

# Neural fuzzy network configurator for calculating a residual life of production equipment

M E Semenov<sup>1</sup>, S A Ilina<sup>1</sup> and A L Rutskov<sup>2</sup>

<sup>1</sup>National Research Tomsk Polytechnic University, Lenin Avenue, 30, Tomsk, Russia, 634050

<sup>2</sup>Branch of JSC "UK EFKO" in Voronezh, Taranchenko St., 40, Voronezh, Russia, 394018

E-mail: <sup>1</sup>sme@tpu.ru

**Abstract.** In this paper, an intelligent fault diagnosis scheme based on big data analysis method and adaptive neuro-fuzzy inference system (ANFIS) is proposed. An experimental study is given on the example of an electric drive of a stirrer of a hydrogenator reactor to illustrate the effectiveness of the proposed method and algorithmic solutions. Numerical experiments demonstrated the possibility of scaling proposed methodology for a wide class of similar process objects with the achievement of accuracy not less than 96.5%. The residual life calculation based on the ANFIS model have been carried out.

## 1. Introduction

Nowday, systems for monitoring the condition of production equipment are applied to collect real-time data by multiple sensors. The amount of data collected opens new perspectives for processing and discovering valuable information from such big data [1, 2].

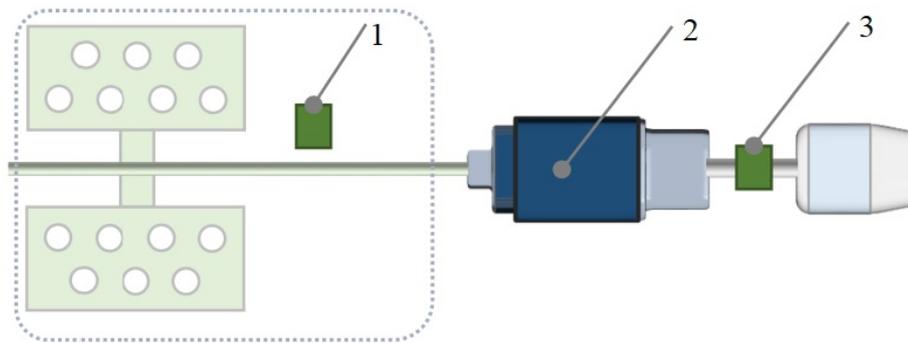
In the paper [3], the various techniques, models and algorithms on machinery diagnostics and prognostics were reviewed. Authors divided the approaches into three main categories: a) statistical (multivariate statistical methods, regression models) [2], b) model-based (different state space models) and c) artificial intelligent (AI) ones (fuzzy logic, neural networks, evolutionary algorithms or their combination).

We pay special attention to intelligent techniques fault diagnosis: self-organising neural networks [4], dynamic wavelet neural networks [5], recurrent neural network [6, 7], recurrent neural networks and neural-fuzzy inference systems [8], neural-fuzzy approach [9–14], hybrid intelligent methods [1], applied to estimation the different features of process equipment: remaining useful life [3], mean-residual-life [9], onset of a failure or predicting the time of ultimate failure [15].

Fuzzy inference systems based on the principles of human thinking and logic are what makes them different from other machine learning methods. Fuzzy rules and membership functions are fairly easy to interpret, which allows us to understand the fuzzy inference system.

Neuro-fuzzy systems combine the advantages of the individual approaches noted above. Thus a neural network can be used to tune the parameters of fuzzy inference system, in turn, the fuzzy logic principles can improve the performance of a neural network in estimating nonlinear data sets. Among such systems, we note the adaptive neuro-fuzzy inference system (ANFIS) [9, 11], it combining neural network and a fuzzy rule-based system of Mamdani type [15], the Takagi-Sugeno type [16] or non-parametric antecedents [17].





**Figure 1.** Sensors of an electric drive of a stirrer of a hydrogenator reactor: 1 – oil level sensor, 2 – electronic and an infrared temperature sensors, 3 – current sensor.

The purpose of this study is the implementation of a neural fuzzy network (NFN) configurator based on the ANFIS model to calculate the residual life of production equipment.

