

# Determination of Relaxation Modulus of Time-Dependent Materials Using Neural Networks

Alexandra Aulova<sup>1\*</sup>, Edvard Govekar<sup>2</sup>, Igor Emri<sup>1</sup>

<sup>1</sup> *Center for Experimental Mechanics, Faculty of Mechanical Engineering, University of Ljubljana*

*Aškerčeva 6, Ljubljana 1000, Slovenia,*

tel. +368 1 6207 100

fax. +368 1 6207 110

e-mail: alexandra.aulova@fs.uni-lj.si

<sup>2</sup> *Laboratory of Synergetics, Faculty of Mechanical Engineering, University of Ljubljana,*

*Aškerčeva 6, Ljubljana 1000, Slovenia*

## Abstract

Health monitoring systems for plastic based structures require capability of real time tracking changes in response of polymer based structures, related to their time-dependent behavior. The paper proposes artificial neural networks as a tool of solving inverse problem appearing within time-dependent material characterization, since the conventional methods are computationally demanding and cannot operate in a real time mode. Abilities of a Multilayer Perceptron (MLP) and a Radial Basis Function Neural Network (RBFN) to solve ill-posed inverse problems on an example of determination of a time-dependent relaxation modulus curve segment from constant strain rate tensile test data are investigated. The required modelling data composed of strain rate, tensile and related relaxation modulus were generated using existing closed-form solution. Several neural networks topologies were tested with respect to the structure of input data and their performance was compared to an exponential fitting technique. Selected optimal topologies of MLP and RBFN were tested for generalization and robustness on noisy data; performance of all the modeling methods with respect to the number of data points in the input vector was analyzed as well. It was shown that MLP and RBFN are capable of solving inverse problems related to the determination of time dependent relaxation modulus curve segment. Particular topologies demonstrate good generalization and robustness capabilities, where the topology of RBFN with data provided in parallel proved to be superior compared to other methods.

*Keywords: relaxation modulus, inverse problem, neural network, multilayer perceptron, radial basis function neural network, structural health monitoring*

# 1 Introduction

Plastics and plastic based composites are slowly replacing metals in automotive and aeronautical industries, which is mainly due to their more favorable strength-to-weight ratio. However, with all advantages of plastics their use for highly demanding engineering applications on which human lives depend, requires exact predictions of durability and lifespan of polymeric structures. Unfortunately, standardized procedures for this do not exist yet.

Control of durability of structures made of elastic materials, such as metals, can be accomplished by health monitoring systems, that are commercially available. But in case of viscoelastic materials including plastics and polymers their time-dependent properties and effects related to it should be taken into account. In order to detect changes in material behavior, its material transfer functions should be tracked and calculated based on the response of a structure to external excitations. This means that the method used for plastics structures health monitoring should be able to comprehend time-dependent material transfer functions that affect structural responses.

One of the most important material transfer functions, which requires monitoring is time-dependent relaxation modulus. This transfer function describes process of relaxation, which appears as a decrease of stress under condition of constant deformation and can be detected as softening of material. Relaxation modulus jointly with geometry of the construction determines its stiffness and strength, and therefore, should be known for construction purposes. Typically, relaxation modulus is determined by tensile tests (ISO 527-1 2012), however, the standardized tests do not provide information on time-dependency of the material behavior. Measurements of time-dependent relaxation modulus are not standardized and require either very long time or according to the principle of time-temperature superposition (Ferry, 1980) – measurements at different temperatures. The second approach is most widely used, however its drawback is that for each of the measurements (segments) at a certain temperature an inverse problem of obtaining relaxation modulus from measured stress and applied strain data has to be solved. This problem has analytical solution for standard excitations (step and sine), while for non-standard or with presence of noise in the read signal it turns into ill-posed

inverse problem. Such problems can be solved only numerically by time-demanding mathematical techniques, e.g., exponential fitting (Saprunov et al., 2014) or regularization (Tikhonov & Arsenin, 1977). Due to long calculation time, these methods are not applicable for structural health monitoring in real time. On the other hand, artificial neural networks (NN) have been proven as a suitable tool to solve inverse problems and for real-time applications (Xiao et al., 2006). Once trained NN are able to deliver results fast, they are capable of parallel calculations due to their nature, and they are able to generalize and process noisy data.

Therefore, the paper proposes artificial neural networks (NN) for obtaining a segment of relaxation modulus curve based on the tensile data from constant strain rate experiment. This initial step is essential for further application of NN as a tool for health monitoring of polymeric structures in automotive, railway, or aeronautical applications. Neural network for this purpose should be capable of solving inverse problem for obtaining time-dependent material properties in order to track qualitatively changes caused by viscoelastic nature of materials from which structure is built. Of course, obtaining time-dependent material functions as a prime task is not the main purpose of NN because there are many closed-form solutions available, see (Saprunov et al., 2014) and references herein.

As an initial step it is proposed to apply Multilayer Perceptron (MLP) neural network with sigmoidal activation function and Radial Basis Function Network (RBFN) with Gaussian activation function for solving inverse problem arising within characterization of time-dependent properties of viscoelastic materials. As a reference method non-linear exponential parametric regression is used.

Two different topology types with respect to the structure of the input data for MLP and RBFN were tested. Optimal number of neurons in hidden layers was chosen according to an optimization criterion. For investigating the NN performance we have used an example for which the closed-form solution is known (Saprunov et al., 2014) and we could generate the NN training data numerically. In order to check generalization capabilities and robustness of the networks, the validation of NN performance was done on the noisy set of data that were not used for training.

With the aim to present the capabilities of the NN to estimate the relaxation modulus from the measured stress and applied strain data as first in the following sections

the problem statement is presented in detail. Then the process of NN implementation starting with the training data generation using closed-form solution, is followed by the description of the methodology of optimal topologies choice and their validation. Results and discussion section presents generalization and robustness capabilities of the obtained networks in comparison to the exponential fitting numerical technique.

## 2 Stress, strain and relaxation modulus

The following section presents the constitutive relations between strain and stress within relaxation process, problem related to determination of relaxation modulus and process of data generation.

### 2.1 Theory and problem

Constitutive relation between strain excitation  $\varepsilon(t)$  and stress response  $\sigma(t)$  in relaxation process of a time-dependent material under uniaxial stress state is given as:

$$\sigma(t) = \int_0^t E(t - \tau) \frac{\partial \varepsilon}{\partial \tau} d\tau. \quad (1)$$

Here  $E(t)$  is a time-dependent relaxation modulus and it is a material function of interest. Eq. (1) represents a convolution integral equation that has an analytical solution only for standard types of excitation including a step function and a harmonic excitation. In the case of other excitation functions, for determination of the relaxation modulus  $E(t)$  from given strain excitation  $\varepsilon(t)$  and stress response  $\sigma(t)$ , application of numerical techniques for solution of the related ill-posed inverse problem is required.

Existing numerical methods of solving ill-posed inverse problems are based on adding some disturbance (additional restriction) to the initial problem to turn it into close to the original but not ill-posed problem. In mathematics this approach is called regularization (Samarskii et al., 2009). The most well known and widely used regularization technique is the one introduced by the Russian mathematician Tikhonov in 1955 (Tikhonov & Arsenin, 1977). This approach is currently prevalent in linear theory of viscoelasticity. It is important to mention that even

though Tikhonov regularization technique and its derivatives are widely and successfully used in solving ill-posed inverse problems, this group of techniques is computationally- and time- demanding, and mathematically challenging and, consequently, not suitable for health monitoring of plastics-based structures. Due to this, the paper introduces an empirical modeling approach based on artificial neural networks, known by their high computational and generalization capabilities, and robustness.

## 2.2 Modelling data generation

Data for training neural network should be chosen thoughtfully due to the fact that NN can properly function only within the range of data that was covered during its training. Training data for this work were generated artificially using closed-form solution, which does not represent an inverse problem (Saprunov et al., 2014). Data consists of vectors of strain  $\boldsymbol{\varepsilon}$ , stress  $\boldsymbol{\sigma}$ , time  $\mathbf{t}$  and corresponding to them vector of relaxation modulus  $\mathbf{E}$ .

