość wyników dla zaproponowanego podejścia.

Received: July 31, 2014
Received in a revised form: October 23, 2014
Accepted: November 12, 2014