## 2. Experimental setup and methodology

Figure 1 shows the composition sensors of an electric drive of a stirrer of a hydrogenator reactor, which is the key device in the cycle of processing vegetable fats: a float-type oil level sensor – *Owen FLS – 1* (Owen Ltd, Russia) allows us to calculate a resistance of the oil volume to the mixing process, a linearized vibration sensors – IFM electronic and an infrared temperature sensors *Kelvin – 2* allow us to estimate a wear rate of gearbox bearings, an *ACT current sensor – 3* allows us to estimate an electromagnetic torque generated by an electric motor. Based on sensors 1–3 we can to model the total loading of the medium on a reducer of an electric drive of a stirrer of a hydrogenator reactor.

The time left to failure, given the current state of the equipment and the past performance profile, is one of the widely used functions and is the target variable for prediction. The time left before observing a failure is usually called *remaining useful life* [1, 3]. It is defined as the *conditional random variable*  $l_t = T - t | T > t, Z(t)$ , where  $T$  denotes the random variable of time to failure,  $t$  is the current age and  $Z(t)$  is the past condition profile up to the current time.

In our research, we propose to calculate the residual life of an equipment:

$$R = \left( 1 - k_{nuni f} \cdot \frac{t_2}{t_1} \right) \cdot 100\%, \quad (1)$$

where  $t_1, t_2$  are the real and nominal operating time of the equipment, respectively, and  $k_{nuni f}$  is the load non-uniform factor of equipment (load factor), it is a deviation of the nominal load from the real one. The  $k_{nuni f}$  domain can be defined from dataset range.

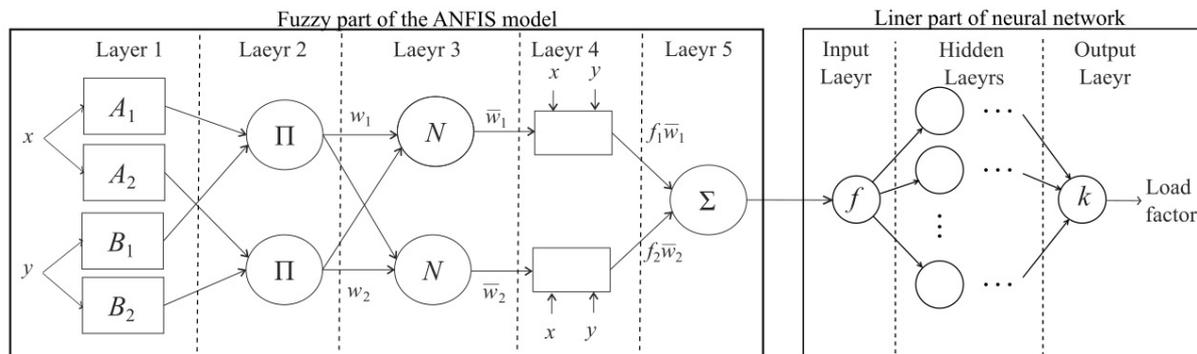
## 3. Fuzzy classifier structure

The Mamdani type rule of defuzzification [15] is used for classification tasks while the Takagi-Sugeno [16] – for an approximation tasks. The fuzzy inference system performs a fuzzy reasoning through next steps:

(i) A fuzzy implication of the form:

$$\text{IF } x \text{ IS } A, \text{ THEN } y \text{ IS } B,$$

where  $x \in X$  is an input variable,  $X$  is a field of definition of the premise of a fuzzy rule,  $y \in Y$  is an output variable,  $Y$  is a field of definition of the subconclusion of a fuzzy



**Figure 2.** General scheme of a neural fuzzy network based on the ANFIS model.

rule;  $A, B$  are fuzzy sets defined on  $X, Y$  with membership functions  $\mu_A(x) : X \rightarrow [0, 1]$ ,  $\mu_B(y) : Y \rightarrow [0, 1]$ .