The data generation procedure is schematically shown in Fig. 1 and consists of two steps including determination of relaxation modulus  $E(t)$  and related stress response  $\sigma(t)$  for a given strain excitation  $\varepsilon(t)$  respectively. To determine the relaxation modulus  $E(t)$  as first generation of relaxation mechanical spectrum  $H_i(\tau_i)$  representing different engineering materials is performed. Relaxation mechanical spectrum  $H_i(\tau_i)$  is a characteristic of a polymeric material and constitutes its transfer function. It describes contributions of groups of molecules of different size/length to the overall response of a material to an external excitation.

Magnitudes of relaxation spectrum lines  $H_i(\tau_i)$  were determined according to the Gaussian distribution:

$$H_i(\tau_i) = \frac{\exp\left(-\frac{(\tau_i - \mu)^2}{2}\right)}{\sigma_G \sqrt{2\pi}}, \quad i=1, \dots, N \quad (2)$$

where  $\tau_i$  is a response time of a particular material molecular group,  $\mu$  is a mean value of distribution and was taken to be 0,  $i$  is a number of molecular group and  $\sigma_G$  is a standard deviation of Gaussian distribution which was varied from 0.4 to 1.6 with a step of 0.1.  $N=49$  spectrum lines were equally distributed in a logarithmic

time scale  $\log(\tau_i)$  with the step 0.5 from value -12 to 12 Table 1. Afterwards, the obtained values were normalized according to:

$$h_i = \frac{H_i}{\sum_{j=1}^{N=49} H_j}, \quad i=1, \dots, N. \quad (3)$$

As a result,  $l=13$  different relaxation spectra for  $\sigma_G \in [0.4, 1.6]$  with step 0.1 were generated, using parameters displayed in Table 1. The first diagram on the left in Fig. 1 shows examples of the three selected relaxation spectra for  $\sigma_G=0.4, 1.0$  and 1.6.

Knowing equilibrium values of relaxation modulus and its normalized value of spectra  $h_i$  allows one to calculate the whole relaxation modulus  $E(t)$  curve according to the formula:

$$E(t) = E_0 + (E_g - E_0) \sum_{i=1}^{N=49} h_i \cdot e^{-\frac{t}{\tau_i}}, \quad (4)$$

where  $E_0$  and  $E_g$  are fixed equilibrium and glassy relaxation moduli values. In our case the relaxation modulus  $E(t)$  was calculated as a segment corresponding to the time from 0 to 50 seconds. The number of data points representing  $E(t)$  in the time interval was varied as  $n=10, 50$  and 100 data points.

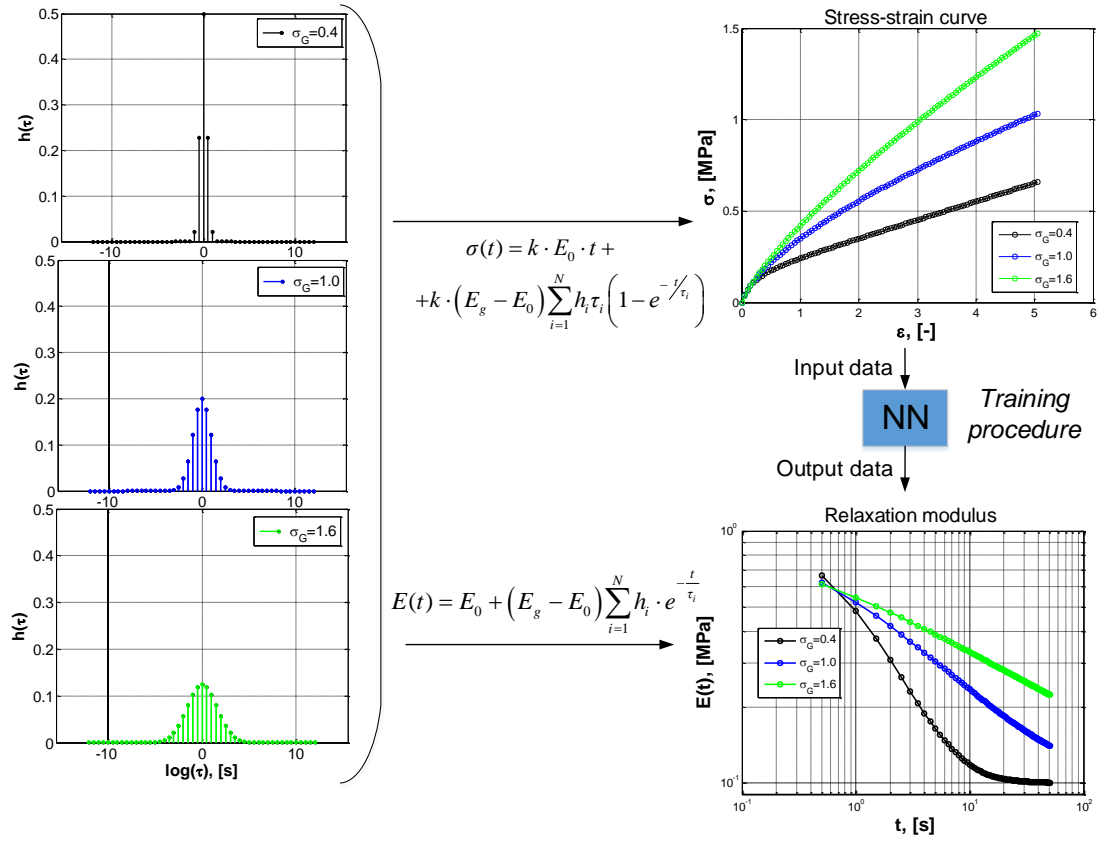
In the next step the stress response  $\sigma(t)$  of the material is calculated by the constitutive equation (1) taking into account defined  $E(t)$  and excitation strain  $\varepsilon(t)$ . Considering a constant strain rate the strain changes according to:

$$\varepsilon(t) = k \cdot t, \quad (5)$$

where  $k$  is a constant strain rate and is taken as 0.1 (ISO 527-1 2012). In this case Eq. (1) with incorporation of Eq. (4) and after integration turns into:

$$\sigma(t) = k \cdot E_0 \cdot t + k \cdot (E_g - E_0) \sum_{i=1}^N h_i \tau_i \left( 1 - e^{-\frac{t}{\tau_i}} \right). \quad (6)$$