(ii) Fuzzy subconditions of the form:

$$x' \text{ IS } A',$$

where  $x'$  is an actual value of the variable  $x$ ,  $A'$  is a fuzzy set representing the value of  $x'$ .

(iii) Fuzzy subconclusions of the form:

$$y' \text{ IS } B',$$

where  $y'$  is an actual value of the variable  $y$ ,  $B'$  is a fuzzy set representing the value of  $y'$ .

In this article, we use the Takagi-Sugeno fuzzy inference system [16]. The general view of this model can be represented in a vector form:

$$\text{IF } X \text{ IS } A, \text{ THEN } Y = f(X), \quad f(X) : X \rightarrow \mathbb{R},$$

here conditions are represented via membership functions. A linear polynomial is usually used to conclusion function  $y_i = p_{i0} + \sum_{j=1}^n p_{ij}x_j$ , where  $p_{ij}$  are the coefficients adjusted in the process of training the network,  $n$  is the overall number of fuzzy rules.

#### 4. Setting parameters of a fuzzy classifier

A neural fuzzy network (NFN) have a fuzzy and linear parts (Figure 2). The fuzzy part includes the ANFIS model that consists of five layers (Figure 2, left). The two-input  $x$  and  $y$ , the one-output Takagi-Sugeno fuzzy inference system  $f$  carries two membership functions  $\Pi$  for premise parameters, namely,  $A_1, A_2$  and  $B_1, B_2$  is illustrated [9, 11].

The *Layer 1* acquires the inputs  $x, y$  and introduces them to the ANFIS. The *Layer 1* is considered as the input of the fuzzy system. Every node of *Layer 1* is an adaptive node with a node function  $\mu_{A_i}(x), \mu_{B_i}(y)$ ,  $i = 1, 2$ . The output of the *Layer 1* becomes the input of the *Layer 2* carrying prior values of membership functions  $\Pi$  that are allocated based on the input values. The nodes on the *Layer 2* decide the fuzzy rules and send them to the *Layer 3* with a related degree of activity

$$w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2.$$

Then, degrees of activity are normalized to

$$\bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2$$

on the nodes  $N$  of the *Layer 3*. The *Layer 4* adopts the nodes  $N$ , function  $\Pi$  and inputs  $(x, y)$ :

$$f_i \cdot \bar{w}_i = (p_i x + q_i y + r_i) \cdot \bar{w}_i, \quad i = 1, 2$$

where  $p_i$ ,  $q_i$ , and  $r_i$  are the consequent parameters, and provides a first-order model by derived parameters sending them to the summator  $\Sigma$  on the *Layer 5* [11]:

$$f = \sum_i f_i \bar{w}_i.$$

Here, a first order model is one in which all output membership functions must be of the same type.

The Takagi-Sugeno model is used on the *Layer 4* of ANFIS. Fuzzy inference membership functions can be represented as sigmoidal  $f_S(x, a, b)$ , trapezoidal  $f_T(x, a, b, c, d)$ , or Gaussian function  $f_G(x, \mu, \sigma)$ :

$$f_S(x, a, b) = \left(1 + e^{-a(x-b)}\right)^{-1}, \quad a > b, \quad a, b \in \mathbb{R}, \quad (2)$$

$$f_T(x, a, b, c, d) = \begin{cases} 0, & x < a, d \leq x \\ \frac{x-a}{b-a}, & a \leq x < b \\ 1, & b \leq x < c \\ \frac{d-x}{d-c}, & c \leq x < d \end{cases} \quad a \leq b \leq c \leq d, \quad a, b, c, d \in \mathbb{R}. \quad (3)$$

$$f_G(x, \mu, \sigma) = e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad \mu, \sigma^2 - \text{expectation and variance of dataset, respectively.} \quad (4)$$

A user can choose one of the membership functions  $\{f_S, f_T, f_G\}$  to assess a particular parameter, this makes it possible to describe different assessment systems, since there are options for membership functions of different types. It should be noted the sigmoidal membership function (2) can be S-shape or Z-shape depending on the sign( $a$ ).