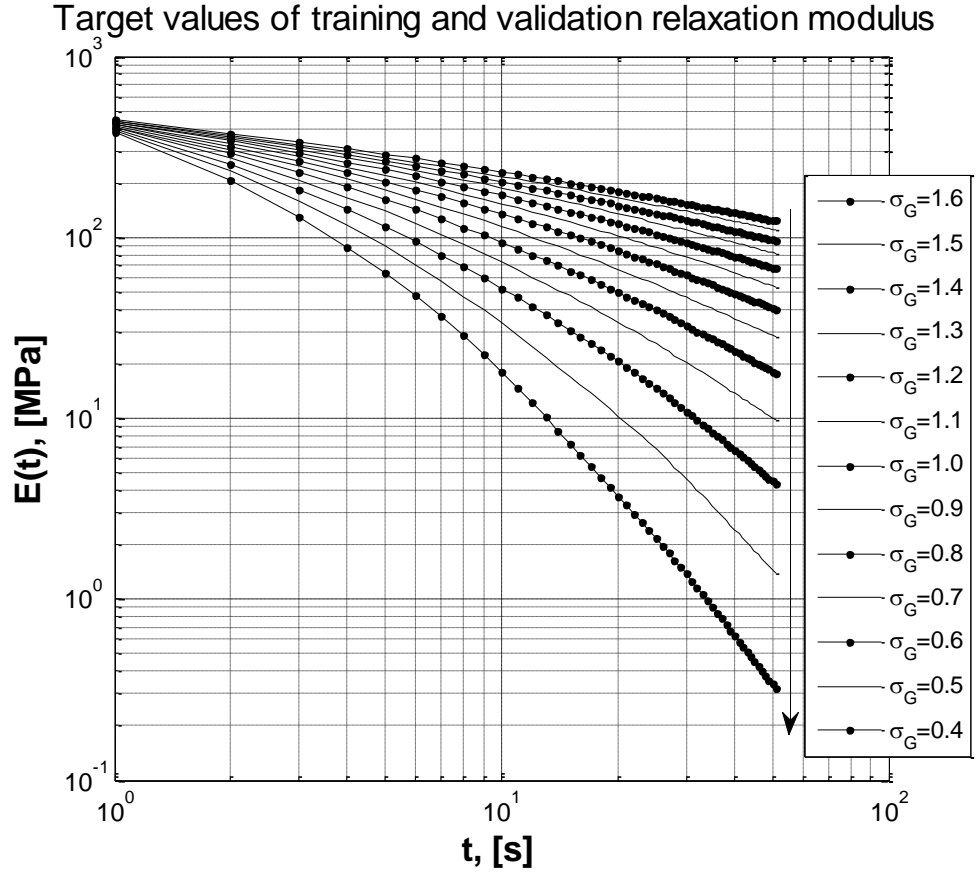
Eq. (6) allows calculation of stress response  $\sigma(t)$  to constant strain rate input (Eq. (5)) using analytical closed-form solution of Eq. (1). Values of the used constants and parameters of calculation are given in Table 1.



**Fig. 1.** Training data generation procedure on examples of three materials

**Table 1.** Parameters of training data generation

Parameter, [units]	Value
Standard deviation, $\sigma_G$	0.4..1.6, step 0.1
Number of spectrum lines, $N$	49
Step for spectrum lines along logarithmic timescale, [s]	0.5
Position of the first spectrum line in logarithmic timescale, [s]	-12
Start time of calculation in real timescale, [s]	0
End time of calculation in real timescale, [s]	50
Glassy modulus, $E_g$ , [MPa]	1000
Equilibrium modulus $E_0$ , [MPa]	0.1



**Fig. 2.** Target relaxation moduli segments for  $n=50$  data points

The calculated  $l=13$  relaxation moduli for  $\sigma_G \in [0.4, 1.6]$  with step 0.1 are shown in Fig. 2. Among the  $l=13$  generated datasets  $(\epsilon, \sigma, t; \mathbf{E})$   $l_{tr}=7$  curves  $E(t)$  marked with circular markers and corresponding stress-strain data were taken as training data while the remaining  $l_{val}=6$  denoted with solid thin curves were taken as testing data to validate the modelling performance of NN.

In order to investigate how performance of MLP changes in respect to the number  $n=10, 50$  and  $100$  of data points used to represent the curve of relaxation modulus, the training data sets with different number of data points were generated.

Robustness of MLP will be checked on noisy data, therefore relative noise of values 1, 5, and 10% was added to the values of stress to simulate the real experiment.



### **3. Empirical models for inverse problem solution**

The literature shows that artificial neural networks are widely used for solution of inverse problems (Hagan et al., 1996) in a variety of fields including. For example, inverse problems appearing in such fields as electromagnetics (Sammany et al., 2010) (Elshafiey et al., 1994) (Elshafiey et al., 1995), material characterization (Czél et al., 2013) (Li et al., 2009) (Li et al., 2008), imaging techniques (Adler & Guardo, 1994) (Lampinen & Vehtari, 1999), geology (Baddari et al., 2010) and variety of other fields represent a challenge to solve. All of these problems are ill-posed according to definition of Jacques Hadamard (1902), due to presence of noise in measured data, and computationally demanding. Conventional ways of solving those inverse problems are not efficient, or are not applicable for particular conditions involved, i.e., for on-line monitoring, or in case of high levels of noise in input signal.

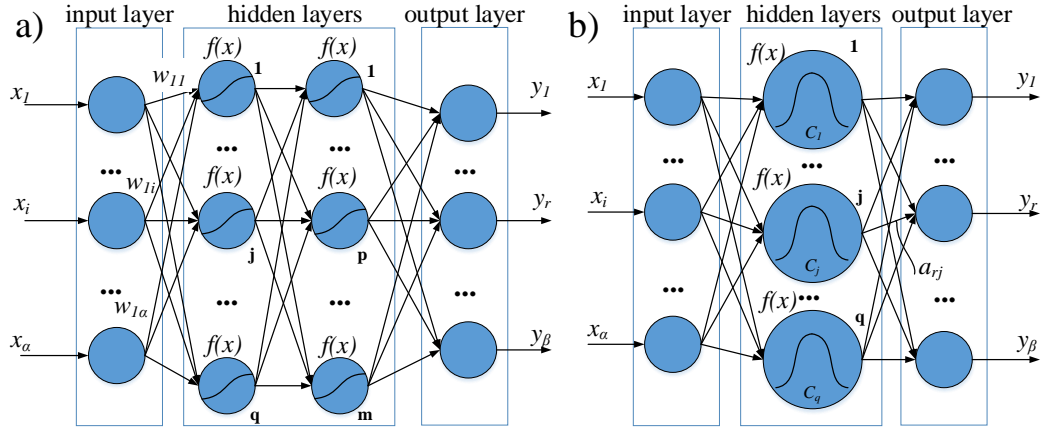
These are the reasons why scientists turned to artificial neural networks as a tool for solving inverse problems (Hagan et al., 1996). Although used in so many different applications there are practically no papers addressing the NN modeling of behavior of viscoelastic materials and determination of their time-dependent material functions such as relaxation modulus.

Therefore, in this paper Multilayer Perceptron and Radial Basis Function neural networks known as an universal function approximators have been used for determination of relaxation modulus segment from constant strain rate tensile experiment data. Determination of relaxation modulus from constant strain rate experiments was already addressed by number of other researchers that were using generalization or parametric regression methods (Tscharnuter et al., 2011) (Knauss & Zhao, 2007). Therefore comparison of neural network performance will be made with respect to the exponential parametric regression algorithm (nonlinear parametric regression) that showed good performance for this problem (Saprunov et al. 2014).

### 3.1 Neural Networks types

#### 3.1.1 Multilayer Perceptron (MLP)

Multilayer perceptron (MLP) is a feedforward artificial neural network consisting of fully interconnected neurons in several layers with non-linear activation functions (Haykin 1999) as shown in Fig. 3a. Number of hidden layers as well as neurons is arbitrary.



**Fig. 3.** Schematic representation of a) MLP topology, b) RBFN topology

MLP can be trained in a supervised manner with a very popular back propagation algorithm (Werbos 1994).

Output of the network with two hidden layers presented in Fig. 3, a will be:

$$y_r = \sum_p \omega_{rp} f \left[ \sum_j \omega_{pj} f \left( \sum_i \omega_{ji} x_i + c_j \right) + c_p \right], \quad (7)$$

where  $\alpha$  and  $\beta$  are numbers of inputs and outputs respectively, indexed by  $i$  and  $r$ ;  $q$  and  $m$  are number of neurons in hidden layers indexed by  $j$  and  $p$ . Synaptic weights are represented by  $\omega$  with two indices, where the first one presents the neuron accepting the signal for and the second – the neuron sending the signal.  $c_j$  and  $c_p$  are weights that are called biases and may not be present in a particular structure.

For solving the stated inverse problem  $(\mathbf{\varepsilon}, \mathbf{\sigma}, \mathbf{t}) \rightarrow \mathbf{E}$ , a MLP with classical logistic sigmoid activation function was utilized:

$$f(x) = \frac{1}{1 + e^{-x}}, \quad (8)$$

where  $f(x)$  is a neuron activation function of parameter  $x$ , which in case of neural network is presented as weighted and biased sum of values of neuron's input vectors of strain  $\boldsymbol{\varepsilon}$ , stress  $\boldsymbol{\sigma}$  and time  $\mathbf{t}$ . The output of a MLP is defined by vector of relaxation modulus  $\mathbf{E}$ .

Training data was rescaled into the range between 0 and 1. Nguyen-Widrow initialization algorithm (Nguyen & Widrow 1990) was used for initialization. It chooses initial values of synaptic weights in order to distribute the active region of each neuron in the layer approximately evenly across the layer's input space. The selection of values contains a degree of randomness, so they are not the same each time this function is called (Demuth et al. n.d.). In order to avoid this randomness during MLP topology optimization process generator of random number in MATLAB is set to default values before initialization function is called.

There is a variety of training functions available for MLP, however, in order to avoid overfitting backpropagation algorithm with Bayesian regularization is chosen (Demuth et al. n.d.). More information on Bayesian networks can be found elsewhere (MacKay 1992).

### 3.1.2 Radial Basis Function Neural Network

Radial Basis Functions Neural Networks (RBFN) is a type of feedforward neural network that utilize radial-basis activation functions and typically consist of 3 layers with different roles as shown in Fig. 3b. The input layer is made up of source nodes (inputs) that connect network to its environment. The second layer, which is the only hidden layer in the RBFN, applies a nonlinear transformation from the input space to the hidden space which in the most applications is of high dimensionality. The output layer is linear, supplying the response of the network to the input signal. Within this work Gaussian Radial Basis activation function was used:

$$f(x) = e^{-\frac{x^2}{2s^2}}, \quad (9)$$

where  $s$  is the standard deviation of Gaussian distribution. The output of the RBFN will be a scalar function of real input vector  $\mathbf{X}$  consisting of  $\alpha$  components:

$$y_r = \sum_{j=1}^q a_{rj} f(\|\mathbf{X} - C_j\|), \quad (10)$$

where  $q$  is a number of neurons in a hidden layer,  $C_j$  is the center vector of a neuron  $j$ ,  $f$  is a radial basic function of a neuron (Eq.(9)) and  $a_j$  is the weight of a neuron  $j$  in the linear output layer. The norm is typically taken in Euclidean space.

RBFN was formed by consecutive addition of the neurons with Gaussian Radial Basis Function in order to satisfy the condition of the error 0.001 (MATLAB function *newrb*), therefore the only topological parameter to be determined is the spread (in notation of MATLAB spread equals to  $s \cdot \log(4) \approx 1.177s$ ) of the network. It was varied in wide ranges in order to determine the optimal values of spread resulting in good RBFN performance.

### 3.1.3 Topologies of NN

Topology of a neural network defines the way the neurons of particular layer are connected, and it is an important factor in network functioning and learning (Sammut & Webb 2011).

In supervised learning the most common topology is the fully connected, at least three-layer, feedforward network. In such a network all input values to the network are connected to all neurons in the first hidden layer, the outputs of the last hidden neurons are connected to all neurons in the output layer, which neurons activation function defines the output of the network.

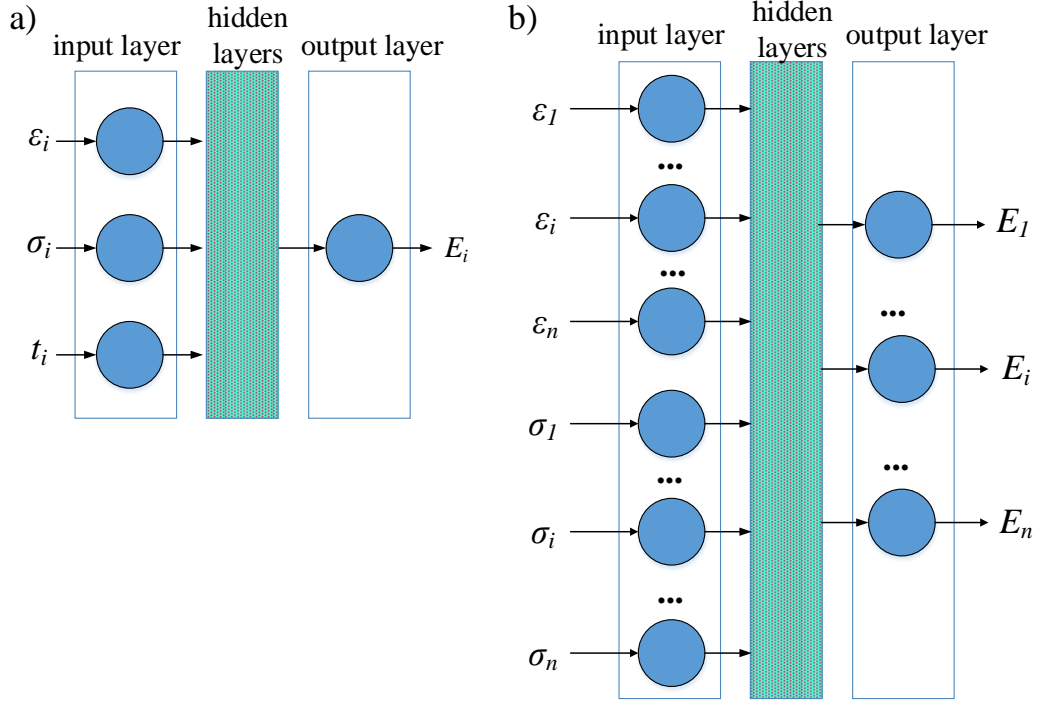
The number of hidden layers and related number of neurons determines computational capabilities of a network. According to universal approximation theorem a single hidden layer is sufficient for MLP to compute a uniform approximation to a given training set (Haykin 1999). Nevertheless, due to complexity of the problem and related poor results of single layered MLP, authors selected topology of MLP to two hidden layers with maximum  $q=m=30$  neurons in each (see Fig. 3, a). Number of variable parameters provided by such structure is more than sufficient for proper training but utilization of training algorithm with regularization term will provide network without tendency to overfitting (Sjoberg & Ljung 1992). For RBFN number of hidden neurons was determined by the algorithm used for neural network creation.

Once the number of the hidden layer and the related number of neuron is defined the neural network topology is also defined by the way of representation of input and related output data. In our case, two different ways of presenting input and output data were considered.

The first possibility is to provide data points of one training sample in series one by one, (Fig. 4, a) and therefore, provide mapping  $(\varepsilon_i, \sigma_i, t_i) \rightarrow (E_i)$ , while the other option is to provide data of the whole training set at once, “in parallel” (Fig. 4, b) and mapping  $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}) \rightarrow (\mathbf{E})$ .

In case of providing input data in series, the input data vector  $(\varepsilon_i, \sigma_i, t_i)$  is mapped into the related scalar value  $(E_i)$  representing the  $E$  at time  $t_i$ . The training time is significantly shorter and it mimics application for real time monitoring, however transition between two different training samples represents irregularity in the data. Values of time  $t_i$  in this case is required for introduction of time-dependency into the system, which is required for the solution of the stated inverse problem described by Eq. (1).

For parallel feeding strain and stress vectors  $(\boldsymbol{\varepsilon}$  and  $\boldsymbol{\sigma}$ , respectively) were given as inputs and mapped into the vector  $\mathbf{E}$ , representing the complete relaxation modulus function  $E(t)$ . In this case, since the whole strain and stress curves were provided as input each data point had its own input neuron (see Fig.4, b). Presenting the input data in parallel on one hand slows down the training procedure and results in complex topology (input neuron for each data point) but on the other hand provides to the network all the information on the curve that is necessary for solving inverse problem (all history of material behavior). In this case vector of time is not needed, since vectors of strain  $\boldsymbol{\varepsilon}$  and stress  $\boldsymbol{\sigma}$  fully presented to the network already contain all required information on time-dependency. Relaxation modulus values of vector  $\mathbf{E}$  were outputs. Number of output neurons corresponds to the  $n$  number of data points representing the curve of relaxation modulus  $E(t)$ .