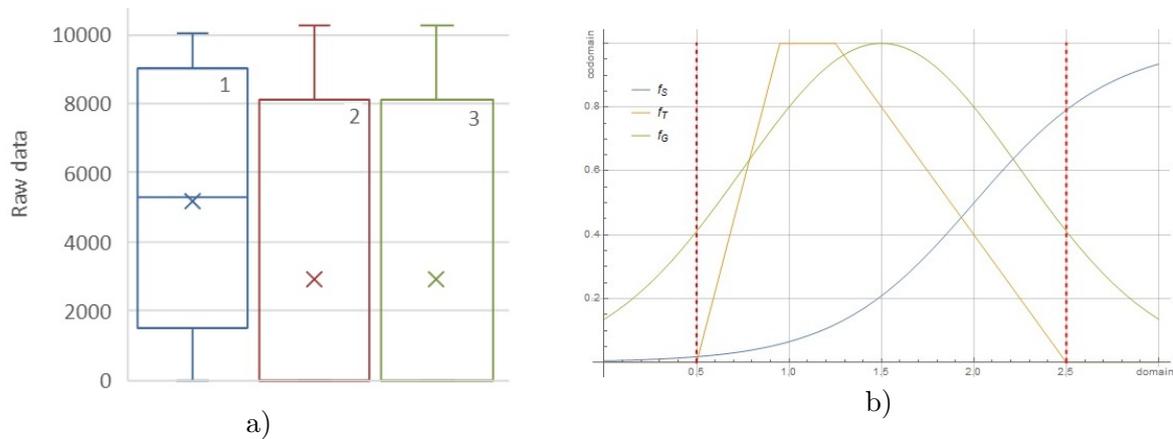
The linear part (Figure 2, right) consists of the *Input layer* with one neuron, several *Hidden layers* with an arbitrary number of neurons on each of them, and the *Output Layer* with one neuron. The *Input layer* acquires output of the summator  $\Sigma$  on the *Layer 5* and introduces it to the first *Hidden layer*. Each *Hidden layer* applies a different transformations to the input and passes results to the next one. The *Output layer* takes the data from the last *Hidden layer* and transforms it into a one-dimensional vector.

## 5. Numerical experiments

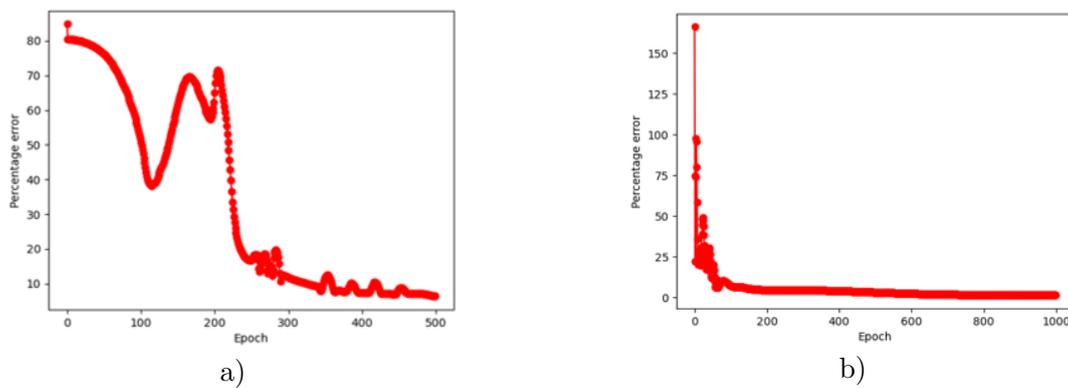
The input dataset for the proposed configurator consists of three vectors: a) a weighted sum vector for all sensors, b) a weighted sum vector for all operational reports, and c) a weighted sum vector of the simulation model data. The input data size is 46,077 observations with 1 second interval. In Figure 3a, the box plot of input raw data is shown. One can see the lower quartile equals median for the Kelvin (red box) and the IFM electronic (green box) sensors.