**Fig. 4.** Schematic representation of topology with data provided a) in series and b) in parallel

### 3.2. Exponential parametric regression

As a reference modeling method an exponential parametric regression referred as exponential fitting (Saprunov et al. 2014), was used. The exponential fitting is used to represent Eq. (1) in general form:

$$\sigma(t) = c \cdot t + \sum_{i=1}^N a_i (1 - e^{b_i t}) \quad (11)$$

where  $i$  is an index of  $N$  spectrum lines. Using Eq. (12) in Eq. (1), taking into account Eq. (5) for constant change of strain  $\varepsilon(t)$  and deriving resulting equation with respect to time, relaxation modulus can be obtained in the closed form (Knauss & Zhao 2007):

$$E(t) = \frac{1}{k} \frac{\partial \varepsilon}{\partial t} = \frac{1}{k} \left( c - \sum_{i=1}^N a_i b_i e^{b_i t} \right). \quad (12)$$

The unknown coefficients  $c$ ,  $a_i$ ,  $b_i$  in Eq. (13) were determined by fitting the stress and strain data with Eq. (12) using an optimization criteria of minimum least-squares error and taking into account constraints caused by physical meaning of coefficients:

$$\begin{cases} c > 0, \\ a_i > 0, \\ -1/t_1 \leq b_i \leq -1/t_M, \end{cases} \quad i = 1, 2, \dots, N \quad (13)$$

where  $b_i$  was varied in the window corresponding to changes of relaxation times  $\lambda_i = 1/b_i$  within the experimental window  $t_1 \leq \lambda_i \leq t_n$  corresponding to the number of measurement points for every  $i=1, 2, \dots, N$  (Knauss & Zhao 2007). The optimization was done using Trust Region method implemented using “lsqcurvefit” of MATLAB Optimization Toolbox (Saprunov et al. 2014).

### 3.3. Measures of modeling performance

To evaluate and to compare the performance of particular NN and exponential fitting modelling the following measures have been used:

1. Mean square error,  $MSE$ , [MPa<sup>2</sup>] is calculated as follows:

$$MSE_j = \frac{\sum_{i=1}^n (E_{ji} - E_{ji}^{target})^2}{n}, \quad (14)$$

where  $j$  is an index representing the  $j$ -th testing curve  $E_j(t)$ ,  $E_{ji}$  is a value of the relaxation modulus predicted in the  $i$ -th point,  $E_{ji}^{target}$  is a true value of the relaxation modulus in the  $i$ -th point and  $n$  is a number of points representing the relaxation modulus  $E_j(t)$  curve. The lower  $MSE_j$  value the better performance is.  $MSE_j$  is calculated for each curve from  $j=1$  to  $l$  corresponding to the number of curves in the set of the testing set and is averaged over the number  $l$  of testing curves of curves analyzed.

2.  $R_{j,0.05}$ , [%] is defined as a percentage of the number  $n_{0.05}$  of data points of the  $E_j(t)$  curve that are estimated with equal or less than 5% relative error. This value of 5% relative error was chosen as maximal error acceptable for engineering purposes. The data points of a curve  $E_j(t)$  which are predicted with equal or less than 5% relative error satisfy the following condition:

$$E_i^{0.05} : \frac{(E_i - E_i^{target})}{E_i^{target}} \leq 0.05, \quad i = 1..n \quad (15)$$

The performance measure  $R_{j,0.05}$  of modelling of particular  $E_j(t)$  curve is determined as:

$$R_{j,0.05} = \frac{n_{0.05}}{n} \cdot 100\%, \quad (16)$$

where  $n$  is the number of the data representing the  $E_j(t)$  curve. The modelling performance of particular NN is then characterized as average value  $R_{0.05}$  of  $R_{j,0.05}$  with respect to the number of all analyzed testing curves  $E_j(t)$ .

With the aim to perform the validation of the modeling performance of NN and related choice of an optimal topology in two dimensional space an optimization criteria  $J$  was defined in such a way that maximization of  $J$  leads to an optimal performance of the NN. For this purpose the compliment  $R_{0.95}$  of  $R_{0.05}$  was introduced by:

$$R_{0.95} = 100\% - R_{0.05}, \quad (17)$$

determining a percent of data points that were calculated with error of more than 5%.

Then the normalized values  $R_{0.95}$  and  $MSE$  in the interval from 0 to 1 are calculated as:

$$R'_{0.95j} = \frac{R_{0.95j} - R_{0.95min}}{R_{0.95max} - R_{0.95min}}; \quad (18)$$

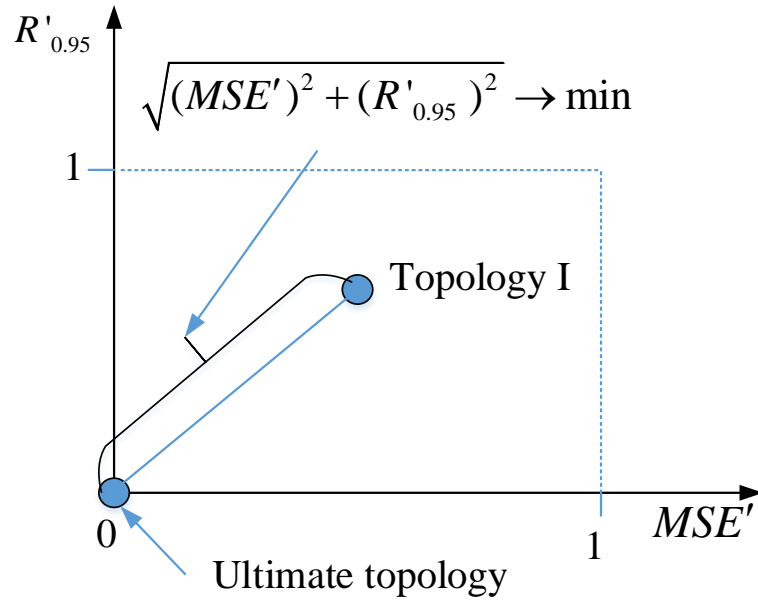
$$MSE'_j = \frac{MSE_j - MSE_{min}}{MSE_{max} - MSE_{min}}, \quad (19)$$

where  $R_{0.95min}$ ,  $R_{0.95max}$ , and  $MSE_{min}$ ,  $MSE_{max}$  are the related minimal and maximal values  $j$ -th curve. Using the average  $MSE$  and  $R_{0.95}$  for the whole training set the optimization criteria  $J$  is defined by:

$$J(MSE', R'_{0.95}) = \left( \frac{\sqrt{(MSE')^2 + (R'_{0.95})^2}}{\sqrt{2}} \right) \cdot 100\%. \quad (20)$$

Function  $J$  utilizes Euclidian distance in the space of  $MSE'$  and  $R_{0.95}'$  between the optimal zero values of both parameters and current topology as shown in Fig. 5. Minimization of the Euclidian distance will correspond to the minimal value of function  $J$ , which represents the best relaxation module  $E(t)$  function approximation.





**Fig. 5.** Schematic representation of topology optimization principle

### 3.4. Validation procedure

Validation NN modeling performance and determination of optimal topologies was done in three respects: 1) generalization. 2) robustness properties, and 3) performance of NN with respect to number of data points in the signal. First two are important for application of the NN, the third one is related to convergence of optimization algorithm within training. Comparison with exponential fitting for robustness and effect of number of data points is done. Data sets representing the  $E(t)$  curves with  $n=10, 50$  and  $100$  data points are considered.

1. Generalization of the particular NN was checked on  $l_{val}=6$  datasets representing 6 curves  $E(t)$  that were not used for training.
2. Robustness test of the NN was done based on data sets including data for training and validation ( $\varepsilon, \sigma, t; \mathbf{E}$ ) to the stress component of which 1, 5 and 10% relative noise was added. Noise levels higher than 10% are not considered, since errors of measurements caused by sensors will not exceed this value.
3. Mathematical convergence of training algorithm and performance of the trained NN were analyzed in respect to number  $n$  of data points in the input signal.