In our case, average values of normalized input data are define the membership functions domain  $D = [0.5, 2.5]$  while the codomain is always the interval  $[0, 1]$ . Following the idea [16], we estimated the parameters of the membership functions on the normalized dataset. In Figure 3b, one can see the the domain and codomain of used membership functions (2)-(4):  $f_S(x, \frac{8}{3}, 2)$ ,  $f_T(x, 0.5, \frac{19}{20}, \frac{25}{20}, 2.5)$ ,  $f_G(x, 1.5, 0.75)$ .

The NFN combines fuzzy and neural network techniques, which means we can use neural network training algorithms to tune the NFN. The NFN training was carried out according to the table containing  $n = 26$  times points. As a result, the NFN includes three membership functions



**Figure 3.** a) Box plot of input raw data (numbers correspond to Figure 1), b) the domain and codomain of membership functions.



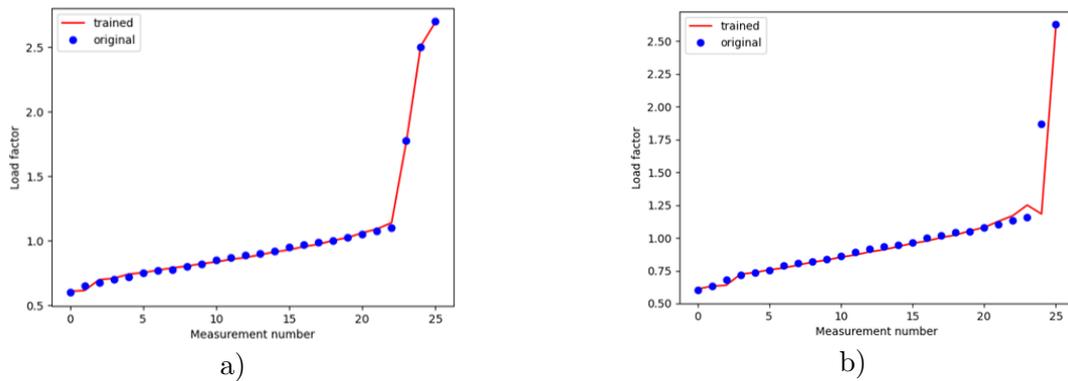
**Figure 4.** Errors in training the a) fuzzy, b) linear part of the NFN.

on each input neuron, uses a Gaussian membership function  $f_G$  for a neuron corresponding to the weighted sum of sensors data, and trapezoidal functions  $f_T$  for the rest, three linear *Layers 2-4* (Figure 2) with ten neurons on each, and two sigmoidal activation functions  $f_S$ . We used an error back propagation algorithm where the mean of absolute percentage error is calculated [1]:

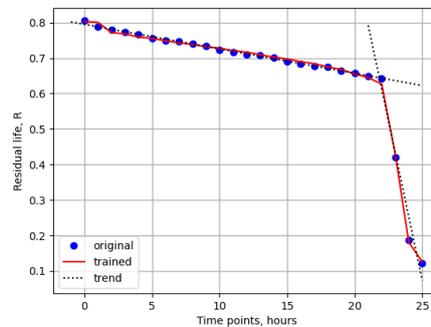
$$\varepsilon = \frac{1}{n} \sum_{i=1}^n \frac{|k_i - k_i^p|}{k_i} \cdot 100\%, \quad (5)$$

$k^p$ ,  $k$  are the  $n$ -size vectors of predicted and actual values of the load factor, respectively. With stochastic gradient descent as an optimizer at the training stage of 1,500 epochs, for the NFN the percentage errors (5) were calculated (Figure 4). In Figure 4a shown the curve of percentage error of the fuzzy part of the NFN, it is reached the stationary behavior after approximately 450 epochs. Figure 4b shows the curve of percentage error of the linear part of the NFN, it is reached stationary behavior after 100 epochs and then slowly decreased.

The comparison of the predicted load factor values,  $k^p$  with the actual ones both for the NFN training stage and for one of the tests are shown in Figure 5. The prediction error in the performed experiments does not exceed 3.5%. At the output of the configurator, the value of the load factor  $k_{nuni f}$  is obtained, which is in the range from 0.6 to 2.7 (Figure 5). Based on the obtained load factors for the training sample, the residual resource (1) was calculated with the initial values of the parameters  $t_1 = 32, 500$  and  $t_2 = 100, 000$  hours. A comparison of



**Figure 5.** Comparing the predicted values of the load factor with the actual values at the a) training stage, error  $\varepsilon = 1.02\%$ , b) test stage, error  $\varepsilon = 3.18\%$ .



**Figure 6.** Comparing the predicted values of the residual life with the actual values at the training stage, error  $\varepsilon = 0.86\%$ .

the calculated results with the actual values is shown in Figure 6. One can see the time series behavior used for training changes quite strongly on the observation interval. But the proposed model accounts for these changes at the 22-th times point. In Figure 6, we have shown two linear regression functions (dotted lines): on the time interval  $[0, 22]$  it is  $R = -0.000704t + 0.9793$ , while on the time interval  $[22, 25]$  it is  $R = -0.01823t + 1.362$ . The prediction error for these calculations is approximately 0.86%.

## 6. Conclusion

This article presents a neural fuzzy network configurator based on the adaptive neuro-fuzzy inference system, which is used to predict the load factor of production equipment. The software implementation is written in Python using the open source PyTorch machine learning framework. At the same time, it should be noted the possibility of scaling this methodology for a wide class of similar process objects with the achievement of accuracy not less than 96.5% in the conditions of short-term (up to one day) and medium-term forecasting (from a week to a month).

The proposed model based on the neural fuzzy network shows a good accuracy of residual life calculation in comparison with expert approaches used in industry (in particular, the average method on a sample for similar equipment). At the same time, the calculation of the the residual life of an equipment requires an increase in the number of experimental observations and the formation of an health monitoring and fault detection database.

## References

- [1] Lei Y 2017 *Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery* ed Lei Y (Butterworth-Heinemann) pp 231–280
- [2] Sun F, Wang N, Li X and Zhang W 2017 *IEEE Access* **5** 16277–16287
- [3] Jardine A K, Lin D and Banjevic D 2006 *Mechanical Systems and Signal Processing* **20** 1483–1510
- [4] Zhang S and Ganesan R 1997 *Journal of Engineering for Gas Turbines and Power* **119** 378–384
- [5] Vachtsevanos G and Wang P 2001 *2001 IEEE Autotestcon Proceedings. IEEE Systems Readiness Technology Conference. (Cat. No. 01CH37237)* pp 857–870
- [6] Yam R, Tse P, Li L and Tu P 2001 *The International Journal of Advanced Manufacturing Technology* **17** 383–391
- [7] Yang R, Huang M, Lu Q and Zhong M 2018 *IFAC-PapersOnLine* **51** 228–232
- [8] Wang W Q, Golnaraghi M F and Fathy I 2004 *Mechanical Systems and Signal Processing* **18** 813–831
- [9] Chinnam R B and Baruah P 2004 *International Journal of Materials and Product Technology* **20** 166–179
- [10] Lei Y, He Z and Zi Y 2008 *Expert Systems with Applications* **35** 1593–1600
- [11] Najafi B and Ardabili S F 2018 *Resources, Conservation and Recycling* **133** 169–178
- [12] Gai J, Hu Y and Shen J 2019 *Shock and Vibration* **2019** 1–10
- [13] Rutskov A L, Sidorenko E V, Burkovsky V L and Fedorov Y P 2020 *IOP Conference Series: Materials Science and Engineering* vol 791 p 012034
- [14] Lv Y, Zhou Q, Li Y and Li W 2021 *Advanced Engineering Informatics* **49** 101318
- [15] Verma A K, Srividya A and Ramesh P G 2011 *International Journal of System Assurance Engineering and Management* **2**(1) 14–20
- [16] Jang J S 1993 *IEEE Transactions on Systems, Man, and Cybernetics* **23** 665–685
- [17] Angelov P and Yager R 2011 *2011 IEEE Workshop on Evolving and Adaptive Intelligent Systems (EAIS)* pp 62–69