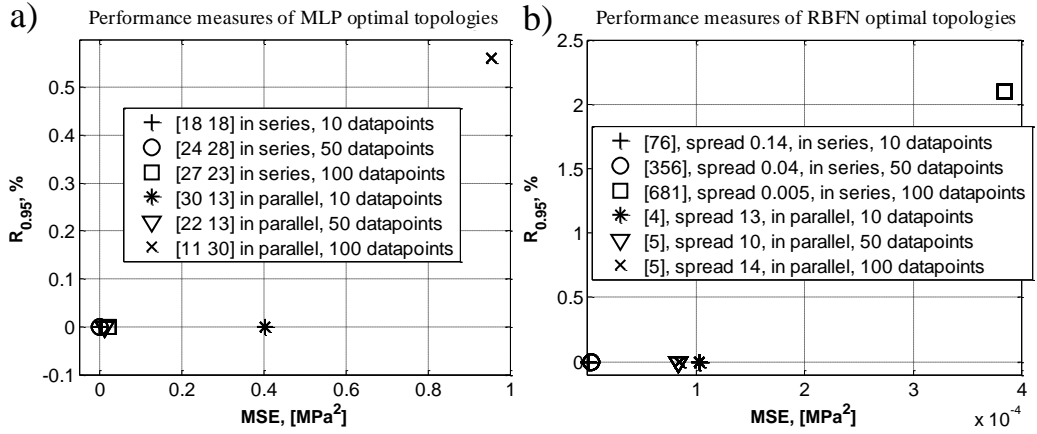
## 4 Results and Discussion

Within this section the optimization results and related NN topologies are presented and their performance is evaluated with respect to their generalization ability and robustness. In addition to this, the influence of the number of data points  $n$  used to present the  $E(t)$  curves is analyzed. The results are then compared with the modeling results obtained by the exponential fitting technique (Saprunov et al. 2014).

### 4.1 Choice and validation of NN topologies

All possible variations of maximum 30 neurons in 2 hidden layers (900 iterations) were tested and compared based on NN performance measures and introduced optimization criteria  $J$  presented in chapter 3.2. For RBFN number of hidden neurons was determined by the algorithm used for neural network creation.

Fig. 6 shows the results of NN performance in the plane of  $MSE$  and  $R_{0.95}$  for 6 different topologies of NN obtained by taking into account the defined criterion function  $J$  (eq. 21). Fig. 6a shows the related results for MLP for various number of neurons  $[q, m]$  in the two hidden layers and Fig. 6b the results for the RBFN with respect to number of data points ( $n=10, 50$  and  $100$ ) and way of the presentation (serial and parallel) of the input data to the NN.



**Fig. 6.** Performance parameters in the plane ( $MSE, R_{0.95}$ ) for the selected optimal topologies of a) MLP and b) RBFN

From the presented points in the plane  $MSE$  and  $R_{0.95}$  being closer to the origin of the plane ( $MSE, R_{0.95}$ ) it is visible that neural networks in the case when the data

were provided in series performed better than in the case when the data were provided in parallel.

In the following the selected topologies were validated in respect to their generalization abilities Robustness and influence of number of data points  $n$  used to present the  $E(t)$ . For the purpose of the NN validation the data set representing the modeling data of all  $l=13$  curves  $E(t)$  was divided into training and testing data set. The training data set consisted of the data representing the selected  $l_{tr}=7$   $E(t)$  curves while the rest  $l_{val}=6$  was used as testing data.

#### 4.1.1 Generalization

To demonstrate the generalization ability of the NN in tables 2 and 3 the values of NN performance parameters averaged for the training and validation data sets for MLP and RBFN are presented. As measures of performance the defined MSE for each selected topology for training and validation data, their absolute difference ( $\Delta MSE = |MSE_{val} - MSE_{tr}|$ ) and ratio ( $MSE_{val}/MSE_{tr}$ ) were used. In addition the efficiency  $R_{0.05}$  for training and validation data, as well as their absolute difference ( $\Delta R_{0.05} = |R_{0.05val} - R_{0.05tr}|$ ) are presented in the last three columns.

From the Table 2 it is visible that for MLP,  $MSE$  for training and validation data are of the same scale and are comparable. NN with both ways of representation of input data demonstrated decrease in generalization in respect to number of data points  $n_{0.05}$  reconstructed within 5% relative error  $R_{0.05}$  for the highest number of data points ( $n=100$ ) in the data set. Additionally a drop in performance and related decrease of generalization ability not detected by  $R_{0.05}$  is evident in decrease of  $MSE$  for validation set of data in the case of MLP with parallel input and  $n=50$  data points in the data set.

Table 3 presents results of generalization tests for RBFN. Here, on contrary to the MLP, better results are obtained for the networks with data presented in parallel, while RBFNs with data provided in series show poor performance in respect to both, the  $MSE$  and the  $R_{0.05}$  parameters. Among RBFNs with data provides in parallel the ideal values of  $R_{0.05}=100\%$  parameter were observed for validation data. The related ratio  $MSE_{val}/MSE_{tr}$  between training and validation increases up to 32. For RBFN this ratio is higher than the maximal ratio for MLP networks, and in

general indicates that MLP NNs showed better generalization capability than RBFNs.

**Table 2.** Comparison of MLP performance on training and validation data

Input type	Number of data points, $n$	Optimal topology $[q \ m]$	Training data, $MSE_{tr}$ , MPa <sup>2</sup>	Validation data, $MSE_{val}$ , MPa <sup>2</sup>	$\Delta MSE$ , MPa <sup>2</sup>	$\frac{MSE_{val}}{MSE_{tr}}$ , [-]	Training data, $R_{0.05tr}$ , %	Validation data, $R_{0.05val}$ , %	$\Delta R_{0.05}$ , %
series	10	[18 18]	$3.32 \cdot 10^{-4}$	$4.77 \cdot 10^{-4}$	$1.45 \cdot 10^{-4}$	1.44	100	100	0
series	50	[24 28]	$8.90 \cdot 10^{-5}$	$3.22 \cdot 10^{-4}$	$2.33 \cdot 10^{-4}$	3.62	100	100	0
series	100	[27 23]	$2.29 \cdot 10^{-2}$	$4.00 \cdot 10^{-2}$	$1.71 \cdot 10^{-2}$	1.75	100	99.22	0.78
parallel	10	[30 13]	$4.03 \cdot 10^{-1}$	$5.14 \cdot 10^{-1}$	$1.11 \cdot 10^{-1}$	1.27	100	91.67	8.33
parallel	50	[22 13]	$1.13 \cdot 10^{-1}$	$6.12 \cdot 10^{-1}$	$4.98 \cdot 10^{-1}$	5.39	100	100	0
parallel	100	[11 30]	$9.54 \cdot 10^{-1}$	2.08	1.13	2.18	99.45	81.86	17.58

**Table 3.** Comparison of RBFN performance on training and validation data

Input type	Number of data points, $n$	Optimal topology $[q]$ , spread	Training data, $MSE_{tr}$ , MPa <sup>2</sup>	Validation data, $MSE_{val}$ , MPa <sup>2</sup>	$\Delta MSE$ , MPa <sup>2</sup>	$\frac{MSE_{val}}{MSE_{tr}}$ , [-]	Training data, $R_{0.05tr}$ , %	Validation data, $R_{0.05val}$ , %	$\Delta R_{0.05}$ , %
series	10	[76], 0.14	$5.48 \cdot 10^{-8}$	3.67	3.67	$6.68 \cdot 10^7$	100	73.6	26.4
series	50	[356], 0.04	$2.99 \cdot 10^{-6}$	$2.04 \cdot 10^2$	$2.04 \cdot 10^2$	$6.83 \cdot 10^7$	100	32.7	67.3
series	100	[681], 0.005	$3.84 \cdot 10^{-4}$	$7.59 \cdot 10^3$	$7.59 \cdot 10^3$	$1.98 \cdot 10^7$	97.9	8.7	89.2
parallel	10	[4], 13	$1.03 \cdot 10^{-4}$	$5.12 \cdot 10^{-4}$	$4.09 \cdot 10^{-4}$	4.99	100	100	0
parallel	50	[5], 10	$8.33 \cdot 10^{-5}$	$2.6 \cdot 10^{-3}$	$2.52 \cdot 10^{-3}$	31.23	100	100	0
parallel	100	[5], 14	$8.43 \cdot 10^{-5}$	$2.7 \cdot 10^{-3}$	$2.62 \cdot 10^{-3}$	32.02	100	100	0

Considering split-sample validation of generalization of MLP and RBFN with two different ways of input data representation and 3 different number of data points in the set  $n$ , MLP showed better generalization abilities in respect to both  $MSE$  and  $R_{0.05}$ .

#### 4.1.2 Robustness

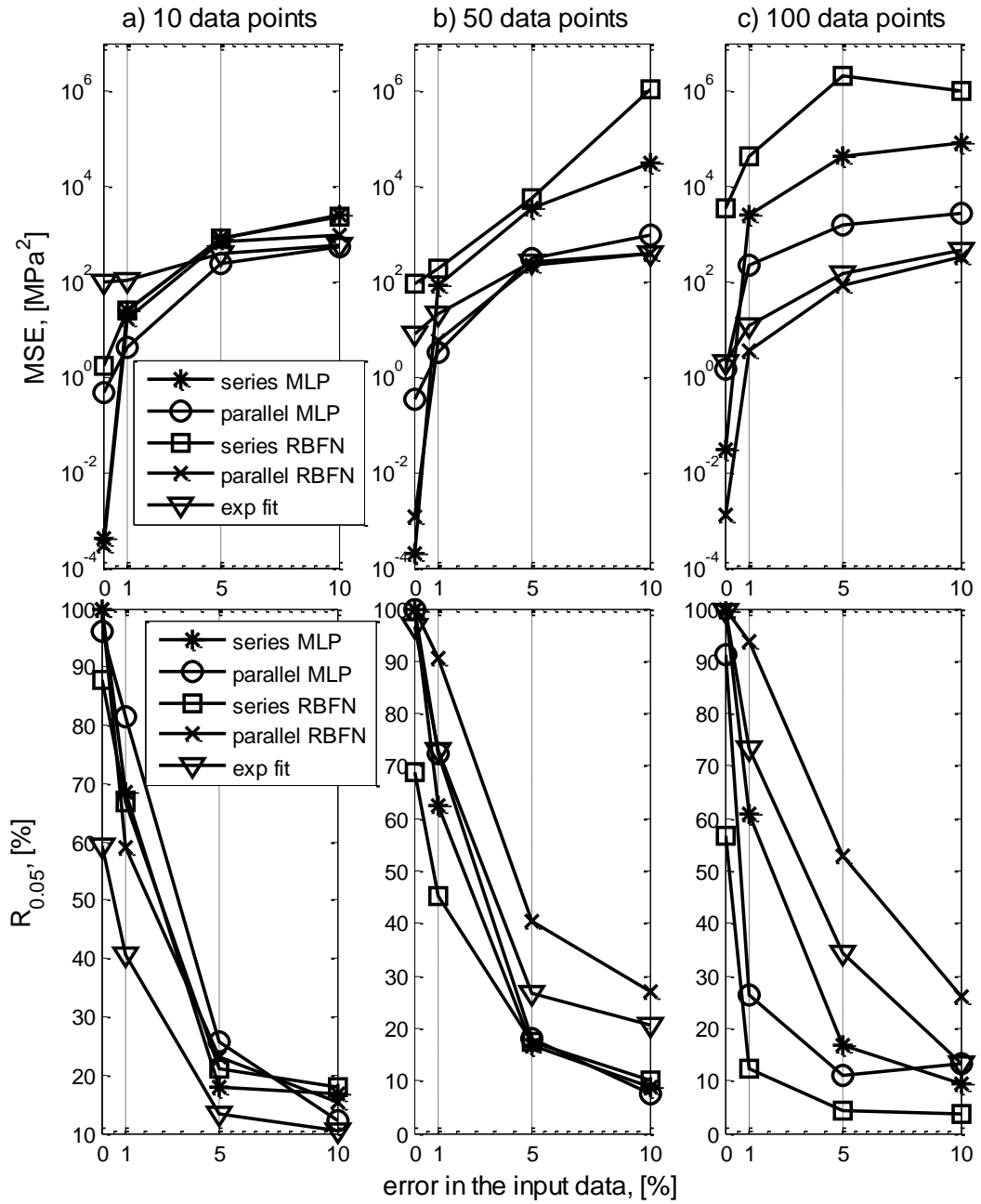
To analyze the robustness of NN performance data sets with relative additive noise of 1, 5 and 10% were considered in modeling the relation  $(\boldsymbol{\varepsilon}, \boldsymbol{\sigma}, \mathbf{t}) \rightarrow \mathbf{E}$ . Robustness of NN was compared to exponential fitting technique. Since the reference method of exponential fitting is sensitive to the width of the spectrum, the whole generated data set including data for training and validation was used into analysis. Graphs in Fig. 7 show averaged over complete data set  $l=13$  (training and validation) values

for  $MSE$  and  $R_{0.05}$  for all 6 topologies of each type of NNs and for the exponential fitting.

The best results for noiseless data in terms of average  $MSE$  and  $R_{0.05}$  were demonstrated by RBFN with data provided in parallel and MLP with data provided in series.

Further the MLP with the data provided in parallel shows the highest robustness for the noise level less than 10% and for small number of data points  $n=10$ . With increase of number of data points RBFN with data provided in parallel becomes competitive and overpasses the performance of the MLP. For the  $n=100$  data points in the data set in terms of both performance measures RBFN with the data presented in parallel is followed by exponential fitting.

Among the NN tested for robustness RBFN with data provided in parallel for number of data points  $n=50$  and 100 showed better results than numerical method of non-linear exponential parametric regression.



**Fig. 7.** Upper panel - Average  $MSE$  for  $n= 10, 50$  and  $100$  data points obtained by MLP, RBFN and exponential fitting. Bottom panel – Average  $R_{0.05}$  for  $n= 10, 50$  and  $100$  data points obtained by MLP, RBFN and exponential fitting

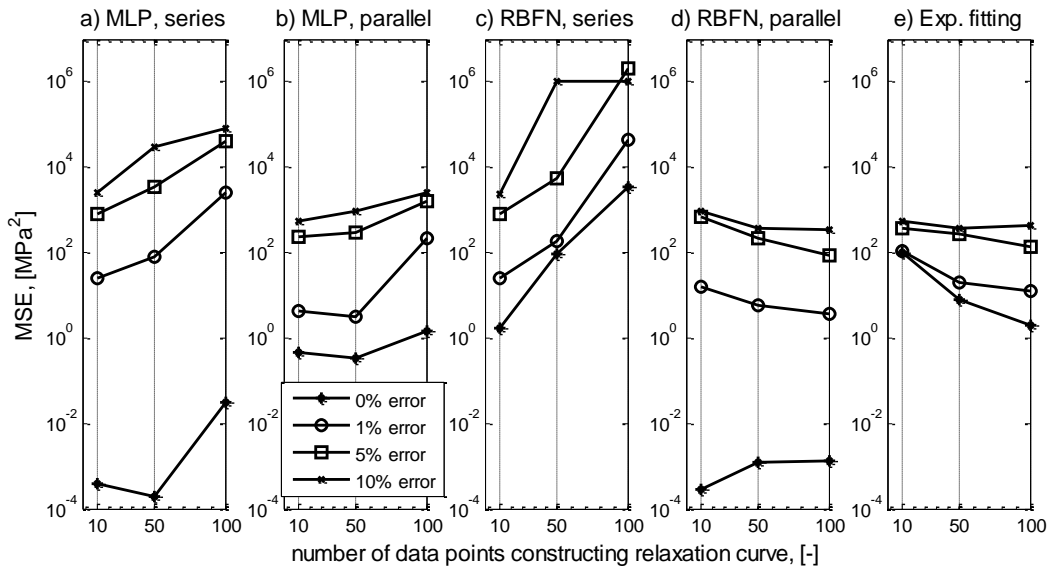
#### 4.1.3 Effect of number of data points

Fig. 8 shows  $MSE$  in dependence on the level of additive noise in the input data for each of the modeling methods at different number of data points  $n$  in the data set. The  $MSE$  was calculated in the same way as for robustness evaluation.

For noiseless data trend of increasing  $MSE$  with increasing number of data points is observed for all methods with exception of MLP. MLP in this case performs the best for 50 data points representing relaxation modulus curve. This might be explained by the optimal relation between number of free parameters of the network and number of data points in the training data.

We can observe that in the case of using RBFN with parallel input data (Fig. 8, d) and exponential fitting for noisy data (Fig. 8, e) the  $MSE$  decreases with increase of number  $n$  of data points, while the opposite trend is detected for all other methods applied to the noisy data.

As presented in Fig. 8 the lowest values of the average  $MSE$  are obtained with 10 data points for both data types using MLP and RBFN with data presented in series. Further we can see that NN with input data provided in series demonstrated the highest  $MSE$  among the compared methods (lowest performance) independently on the number of data points.



**Fig. 8.**  $MSE$  in dependence of number of data points representing relaxation modulus curve obtained by applying a) MLP with parallel input data, b) MLP with input data in series, c) RBFN with parallel

input data, d) RBFN with input data in series, and e) exponential fitting for data with 0, 1, 5 and 10% added noise

## 5 Conclusions

With the increasing use of polymers in demanding applications requiring control of structure health, systems that are not only capable to detect geometrical changes, such as cracks, but also changes related to time-dependency of polymers are required. The system should be able of determination of time-dependent material properties based on external excitation and, therefore to solve an inverse problem. Existing numerical techniques cannot be used for real-time application, therefore neural networks are suggested.

The paper proposes artificial neural networks as a tool for solving inverse problem appearing within characterization of time-dependent properties of relaxation modulus of viscoelastic materials. The simplest case with known closed-form solution was considered for obtaining a segment of relaxation modulus from constant strain rate tensile test data.

The investigation showed that MLP and RBFN of different topologies are capable of solving the stated problem and shows good generalization capabilities. MLP with data provided in series showed better generalization compared with parallel data feeding both measures of performance the of  $MSE$  and  $R_{0.05}$ . Opposite behavior was observed for RBFN.

Considering the robustness, the RBFN with data provided in parallel showed better performance compared to the other NNs and exponential fitting for high number of data points ( $n=50$  and  $100$ ).

Neural networks demonstrated better performance for datasets with smaller number of data points compared to exponential fitting, while the latter works better with larger number of data points in a set. This can be attributed to the utilization of non-linear least square regression algorithm, since small number of data points limit number of parameters of the methods that leads to loss of performance.

Observed decrease of performance of MLPs and RBFN using serial input data, with increase of the number of data points can be related to the ratio of free parameters of the system (NN) to the number of data points in the full training set.



The results show that further development of artificial neural networks, particularly RBFN, is promising for application to real-time health polymer structure monitoring. Generalization and robustness properties of the network, exceeding performance of non-linear parametric exponential regression, as well as possibility to operate in a real time are the advantages of NN compared to conventional methods used to determine time-dependent mechanical properties from non-standard experiments. Further investigation should address such problems as dynamic loadings, real-time data prediction, precision of detection of time-dependent changes, training procedure upgrade.

## 5 Acknowledgements

Authors would like to thank Slovenian Research Agency for financial support in frame of programs P2-0264 and P2-0241.

## 6 References

- Adler, A. & Guardo, R., 1994. A Neural Network Image Reconstruction Technique for Electrical Impedance Tomography. *IEEE Transactions on Medical Imaging*, 13(4), pp.594–600.
- Baddari, K. et al., 2010. Acoustic impedance inversion by feedback artificial neural network. *Journal of Petroleum Science and Engineering*, 71(3-4), pp.106–111.
- Czél, B., Woodbury, K.A. & Gróf, G., 2013. Inverse identification of temperature-dependent volumetric heat capacity by neural networks. *International Journal of Thermophysics*, 34(2), pp.284–305.
- Demuth, H., Beale, M. & Hagan, M., Neural Network Toolbox™ 6 User's Guide.
- Elshafiey, I., Udpa, L. & Udpa, S.S., 1994. Application of neural networks to inverse problems in electromagnetics. *IEEE Transactions on Magnetics*, 30(5), pp.0–3.
- Elshafiey, I., Udpa, L. & Udpa, S.S., 1995. Solution of inverse problems in electromagnetics using Hopfield neural networks. *IEEE Transactions on Magnetics*, 31(1), pp.852–861.
- Ferry, J.D., 1980. Viscoelastic properties of polymers. , p.650.
- Hagan, M.T., Demuth, H.B. & Beale, M.H., 1996. *Neural network design*,
- Haykin, S., 1999. *Neural networks: a comprehensive foundation*,
- ISO 527-1, 2012. *Plastics - Determination of tensile properties - Part 1: General principles*,
- Knauss, W.G. & Zhao, J., 2007. Improved relaxation time coverage in ramp-strain histories. *Mechanics of Time-Dependent Materials*, 11(3-4), pp.199–216.
- Lampinen, J. & Vehtari, A., 1999. Using Bayesian Neural Network to Solve the Inverse Problem in Electrical Impedance Tomography. *Proceedings of the Scandinavian Conference on*

*Image Analysis*, 1.

- Li, M.M. et al., 2009. Intelligent methods for solving inverse problems of backscattering spectra with noise: a comparison between neural networks and simulated annealing. *Neural Computing and Applications*, 18(5), pp.423–430.
- Li, M.M. et al., 2008. RBF neural networks for solving the inverse problem of backscattering spectra. *Neural Computing and Applications*, 17(4), pp.391–397.
- MacKay, D., 1992. Bayesian interpolation. *Neural computation*.
- Nguyen, D. & Widrow, B., 1990. Improving the Learning Speed of 2-Layer Neural Networks by choosing Initial Values of the Adaptive Weights. In *IJCNN International Joint Conference on Neural Networks, vol.3*. pp. 21–26.
- Samarskii, A.A., Vabishchevich, P.N. & De, W., 2009. *Numerical Methods for Solving Inverse Problems of Mathematical Physics*, Walter de Gruyter.
- Sammany, M., Pelican, E. & Harak, T. a., 2010. Hybrid neuro-genetic based method for solving ill-posed inverse problem occurring in synthesis of electromagnetic fields. *Computing*, 91(4), pp.353–364.
- Sammur, C. & Webb, G., 2011. Chapter 19 - Topology of neural Network. *Encyclopedia of Machine Learning*, pp.988–989. Available at:
- Saprunov, I., Gergesova, M. & Emri, I., 2014. Prediction of viscoelastic material functions from constant stress- or strain-rate experiments. *Mechanics of Time-Dependent Materials*, 18(2), pp.349–372.
- Sjoberg, J. & Ljung, L., 1992. Overtraining, Regularization and Searching for Minimum in Neural Networks. , pp.1–16.
- Tikhonov, A. & Arsenin, V., 1977. *Solutions of Inverse Problems*, Washington D.C.: John Wiley & Sons.
- Tscharnuter, D., Jerabek, M., Major, Z. & Lang, R.W., 2011. On the determination of the relaxation modulus of PP compounds from arbitrary strain histories. *Mechanics of Time-Dependent Materials*, 15(1), pp.1–14..
- Werbos, P., 1994. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*, John Wiley & Sons, Inc.
- Xiao, W., Lind, M.G.J., Dunford, W.G. & Capel, A., 2006. Real-time identification of optimal operating points in photovoltaic power systems. *IEEE Transactions on Industrial Electronics*, 53(4), pp.1017–1026.